

一.緒論

1.1 前言

功能方塊指令為國際標準可編程邏輯控制語言(IEC 61131-3)之一，亦為圖形式編輯程式語言，每個功能方塊指令相當於一顆積體電路，各有其特定的控制功能，使用時可以搭配階梯圖、指令列、結構式文字來描述及定義其控制功能，因此也可以將它視為巨集指令。在使用上只需注意功能方塊指令的輸入、輸出的設定功能，而不必處理內部的複雜程式[圖 1.1]。功能方塊指令與電子線路圖中的信號流程圖非常相似，在程序中，可視為兩個過程元素之間的訊息流通，普遍地應用在過程控制領域上。

功能方塊指令是以矩形方塊來表示，每一功能方塊指令的左側有兩個以上的輸入接點，在右側有兩個以上的輸出接點[圖 1.2]，其類型名稱通常寫在方塊指令內，但實際的名稱通常標示在方塊指令的上方，功能方塊指令的輸入/輸出名稱標示在方塊指令的輸入/輸出點的相對應地方。

1.2 研究動機與目的

目前業界上已具有功能方塊指令功能之可程式邏輯控制器(PLC)軟體有西門子(SIEMENS)、洛克威爾(Allen Bradley，簡稱AB)、三菱(Mitsubishi)、歐姆龍(OMRON)等，而WINPC32 PLC的應用軟體模組

之可程式邏輯控制模組(Soft-PLC Controls)方面，其程式控制軟體的骨架是以可程式邏輯控制器為基礎，也採用國際標準可編程邏輯控制語言，在Window作業系統環境下編輯/編譯，目前使用三種不同的編寫語言：階梯圖(Ladder Diagram，簡稱LD)、指令列(Instruction List，簡稱IL)及結構式文字編程語言(Structure Text，簡稱ST)來建構流程控制動作，但由於在歐洲的可程式邏輯控制語言皆偏好使用功能方塊指令來進程式的撰寫，所以在此論文中將探討以WINPC32 PLC軟體作業平台上來建構功能方塊圖語言的功能方塊指令，以增加WINPC32 可程式邏輯控制語言的選擇性及競爭性。



二.研究內容及方法

在此將以A廠PLC軟體架構為參考依據，首先將功能方塊指令進行功能分析及了解進而轉換成階梯圖語言的型態，然後再轉換成WINPC32 PLC階梯圖語言的型態，再將階梯圖語言轉換成功能方塊指令。

2.1 A廠功能方塊指令轉換成A廠階梯圖程式語言

A廠功能方塊指令共有87個，首先將A廠每一功能方塊指令的屬性和元件加以了解及分析，在此以加法器(ADD)功能方塊指令及取最小值(MINC)功能方塊指令為例來做說明：

2.1.1 加法器功能方塊指令

加法器功能方塊指令之屬性[表2.1]有輸入/輸出參數、資料型態及描述等項目，指令接腳有指令執行啟動輸入點(Enable In)、來源A(Source A)、來源B(Source B)、指令執行啟動輸出點(Enable Out)、指令運算結果輸出點(Dest)等。

1. 指令執行啟動輸入點---資料型態為布林(Boolean)，當值為0時其指令無法運作執行以及指令執行啟動輸出點之值也不會被更新，其輸入點之預設值為1。
2. 來源A及來源B---資料型態為實數(Real)，來源A的值將會與來源B的值相加，其來源可為一固定數值或是標記(Tag)。

3. 指令執行啟動輸出點---資料型態為布林值，輸出點之值隨指令執行啟動輸入點值而更新。
4. 指令運算結果輸出點---當指令運作時，其值為來源A及來源B相加後結果值,若指令無法運作時其輸出值為0。

2.1.2 取最小值功能方塊指令

取最小值功能方塊指令之屬性如[表2.2]所示，指令接腳有指令執行啟動輸入點、輸入值(In)、重置輸入點(Reset)、重置值(Reset Value)、指令執行啟動輸出點(Enable Out)、指令運算結果輸出點(Out)等[圖2.1]。

1. 指令執行啟動輸入點---資料型態為布林(Boolean)，當值為0時其指令無法運作執行以及指令執行啟動輸出點之值也不會被更新，其輸入點之預設值為1。
2. 輸入值---資料型態為實數、其輸入值會進入至指令內，其預置值為0.0。
3. 重置輸入點---資料型態為布林值(預設值為0)，當值為1時，會將重置值更新至指令運算結果輸出值；當值為0時，將會有三種狀況值出現：
 - (1)當輸入值小於上回運算結果輸出點值時，輸入值會取代運算結果輸出點值。

(2)當輸入值大於上回運算結果輸出點值時，運算結果輸出點值保持上回運算結果輸出點值。

(3)當輸入值大於上回運算結果輸出點值時，運算結果輸出點值保持上回運算結果輸出點值不變。

4. 重置值---資料型態為實數，當重置輸入點值為1時，會將重置值更新至指令運算結果輸出值，其預設值為0。

5. 指令執行啟動輸出點---資料型態為布林值，輸出點之值隨指令執行啟動輸入點值而更新。

6. 指令運算結果輸出點---當指令運作時，運算所得之值會置於該輸出點。

再將其功能方塊指令的功能及各接腳的位置轉換成A廠階梯圖程式語言方式來呈現，如[圖2.2]及[圖2.3]所示，再進行測試轉換後之功能是否與既有功能方塊指令符合。

2.2 A廠PLC階梯圖語言轉換成WINPC32 PLC階梯圖語言

因A廠階梯圖與WINPC32 PLC階梯圖程式語言型態相近，若要轉寫成WINPC32 PLC階梯圖程式語言之困難度低，雖然有些階梯圖指令會有差異皆可克服。如[圖2.4]及[圖2.5]所示為加法器(ADD)功能方塊指令及取最小值(MINC)功能方塊指令所轉換成之WINPC32 PLC階梯圖程式。

三.理論

這一章中將說明A廠 PLC功能方塊圖程式及WINPC32 PLC階梯圖程式軟體架構，以下分別述之。

3.1 A廠PLC功能方塊圖程式軟體架構

A廠控制器的編輯及規劃軟體，具有單一編輯軟體且符合國際標準可編程邏輯控制語言，其指令有功能方塊圖、階梯圖、順序功能流程圖(SFC)、結構式文字編程語言以及能同時編寫順序控制、運動控制，製程控制及變頻控制，也能跟其它的語言編輯介面完全整合在一起，使用既有的資料庫、選單、結構、控制器組織、輸入/輸出及規劃等。

功能方塊指令集目前合計有87個可運用，可從視窗之指令元件工具列[圖3.1]加以選取,以功能屬性可分為流程指令(Process Instructions)、驅動指令(Drives Instructions)、過濾指令(Filters Instructions)、選擇及界限(Select/Limit Instructions)、統計指令(Statistical Instructions)、位元指令(Bit Instructions)、計時及計數指令(Timer/Counter Instructions)、比較指令(Compare Instructions)、數學運算指令(Compute/math Instructions)、搬移及邏輯指令(Move/Logical Instructions)、程式控制指令(Program/control Instructions)、三角函數指令(Trig Functions Instructions)、高級數學

運算指令(Advanced Math Instructions)、數學轉換指令(Math Conversions Instructions)等，其內部資料型態有布林(Boolean)、簡易整數(SINT)、整數(INT)、雙整數(DINT)、實數[表3.1]，在作業環境上功能方塊指令元件可分類成指令、輸入參考點(Input reference)、輸出參考點(Output reference)、連接線(Wires)、連接線連接器(Wire connectors)、表單(Sheet)。

3.1.1 功能方塊指令元件

1. 指令

功能方塊指令內部有屬性(Properties)功能表，可了解該指令的參數(Parameter)及標記(Tag)之設定值，且可加以選擇設定[表3.2]。

2. 輸入參考點

可為一個標記或是帶入一個數值，從記憶暫存區到進入功能方塊指令程式裡。

3. 輸出參考點

可定義為一個標記或是帶出一個數值，從功能方塊指令程式到進入記憶暫存區裡[表3.3]。

4. 連接線

主要功能為連接指令、輸入及輸出參考點和連接線連接器，可

分為類比連接線和數位連接線兩種型式，類比連接線又稱為實線，負責類比資料傳輸；數位連接線又稱為虛線，負責數位資料傳輸[圖3.2]。

5. 連接線連接器

可分為兩種：

(1) 輸入連接線連接器(Input Wire Connector，簡稱ICON) --- 從

輸出連接線連接器內接收到一個數值或是一個標記。

(2) 輸出連接線連接器(Output Wire Connector，簡稱OCON) ---

可送出一個數值或是一個標記到輸入連接線連接器裡 [圖

3.3]。

6. 表單

在同一程式程序中，表單為指令編輯工作區，且可建立無數個工作表單。

3.1.2 建立功能方塊圖程式

若要建立功能方塊圖程式，需先在程式架構下新增程序(Routine)。其編輯程式選擇”Function Block Diagram”選項[圖3.4]，開啟功能方塊圖編輯畫面便可進行編輯，在指令工具列上選擇所需之指令便可開始程式撰寫。一旦程式撰寫完成，尚須執行完下列步驟，其程式才算是可執行的程式，執行步驟為：1.驗證

功能方塊圖程式，2. 驗證這一功能方塊圖程式所屬之整個專案程式，3. 將整個專案程式執行下載程序。

3.2 WINPC32 PLC階梯圖程式軟體架構

在WINPC32 PLC階梯圖提供下列幾個迴圈，核心迴圈(Kernel Loop)，快速迴圈(Fast Loop)，標準迴圈(Normal Loop)，供使用者針對其工作進行分類，緊急快速的I/O點動作，可以置於核心迴圈或快速迴圈，一般動作可以置於標準迴圈。標準迴圈每執行一次的間隔取決於掃描時間。而掃描時間則是由程式碼的大小所決定。若程式碼越大，則掃描時間越長。程式碼的執行周期是不固定的，而是根據不同的事件而改變。若有三個PLC回路，所以應該要有三個結束函數，第一個結束函數之前為快速迴圈(每10ms執行一次)，第一個結束函數到第二個結束函數為初始迴圈(只執行一次，用以初始暫存器)，第二個結束函數到第三個結束函數為標準迴圈(每100ms 執行一次)[圖3.5]。

3.2.1 程式編輯的注意事項

在進行程式編輯時應遵守下列規則：

1. 不應該有無作用的回路或線段部分[圖3.6]。
2. 不應該有開路或斷線的部分[圖3.7]。
3. 一個回路的最後一個節點不應該沒有輸出型式的節點[圖3.8]。

4. 一行的第一個節點不應該是輸出型式的節點[圖3.9]。

5. 輸出線圈不能是串聯的型式[圖3.10]。

3.2.2 WINPC32 PLC階梯圖語言函數

WINPC32階梯圖語言中的函數分類成下列幾種型態：

1. 觸點函數(Contact)--- 觸點函數的ON/OFF 狀態可以決定下一個輸入節點或輔助節開/ 常閉觸點函數。
2. 線圈函數(Coil)--- 線圈函數是被用來表示輸出觸點或輔助觸點ON/OFF(高/低電位)的狀態。
3. 計時器函數(Timer Function)及計數器函數(Counter)。
4. 布林運算(Boolean Algebra Function)---主要可分AND 函數、OR 函數和XOR函數，並可作暫存器與數器之間的運算。
5. 算數函數(Arithmetic)--- 算術函數主要為加(Add)、減(Subtract)、乘(Multiply)、除(Divide)、傳送(Move)等函數，並可作為暫存器與數值暫存器。
6. 比較函數(Comparison)--- 可分為等於函數(Equal To Function)、大於函數(Larger Than Function)、小於函數(Smaller Than Function)及不等於函數(Not Equal To Function)，並可作暫存器與數暫存器之間的運算。
7. 流程控制(Flow Control Function) ---主要可分為結束函數(END

Function)、卷結束標函數(Label Function)、跳轉函數(Jump Function)、子卷標函數(Subroutine Label Function)、跳轉到副程式函數(Jump to Subroutine Function)、返回函數(Return Function)。

8. 指標運算(Pointer Function)--- 傳送指定暫存器中的數值至指標指向的暫存器(Move data to pointer)。

9. 數學運算(Mathematic Operate)---主要可為平方根(Square Root)、正弦函數(Sin)、餘弦函數(Cos)、正切函數(Tan)、反正弦函數(ASin)、反餘弦函數(ACos)、反正切函數(ATan)等運算指令。

10. 擴充運算(Extension Operate)---在擴充功能的對話窗中，可分成四大部分[圖3.11]：

(1) 目錄(Catalog)：以功能性來分類，目前有[OM] Omron 函式、[FM] Float Math 浮點數學運算、[FB] Float Basic 浮點數基本運算，[FC] Float Compares 浮點數比較運算和 [FT] Float Translate 浮點數和整數之間的轉換，使用者可以 Check/Un Check 選項，便於尋找所需要的函式。

(2) 註解(Comment)：功能註解其中會有運算式，其符號表示以“D”表示資料暫存器，以”#”表示直接數值，則表示直接數值。

如圖[F.3-13]所示 $[1] = [2] + [3]$ ，第一個參數的暫存器1 = 第二個參數的暫存器1 + 第三個參數的暫存器1，也就是浮點暫存器1 = 浮點暫存器2 + 浮點暫存器3。

(3)參數(Parameter)：可分成兩種暫存器，其旁邊會有出現流覽的按鍵，供使用者選擇所編輯的標記名稱。

(4) 功能表：顯示出所有支援的函數。

3.2.3 編譯階梯圖程式方式

編譯階梯圖程式可透過下列方式來達成：

- 1.透過WINPC32整合環境的”Build Project(F9)”來編譯整個專案。
- 2.透過程式樹狀結構下達”Project Compile/Project Build”，此時所有的程式會被編譯。
- 3.透過PLC的樹狀結構下達Module Compile/Module Build即可。

四.實驗結果

4.0 前言

由第二章及第三章針對功能方塊指令的介紹及說明,本章將進行 WINPC32 PLC功能方塊指令之撰寫及實現

4.1 WINPC32 PLC階梯程式編輯

首先進行PLC階梯程式編輯，以取餘數功能方塊指令為例

4.1.1 取餘數功能方塊指令

1.結構及功能說明：

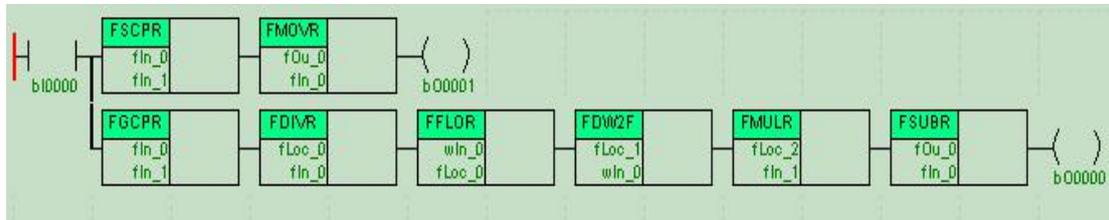
輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將被 Source B(被除數)進行除法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A(除數)進行除法計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

2.指令說明：

當 Source A = 9 Source B = 2 則 商數=4 餘數=1 所以 Result = 1

3. PLC 階梯程式編輯說明：

將其功能定義由階梯程式來進行編寫，其程式如下所示：



首先將Source A與Source B進行比較，若Source A小於Source B時，則將Source A值放入Result位置內，若Source A大於Source B時，則Source A除以Source B所得之值，進行無條件舍去方式獲得一整數值再將該值轉換成浮點數(Float)，與除數Source B進行乘法演算所得之值再與Source A進行減法演算，所獲得的結果值放入Result位置內。

4.2 WINPC32 PLC功能方塊指令程式結構說明

4.2.1 將編輯完成之階梯程式進行程式編譯

若編譯成功即可獲得PLC IL程式碼，此為指令運算主要程式如下列所示。

```
LD bI0
Extension FSCPR fR776 fR777
Extension FMOVR fR826 fR776
LD bO1
Extension FGCPR fR776 fR777
Extension FDIVR fR976 fR776 fR777
Extension FFLOR wR576 fR976
Extension FDW2F fR977 wR576
Extension FMULR fR978 fR777 fR977
Extension FSUBR fR826 fR776 fR978
LD bO2
```

4.2.2 定義功能方塊指令結構

將編譯完成之Ladder程式執行Convert to Function Block程序，開始定義程式之結構位置安排，如[圖4.1]所示。將程式內之各接點(Tagname)給予適當正確之資料型態、接點名稱、功能說明以及定義在功能方塊指令上欲顯現要使用之接點。

功能方塊指令程式結構可分為下列如下：

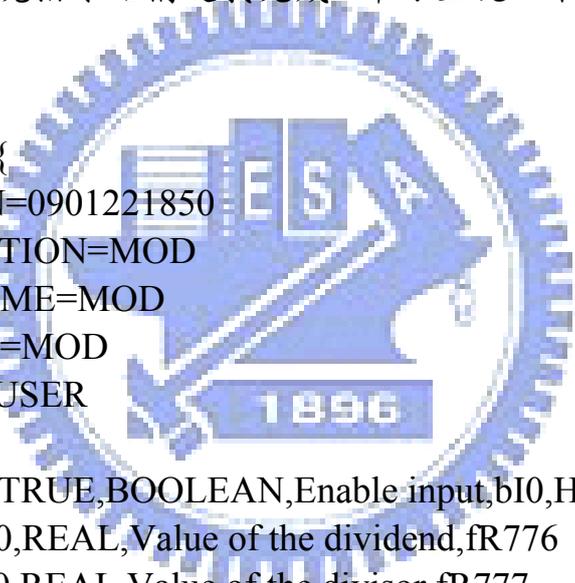
1. FBLOCK{...}：為該功能方塊指令之程式
2. VERSION
3. DESCRIPTION：為功能方塊指令說明欄，說明該指令為何種功能指令
4. CODENAME：為功能方塊指令之名稱
5. SYMBOL：為功能方塊指令在功能方塊編輯平台上呼叫對應功能方塊指令程式之符號。
6. GROUP：為該功能方塊指令之群組名稱。
7. INPUT{...}：為該功能方塊指令所擁有之輸入接點，分別說明各接點之名稱、預設值、資料型態、功能說明、指令主程式所對應之接點名稱，在指令程式編輯時必須定義清楚。

8. OUTPUT {...} : 為該功能方塊指令所擁有之輸出接點，分別說明各接點之名稱、預設值、資料型態、功能說明、指令主程式所對應之接點名稱，在指令程式編輯時必須定義清楚。

9. CODE {...} : 為該功能方塊指令運算主要程式，如4.2.1項所說明之。

當功能方塊指令結構定義完成，即可呈現如下列所示之指令結構程式：

構程式：



```
FBLOCK {  
VERSION=0901221850  
DESCRIPTION=MOD  
CODENAME=MOD  
SYMBOL=MOD  
GROUP=USER  
INPUT {  
Enable In,TRUE,BOOLEAN,Enable input,bi0,HIDE  
Source A,0,REAL,Value of the dividend,fR776  
Source B,0,REAL,Value of the divisor,fR777  
}  
OUTPUT {  
Enable Out,0,BOOLEAN,Enable output,bO0,HIDE  
Result,0,REAL,Result of the math instruction,fR826  
}  
GLOBAL {  
}  
VIDRIVE {  
}  
CODE { PLC IL程式碼 }  
}
```

4.3 WINPC32 PLC 功能方塊指令建立於平台

4.3.1 建立功能方塊指令 FBC 文件檔

如 4.2.2 項說明將編輯完成之 WINPC32 PLC 階梯語言程式進行 Convert to Function Block 程序，完成定義程式之結構位置後，便可在該專案檔案資料內發現所新建立之 FBC 文件檔。

範例 建立 MOD 功能方塊指令

其 MOD FBC 文件檔在路徑 C:\MyProject\Project0512\data\fb\MOD 找出。

1. C:\MyProject：為 WINPC32 PLC 功能方塊平台自動建立資料夾。
2. Project0512：為專案名稱
3. data：該專案程式資料夾
4. fb：若專案程式執行完成 Convert to Function Block 程序便會自動產生並將執行完成之 FBC 文件檔放置在裡面。

4.3.2 建立MOD功能方塊指令屬性資料夾

範例 建立 MOD 功能方塊指令屬性之資料夾

- 1.在路徑 C:\Program Files\Hurco\WINPC32\FBDI\AB 上建立新資料夾並命名為 Compute。
- 2.將 MOD FBC 文件檔放置於 Compute 資料夾內。

4.3.3 建立 MOD 功能方塊指令在平台上所對應之元件資料

- 1.建立符號圖片(命名：MOD)及群組圖片(命名：Compute)其圖片類型為.BMP 格式以及尺寸為 16 x 16 個像素，放置於 Compute 資料夾內。
- 2.重新開啟 WINPC32 PLC 功能方塊編輯軟體，便可在編輯平台工具列上使用所建立之 MOD 功能方塊指令。



4.4 WINPC32 PLC功能方塊指令屬性種類及驗證

4.4.1. WINPC32 功能方塊圖語言中的函數分類可分成下列幾種型態

1. 過程指令(Process Instructions)
2. 驅動指令(Drives Instructions)
3. 選擇/限制指令(Select/Limit Instructions)
4. 統計指令(Statistical Instructions)
5. 位元觸發指令(Bit Instructions)
6. 計時/計數指令(Timer/Counter Instructions)
7. 比較指令(Compare Instructions)
8. 算數指令(Compute/math Instructions)
9. 搬移/邏輯指令(Move/Logical Instructions)
10. 三角函數指令(Trig Functions Instructions)
11. 高等數學指令(Advanced Math Instructions)
12. 數學轉換指令(Math Conversions Instructions)

4.4.2 過程指令(Process Instructions)

過程指令可分為下列指令：警報(Alarm ,ALM)、比例(Scale,SCL)、分時比例控制(Split Range Time Proportional , SRTP)、積算器(Totalizer , TOT)、延遲器(Dead Time , DEDT)，以及過程指令驗證及操作說明

1. 警報(Alarm , ALM)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
HH Limit	REAL	高高警報輸入值，輸入值為任一浮點數。
H Limit	REAL	高警報輸入值，輸入值為任一浮點數。
L Limit	REAL	低低警報輸入值，輸入值為任一浮點數。
LL Limit	REAL	低警報輸入值，輸入值為任一浮點數。

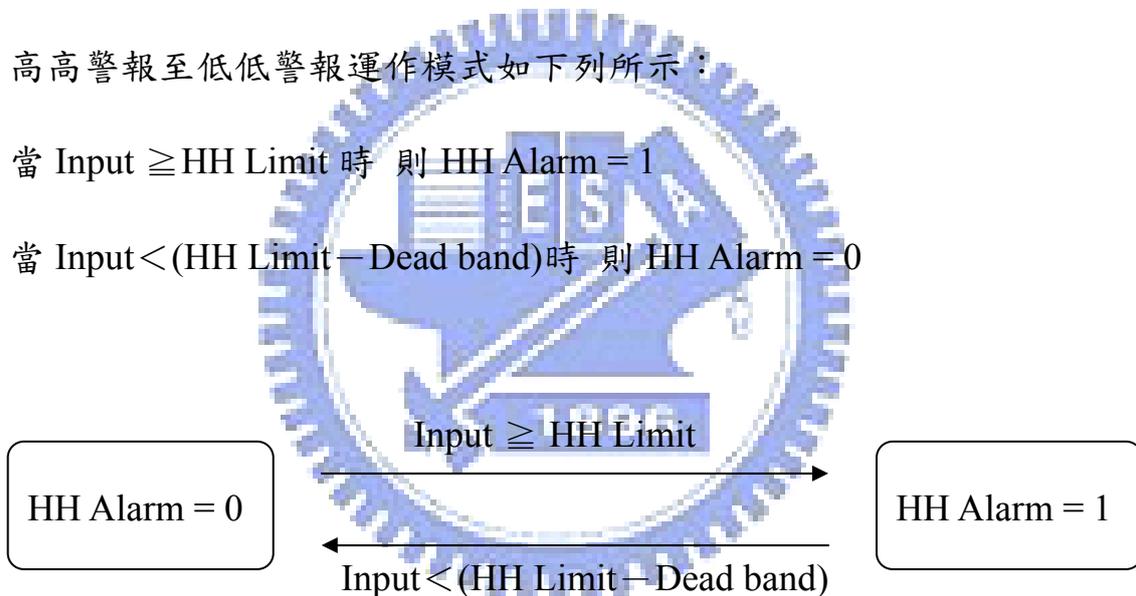
Dead band	REAL	高高警報和低低警報區間之警報緩衝區間值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
HH Alarm	BOOL	高高警報輸出值。
H Alarm	BOOL	高警報輸出值。
L Alarm	BOOL	低警報輸出值
LL Alarm	BOOL	低低警報輸出值

範例說明：

高高警報至低低警報運作模式如下列所示：

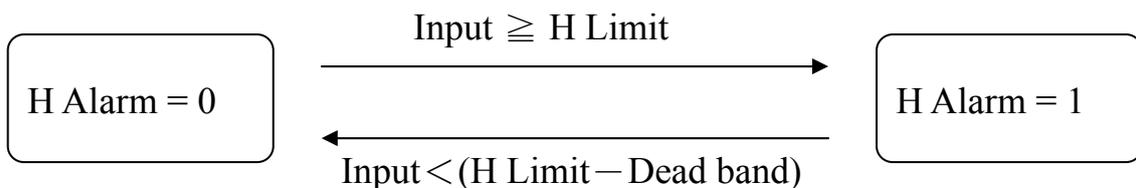
當 $\text{Input} \geq \text{HH Limit}$ 時 則 $\text{HH Alarm} = 1$

當 $\text{Input} < (\text{HH Limit} - \text{Dead band})$ 時 則 $\text{HH Alarm} = 0$



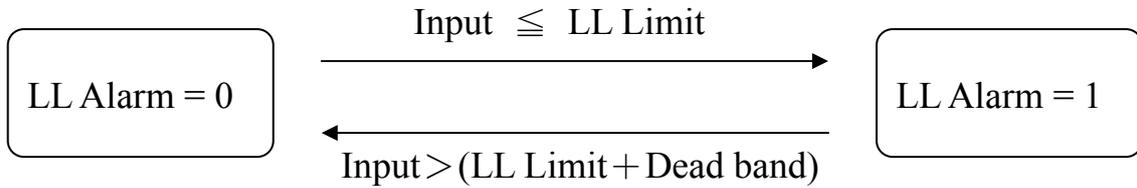
當 $\text{Input} \geq \text{H Limit}$ 時 則 $\text{H Alarm} = 1$

當 $\text{Input} < (\text{H Limit} - \text{Dead band})$ 時 則 $\text{H Alarm} = 0$



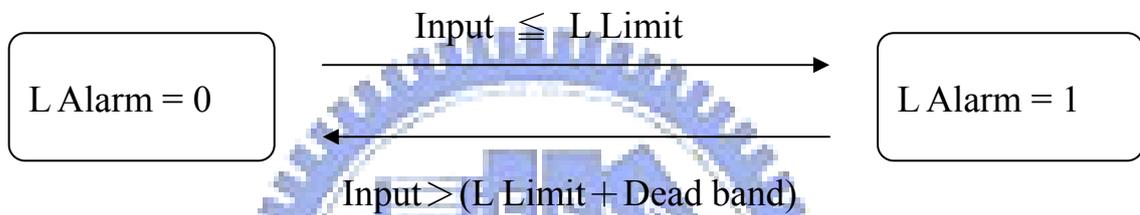
當 $\text{Input} \leq \text{LL Limit}$ 時 則 $\text{LL Alarm} = 1$

當 $\text{Input} > (\text{LL Limit} + \text{Dead band})$ 時 則 $\text{LL Alarm} = 0$



當 $\text{Input} \leq \text{L Limit}$ 時 則 $\text{L Alarm} = 1$

當 $\text{Input} > (\text{L Limit} + \text{Dead band})$ 時 則 $\text{L Alarm} = 0$



2. 比例(Scale, SCL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
In Max	REAL	高警報輸入值，輸入值範圍為 $\text{InRawMax} > \text{InRawMin}$ 。
In Min	REAL	低警報輸入值，輸入值範圍為 $\text{InRawMin} < \text{InRawMax}$ 。
Out Max	REAL	掃描最大限制輸入值，輸入值為任一浮點數，其預設值為 0，。
Out Min	REAL	掃描最小限制輸入值，輸入值為任一浮點數，其預設值為 0。

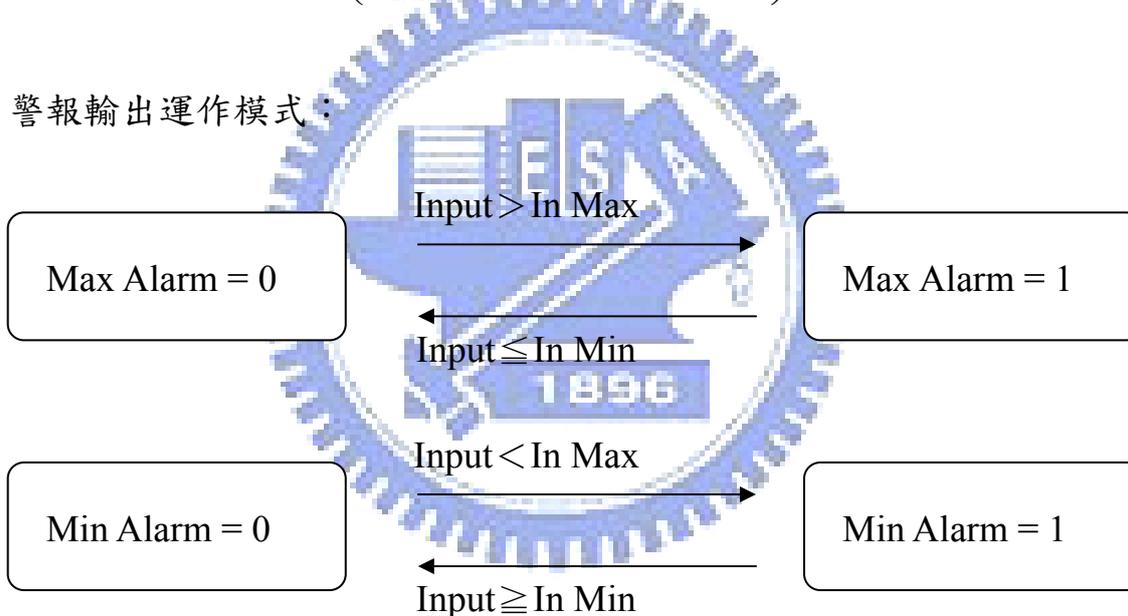
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令運算後輸出值。
Max Alarm	BOOL	高警報輸出值。
Min Alarm	BOOL	低警報輸出值

說明：

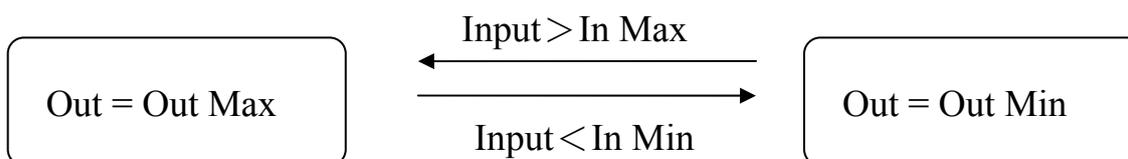
指令運算公式：

$$Out = (Input - InMin) \times \left(\frac{OutMax - OutMin}{InMax - InMin} + OutMin \right)$$

警報輸出運作模式：



限制值輸出運作模式：



3.分時比例控制(Split Range Time Proportional, SRTP)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	加熱或冷卻類比訊號輸入值，輸入值為任一浮點數。
Cycle Time	REAL	輸出時脈週期時間，輸入值為任一正向浮點數，其預設值為 0。
Max Heat In	REAL	加熱最大輸入值，輸入值為任一浮點數，其預設值為 100。
Min Heat In	REAL	加熱最小輸入值，輸入值為任一浮點數，其預設值為 50。
Max Cool In	REAL	冷卻最大輸入值，輸入值為任一浮點數，其預設值為 0。
Min Cool In	REAL	冷卻最小輸入值，輸入值為任一浮點數，其預設值為 50。
Max Heat Time	REAL	最大加熱時間，若 Heat Time > Max Heat Time 時將被受限為 Max Heat Time 值，當 Max Heat Time 輸入值無效時，將會被設定為 Cycle Time 值，輸入值範圍為 0~ Cycle Time，其預設值為 0。
Min Heat Time	REAL	最小加熱時間，若 Heat Time < Min Heat Time 時則 Min Heat Time 值將被重置為 0，當 Min Heat Time 輸入值無效時，也會該值設定為 0，輸入值範圍為 0~ Max Heat Time，其預設值為 0。
Max Cool Time	REAL	最大冷卻時間，若 Heat Time > Max Cool Time 時將被受限為 Max Cool Time 值，當 Max Cool Time 輸入值無效時，將會被設定為 Cycle Time 值，輸入值範圍為 0~ Cycle Time，其預設

		值為 0。
Min Cool Time	REAL	最小冷卻時間，若 Heat Time < Min Cool Time 時則 Min Cool Time 值將被重置為 0，當 Min Cool Time 輸入值無效時，也會該值設定為 0，輸入值範圍為 0~ Max Cool Time，其預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Heat Out	BOOL	加熱中訊號值
Cool Out	BOOL	冷卻中訊號值
Heat Time Percent	REAL	加熱輸出時間百分比
Cool Time Percent	REAL	冷卻輸出時間百分比

說明：

內部Internal timer值是由Delta T來進行更新，而Delta T是程式Scan一次所需時間,所以輸出作動與時間有密切關係，如果Cycle Time = 0時，則Internal timer將被重置歸零以及Heat Out = 0和Cool Out = 0。

計算加熱和冷卻時間：

$$\text{Heat Time} = \frac{\text{Input} - \text{Min Heat In}}{\text{Max Heat In} - \text{Min Heat In}} \times \text{Cycle Time}$$

當Heat Time < Min Heat Time時，則Heat Time = 0。

當Heat Time > Max Heat Time時，則Heat Time = Max Heat Time。

$$\text{Heat Time Percent} = \frac{\text{Heat Time}}{\text{Cycle Time}} \times 100$$

$$\text{Cool Time} = \frac{\text{Input} - \text{Min Cool In}}{\text{Max Cool In} - \text{Min Cool In}} \times \text{Cycle Time}$$

當 Cool Time < Min Cool Time 時，則 Cool Time = 0。

當 Cool Time > Max Cool Time 時，則 Cool Time = Max Cool Time。

$$\text{Cool Time Percent} = \frac{\text{Cool Time}}{\text{Cycle Time}} \times 100$$

控制加熱及冷卻輸出需遵守下列規則：

當 Heat Time \geq Internal cycle time accumulator 時，則 Heat Out = 1。

當 Internal Cycle Timer > Heat Time 時，則 Heat Out = 0。

當 Cool Time \geq Internal cycle time accumulator 時，則 Cool Out = 1。

當 Internal cycle timer > Cool Time 時，則 Cool Out = 0。

當 Cycle Time = 0 時，則 Heat Out=0 及 Cool Out = 0。



4.積算器(Totalizer, TOT)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)										
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。										
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。										
Time Base	DINT	時間模式輸入型態 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>值</th> <th>說明</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>秒</td> </tr> <tr> <td>1</td> <td>分</td> </tr> <tr> <td>2</td> <td>時</td> </tr> <tr> <td>3</td> <td>日</td> </tr> </tbody> </table> 若輸入值非上列值時指令將無法更新輸出值，其輸入值範圍為 0~3，預設值為 0。	值	說明	0	秒	1	分	2	時	3	日
值	說明											
0	秒											
1	分											
2	時											
3	日											
Reset Value	REAL	重新設定輸入值。										
Target	REAL	積算器目標值										
TargetDev1	REAL	積算器總量值與目標值之最大誤差值										
TargetDev2	REAL	積算器總量值與目標值之最小誤差值										
Start	BOOL	積算器啟動計算										
Stop	BOOL	積算器停止計算										
Reset	BOOL	積算器重置										
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)										
Enable In	BOOL	指令可運作執行輸出值。										
Total	REAL	指令計算後輸出值。										
Old TOT	REAL	前一次計算後輸出值。										
Target Flag	BOOL	$Total \geq Target$										
TargetDev1Flag	BOOL	$Total \geq Target - TargetDev1$										

TargetDev2Flag	BOOL	Total \geq Target - TargetDev2
----------------	------	----------------------------------

說明：

積算器計算公式：

$$\text{Total}_n = \text{Total}_{n-1} + \text{Gain} \times \frac{\text{Delta T}}{2 \times \text{Time Base}} \times (\text{In}_n + \text{In}_{n-1})$$

其 Time Base 定義如下列所示：

值	條件
1	Time Base = 0 (秒)
60	Time Base = 1 (分)
3600	Time Base = 2 (時)
86400	Time Base = 3 (日)

當 Reset Req=1 時則將會執行下列動作：

Old Total = Total

Total = Reset Value

若 Target 值將被達到後其作動方式如下：

當 Total \geq Target 時則 Target Flag =1

當 Total \geq (Target - TargetDev1) 時則 TargetDev1Flag=1

當 Total \geq (Target - TargetDev2) 時則 TargetDev2Flag=1

5.延遲器(Dead Time , DEDT)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
In Fault	BOOL	初始值訊息輸入值，若值為 1 時則 Out = Initial Value。
Delay Count	REAL	絕對延遲計數輸入值，若輸入值無效時會將該值設為 1，輸入範圍為 1~11，其預設值為 1。
Gain	REAL	控制過程增益倍數，輸入值為任一浮點數，其預設值為 1。
Bias	REAL	控制過程補償值，輸入值為任一浮點數，其預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當 Dead time 無效時，則 $Out = (In \times Gain) + Bias$

當 In Fault = 1 時則保持最後輸出值

6.過程指令驗證及操作說明：

- (i) 此範例用來測試 ALM，SCL，DEDT，TOT 指令。ALM 指令部分利用不同的輸入值及 Limit 值來驗證 ALM 指令的正確性。

SCL 及 DEDT 則以波形來表示驗測結果是否正確。TOT 指令則是利用不同的輸入，並觀察其輸出，來驗證 TOT 指令累加效果的正確性。

(ii) Alarm 指令操作說明：

在 ALM 畫頁，

(1) 預設=100, Hi Limit 預設=80, Low Limit 預設=40, LowLow Limit 預設=20, Dead Band=1。

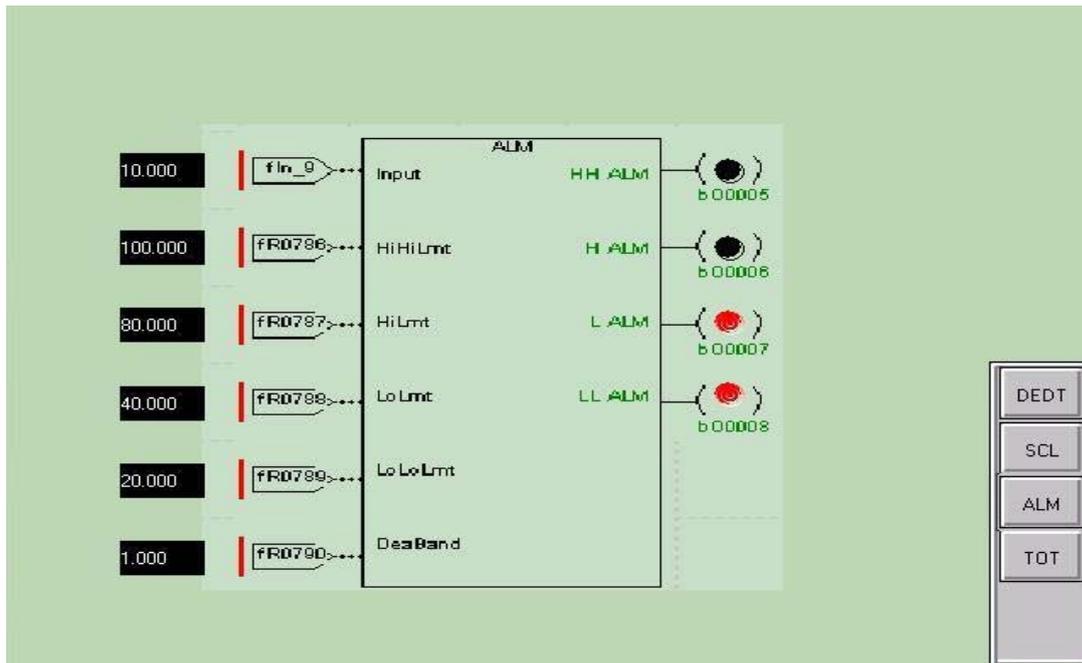
(2) 因此根據 ALARM 函式的說明，分別輸入 Input=99, 101, 110，則 HH ALM 及 HALM 為 TRUE。

(3) 分別輸入 Input = 79, 81, 90，則 HH ALM 為 FALSE, HALM 為 TRUE。

(4) 輸入 Input=50 ($41 \leq \text{Input} \leq 79$)，則所有 ALM 都為 FALSE

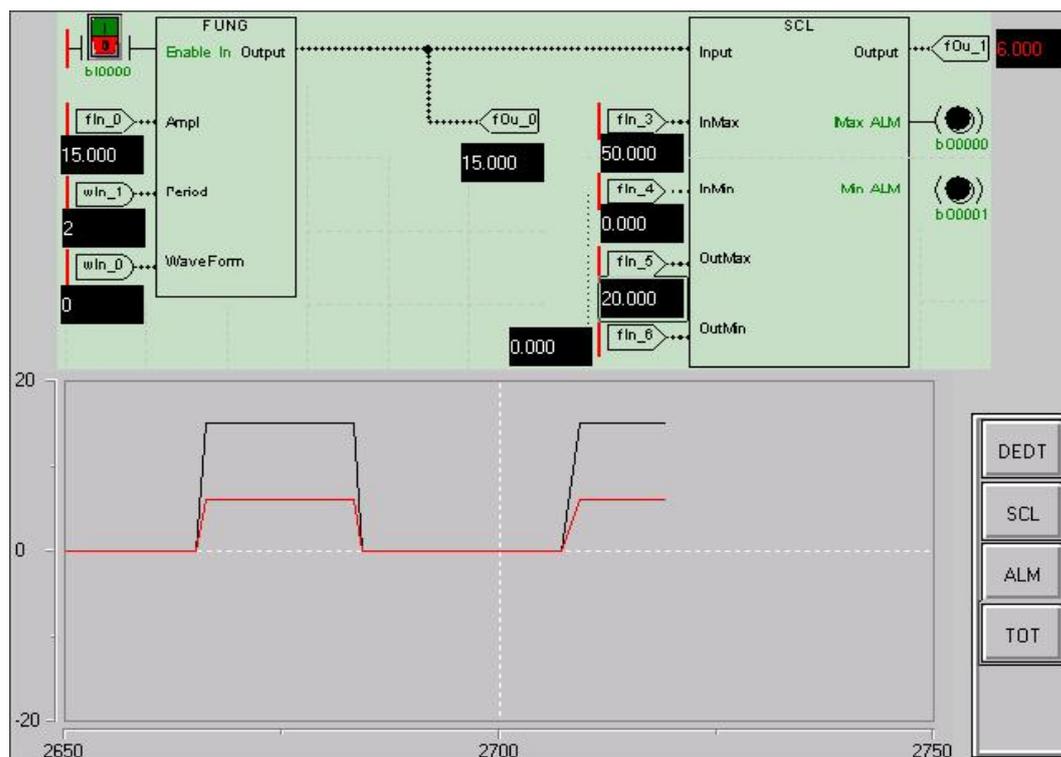
(5) 輸入 Input=39, 41, 33，則 L ALM 為 TRUE，LL ALM 為 FALSE

(6) 輸入 Input=19, 21, 17，則 L ALM 為 TRUE, LL ALM 為 TRUE



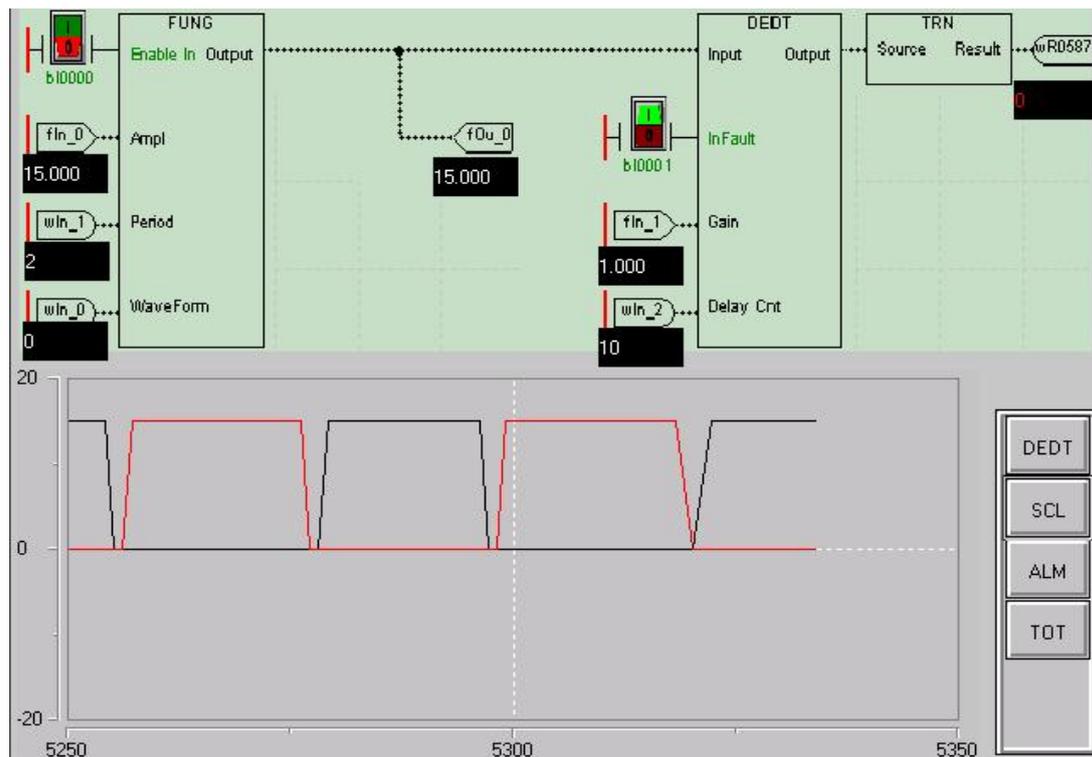
(iii) SCL 指令操作說明：

(1)在 SCL 畫面輸入 Input Max=50, Input Min=0, Output Max=20, Output Min=0, 使得根據 SCL 的計算公式可以得到輸出值為輸入值的 0.4 倍。如下圖，輸入波形為黑色方波，方波波峰值為 15。輸出值為紅色方波，波峰值為 6。



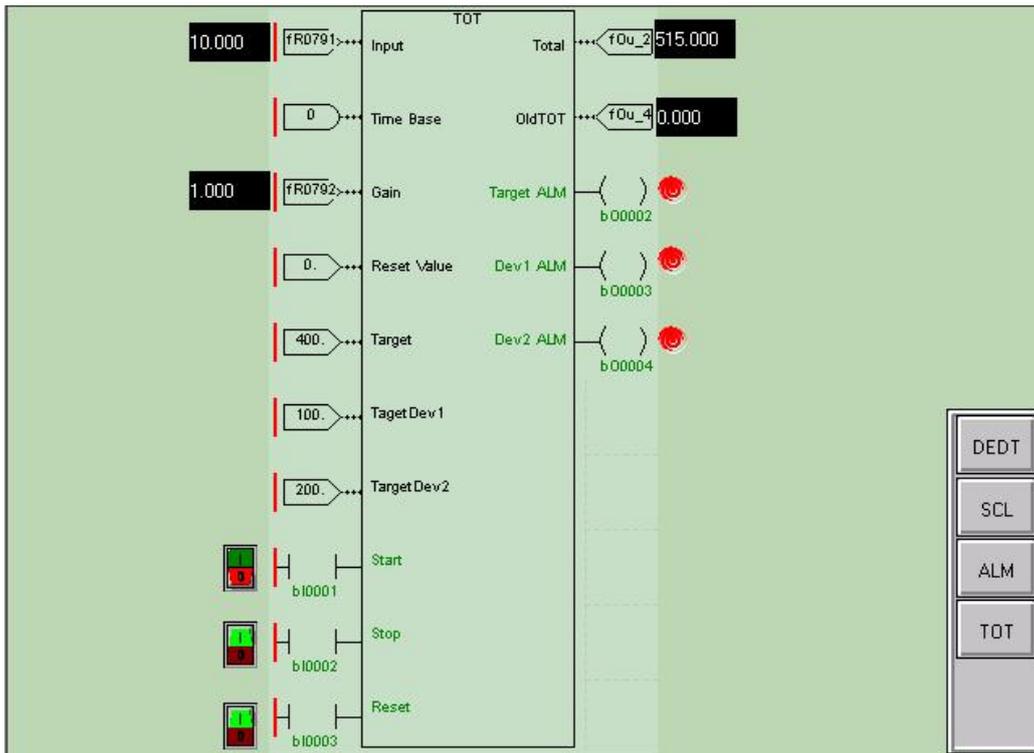
(iv) DEDT 指令操作說明

(1) 在 DEDT 畫頁，預設的 FUNG 產生波形為方波(黑色)，週期為 2 秒。因此 DEDT 函式的 Delay Cnt 中輸入 10，意為 Delay 1 秒，則可看見波形(紅色)較 FUNG 產生的波形延遲 1 秒。如下圖。



(iv) TOT 指令操作說明：

- (1) 輸入 Input=10，Gain=1。按下 Start，根據 TOT 的計算公式，可以得到 Total 值=5, 15, 25,遞增，並且當 Total \geq 200 時 Dev2 ALM 燈號亮起(Target-TargetDev2=400-200=200)。當 Total \geq 100 時 Dev1 ALM 燈亮起。當 Total \geq 400 時，Target ALM 燈號亮起。
- (2) 按下 Reset 鈕，則 Old TOT = Total 值。Total 值會被設定 =ResetValue，ResetValue 預設為 0。並且 Target ALM, Dev1 ALM, Dev2 ALM 燈號熄滅。



4.4.3 驅動指令(Drives Instructions)

驅動指令可分為下列指令：脈衝乘算器(Pulse Multiplier，PMUL)、上/下積算器(Up/Down Accumulator，UPDN) 等以及驅動指令驗證和操作說明

1. 脈衝乘算器(Pulse Multiplier，PMUL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	DINT	類比訊息輸入至指令，輸入值為任一整數值，其預設值為 0。
Initialize	BOOL	控制器初始值需求訊息，當值為 1 時；則 Out =0 以及內部暫存資料皆為

		0，其預設值為 0。
Initial Value	DINT	若 Initialize = 1 時，則 $Input_{n-1} = Initial Value$ ，輸入值為任一整數值，其預設值為 0。
Mode	BOOL	運作模式若值為 1 時則以相對模式 (Relative Mode) 計算輸出，反之以絕對模式 (Absolute Mode) 計算輸出，其預設值為 0。
Multiplier	DINT	倍數輸入值，其輸入值將被除以 100000 再與 Input 進行計算，其輸入值範圍為 -1000000~1000000。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

該指令屬於變頻器專用指令功能類似電子齒輪比，用法為將變頻器之 Encoder 信號於 “In” 腳位輸入，然後齒輪比的比數 “Multiplier” 腳位輸入，而輸出的值 $= In \times (Multiplier / 100,000)$ ；該指令又可分成兩種運作模式：

相對模式 (Relative Mode)

只輸出 Scan time 與 Scan time 之間的相對變化值，例如第 N 次 Scan 的值為 1000，第 N+1 次 Scan 的值為 1100，Multiplier 的值為 250000，則輸出值為 $(1100-1000) \times (250000/100000) = 250$ 。

絕對模式 (Absolute Mode)

該模式則直接將輸入值 “In” 乘上 $(Multiplier / 100000)$ 進行計算輸出。

2.上/下積算器(Up/Down Accumulator，UPDN)

結構及功能說明：

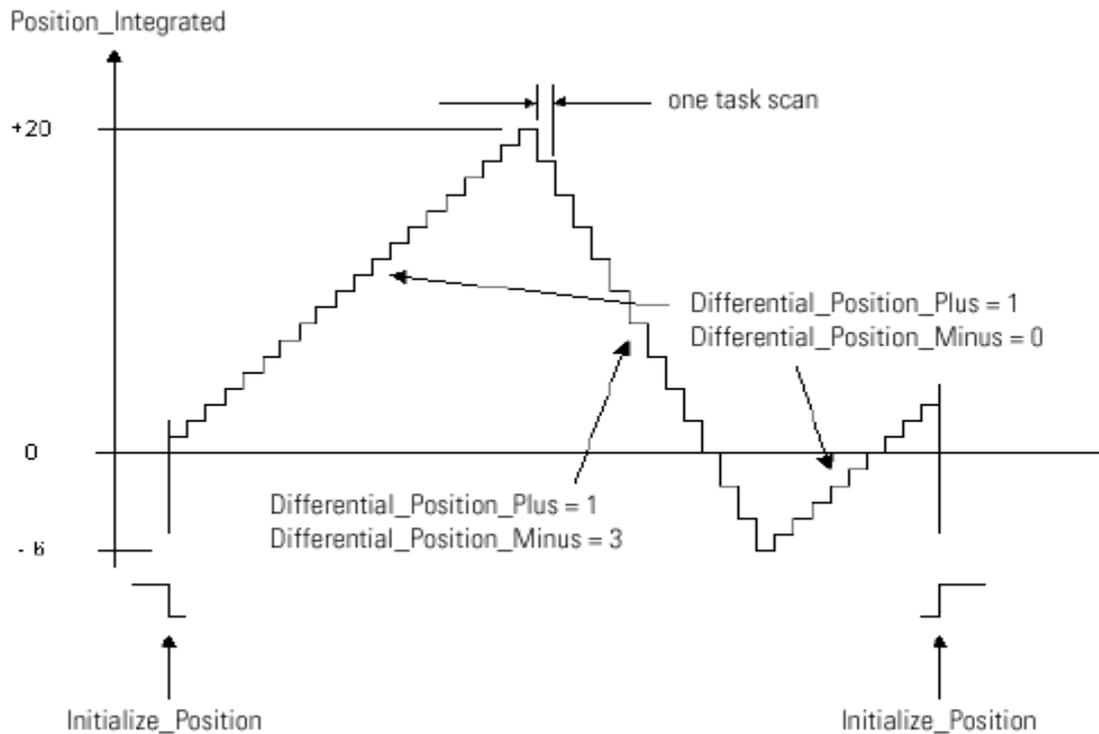
輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Initialize	BOOL	初始值訊息輸入值，若值為 1 時則 $Out = Initial Value$ 。
Initial Value	REAL	初始值
In Plus	REAL	輸入值將與積算器進行加法運算
In Minus	REAL	輸入值將與積算器進行減法運算
Hold	REAL	當 $Hold = 1$ 及 $Initialize=0$ 時則 Out 保持原值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Output	REAL	指令計算後輸出值。

說明：

UPDN 指令遵照下列公式條件進行運算：

條 件	動作說明
Hold=0 Initialize=0	$Sum Value_n = Sum Value_{n-1} + In Plus - In Minus$ $Out = Sum Value_n$
Hold=1 Initialize=0	$Sum Value_n = Sum Value_{n-1}$ $Out = Sum Value_n$
Initialize=1	$Sum Value_n = Initial Value$ $Out = Sum Value_n$

其指令運作執行之範例曲線圖如下圖所示



3. 驅動指令範例操作說明

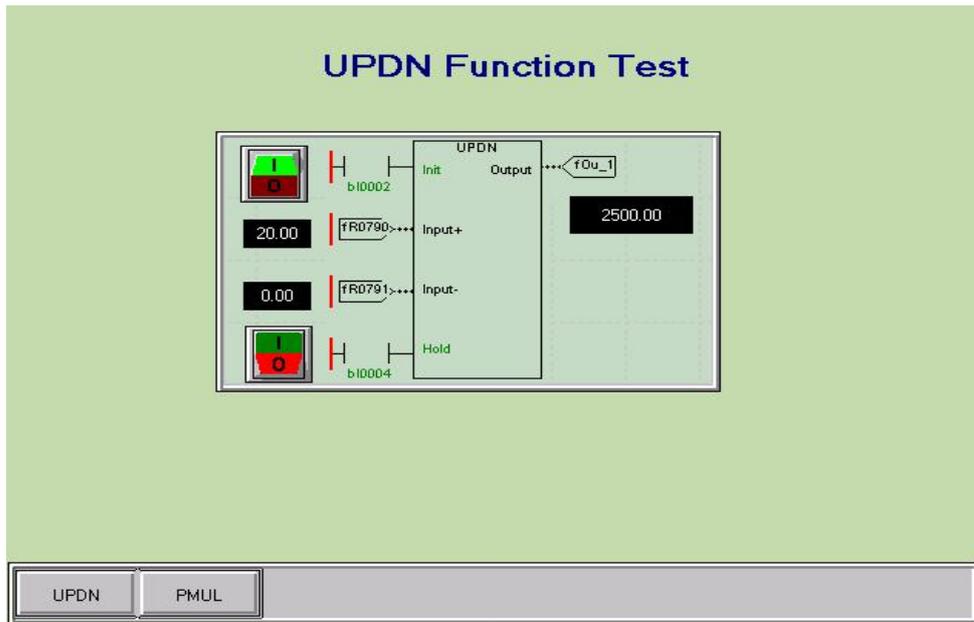
(i) 此範例使用不同的輸入值並觀察輸出值來驗證 UPDN 及 PMUL 指令的正確性。PMUL 指令部分同時可以直接從輸入與輸出的波形來判斷指令運作是否正確。

(ii) UPDN 指令操作說明：

(1) 輸入 "Input+" = 20, "Input-" = 0, 則根據 UPDN 的計算公式, Output 會以 20 等加遞增

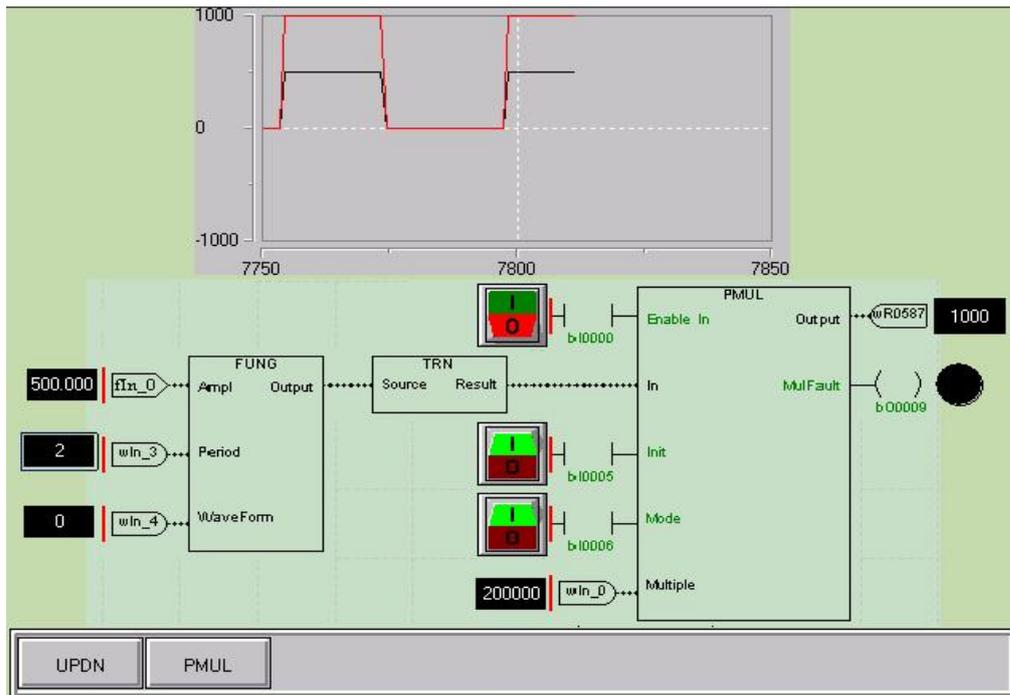
(2) 按下 Hold 鈕, 則 Output 不再遞增

(3) 輸入 "Input-" = 30, 則根據 UPDN 的計算公式, Output 會以 10 等減數遞減。



(iii) PMUL 指令操作說明：

- (1) 在 FUNG 函數中輸入 $Ampl=500$ ， $WaveForm=0$ 。用以產生振幅為 500 的方波(黑色線)
- (2) 在 PMUL 函數中輸入 $Multiple=200000$ ，並且按下 Enable In 按鈕
- (3) 根據 PMUL 公式，Output 數值為 Input 數值的 2 倍，產生波形(紅色線)與 Input 波形比較如下圖



4.4.4 選擇/限制指令 (Select/Limit Instructions)

選擇/限制指令可分為下列指令：選擇器 (Select, SEL)、進階選擇器 (Enhanced Select, ESEL)、負值輸出 (Selected Negate, SNEG)、多工器 (Multiplexer, MUX)、高低準位 (High/Low Limit, HLL)、比率限制器 (Rate Limiter, RLIM) 等以及選擇/限制指令驗證和操作說明。

1. 選擇器 (Select, SEL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1~ Input 2	REAL	類比訊息輸入至指令(1~2)，輸入值為任一浮點數，其預設值為 0。

Selector	BOOL	Input 1 和 Input 2 之輸入選擇器，預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

SEL 指令為以 Input 1 和 Input 2 為基礎之選擇器，當輸入選擇器為 1 時；則指令計算後輸出值為 Input 2 之數值。

2. 進階選擇器(Enhanced Select, ESEL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)												
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。												
Input 1~ Input 6	REAL	類比訊息輸入至指令(1~6)，輸入值為任一浮點數，其預設值為 0。												
Selector Mode	DINT	選擇器輸入型態 <table border="0"> <tr> <td>值</td> <td>說明</td> </tr> <tr> <td>0</td> <td>手動選擇</td> </tr> <tr> <td>1</td> <td>值為大者輸出</td> </tr> <tr> <td>2</td> <td>值為小者輸出</td> </tr> <tr> <td>3</td> <td>以中間值輸出</td> </tr> <tr> <td>4</td> <td>以平均值輸出</td> </tr> </table> 若輸入值非上列值時指令將無法更新輸出值，其輸入值範圍為 0~4，預	值	說明	0	手動選擇	1	值為大者輸出	2	值為小者輸出	3	以中間值輸出	4	以平均值輸出
值	說明													
0	手動選擇													
1	值為大者輸出													
2	值為小者輸出													
3	以中間值輸出													
4	以平均值輸出													

		設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當輸入值有 Input 1，Input 2，Input 3 時，其選擇器型態為 1 時，則指令計算後輸出值為三者中最大者之數值。



3. 負值輸出 (Selected Negate, SNEG)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一

		浮點數，其預設值為 0。
Negate Enable	BOOL	負值輸出執行輸入點，若輸入值為 1 時，其 Input 值前加一個負號，預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當 Negate Enable = 1 時；則 $Out = -In$ 。

當 Negate Enable = 0 時；則 $Out = In$ 。

4. 多工器 (Multiplexer, MUX)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1~ Input 8	REAL	類比訊息輸入至指令(1~8)，輸入值為任一浮點數，其預設值為 0。
Selector	DINT	輸入選擇器，若輸入值非 1~8 時指令將無法更新輸出值，其輸入值範圍為 1~8，預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

輸入值有 Input 1、Input 2、Input 3 時,若輸入選擇器為 2 時,則指令計算後輸出值為 Input 2 之數值。

5. 高低準位(High/Low Limit, HLL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)								
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。								
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。								
High Limit	REAL	輸入值之高限制值，若高限制值 \leq 低限制值時,則指令計算後輸出值為低限制值，正常輸入範圍：高限制值 $>$ 低限制值								
Low Limit	REAL	輸入值之低限制值，若高限制值 \leq 低限制值時,則指令計算後輸出值為低限制值，正常輸入範圍：低限制值 $<$ 高限制值。								
Select Limit	DINT	限制選擇器輸入型態 <table border="0" style="margin-left: 20px;"> <tr> <td>值</td> <td>說明</td> </tr> <tr> <td>0</td> <td>高低限皆使用</td> </tr> <tr> <td>1</td> <td>只使用高限</td> </tr> <tr> <td>2</td> <td>只使用低限</td> </tr> </table> 若輸入值非上列值時指令將無法更新輸出值，其輸入值範圍為 0~2，預設值為 0。	值	說明	0	高低限皆使用	1	只使用高限	2	只使用低限
值	說明									
0	高低限皆使用									
1	只使用高限									
2	只使用低限									
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)								

Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。
High Alarm	BOOL	高限制值警報，當輸入值 \geq 高限制值時輸出為 1。
Low Alarm	BOOL	低限制值警報，當輸入值 \leq 低限制值時輸出為 1。
Limit Fail	BOOL	當高限制值 \leq 低限制值時
Selector Fail	BOOL	限制選擇器輸入值不在 0，1 或 2 時

說明：

若限制選擇器型態為 0 時，則輸入值需在高低限制值內才會被輸出，若限制選擇器型為 1 時，則輸入值僅限制高限制值，若值 \geq 高限制值，則指令計算後輸出值為高限制值之數值。

6. 比率限制器(Rate Limiter, RLIM)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
Inc Rate	REAL	最大每秒遞增率輸入值，若輸入值無效時，則每秒遞增率輸入值將為 0，正常輸入範圍：任一浮點數 \geq 0.0，預計值為 0。

Dec Rate	REAL	最小每秒遞減率輸入值，若輸入值無效時，則每秒遞減率輸入值將為 0，正常輸入範圍：任一浮點數 ≥ 0.0 ，預計值為 0。
By Pass	BOOL	By Pass 輸入訊號，當值為 1 時；則指令計算後輸出值為 Input 之數值，預設值為 0。
Timing Mode	DINT	時間模式輸入型態 值 說明 0 以原程式執行時間來更新資料 1 以採樣時間來更新資料 若輸入值非上列值時指令將無法更新輸出值，其輸入值範圍為 0~1，預設值為 0。
Over sample DT	REAL	採樣時間輸入值
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。
Delta T	REAL	資料更新時間。

說明：

當 By Pass 為 1 時；則 $Out = Input, Out_{n-1} = Input$ 。

當 By Pass 為 0 時；則 $Slope = (In_n - Out_{n-1}) / Delta T$

如果 $Slope \leq -Dec Rate$ 則 $Y Slope = -Dec Rate$

如果 $-Dec Rate \leq Slope \leq Inc Rate$ 則 $Y Slope = Slope$

如果 $Inc Rate \leq Slope$ 則 $Y Slope = Inc Rate$

輸出值為 $Out_n = Out_{n-1} + Delta T \times Y Slope$

PS：此指令目前僅能持續使用 6 年。

7. 選擇/限制指令驗證和操作說明

(i) ESEL 指令操作說明

(1) 切換到 ESEL 畫頁，按下 Enable In 按鈕

(2) 輸入 Sel Mode=0 手動選擇模式。

(3) 輸入 In Used = 3，則 Output = Input 3 = 5

(4) 輸入 Sel Mode=1，比較輸入中的最大值，並將之設為輸出。

(5) 輸入 In Used = 5，比較 Input1~Input5 中的最大值並將之設為輸出。

則 Output = 6。

(6) 輸入 Sel Mode=2，比較輸入中的最小值，並將之設為輸出。

(7) 輸入 In Used=4，比較 Input1~Input4 中的最小值並將之設為輸出。

則 Output=1。

(8) 輸入 Sel Mode=3，輸出中間值。在此模式中 In Used 必須為奇數。

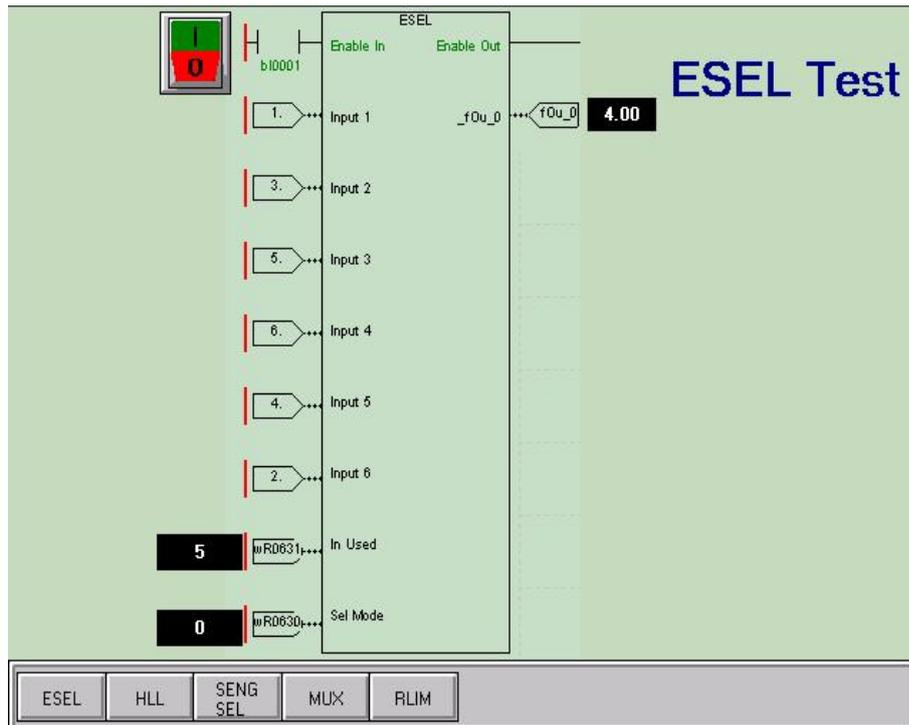
(9) 輸入 In Used=5，得到 Input1~Input5 中的中間值並設為 Output。則

Output=4。

(10) 輸入 Sel Mode=4，輸出平均值。

(11) 輸入 In Used=6，計算 Input1~Input6 的平均值，並設為 Output。

則 Output=3.5。



(ii) HLL 指令操作說明

(1) 切換到 HLL 畫頁，按下 Enable In 按鈕

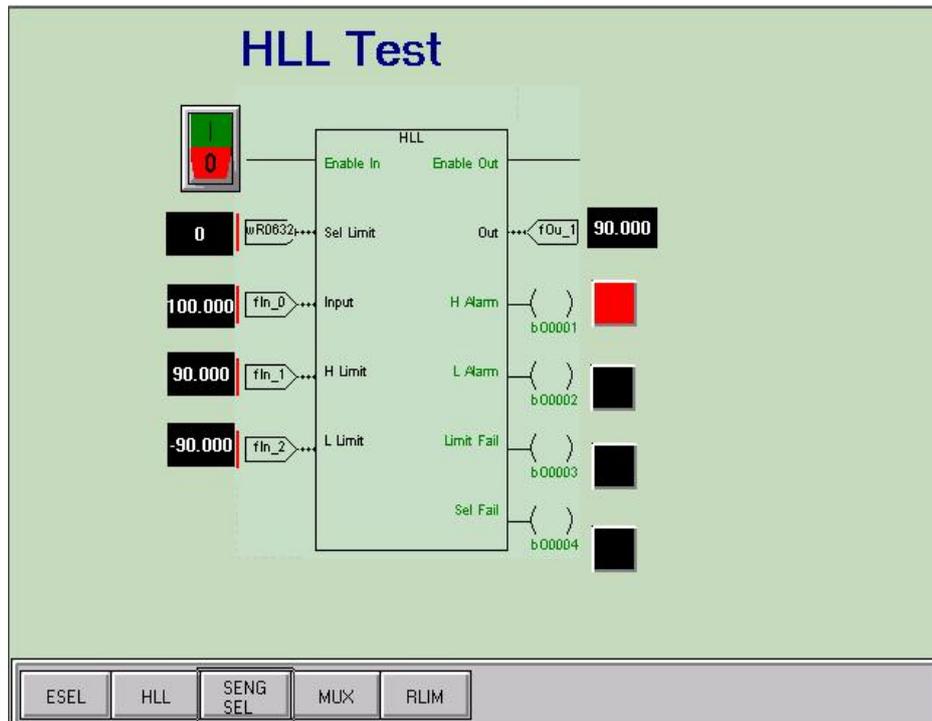
(2) 輸入 Sel Limit=0，同時使用 H Limit 及 L Limit

(3) 預設的 Input=100, H Limit=90, L Limit=-90。此時因為 Input > H Limit，所以 Output=H Limit=90。H Alarm 燈號亮

(4) 輸入 Input=80，則 H Alarm 燈號熄滅

(5) 輸入 Input=-110，則 L Alarm 燈號亮起。

(6) 輸入 Sel Limit=3，則 Sel Fail 燈號亮起。

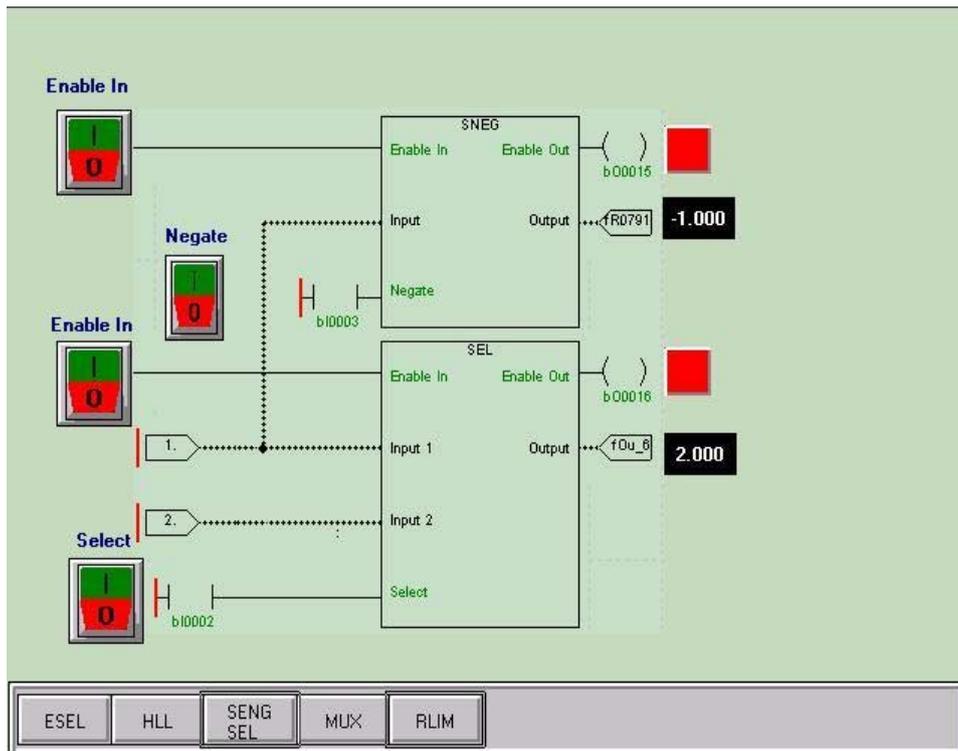


(iii) SNEG 及 SEL 指令操作說明

(1) 切換到 SNEG/SEL 畫頁，並按下 Enable In

(2) 按下 SNEG 函式中 Negate 鈕，則 Output=-1

(3) 按下 SEL 函式中的 Select 鈕，使得 Select=1，則 Output=2

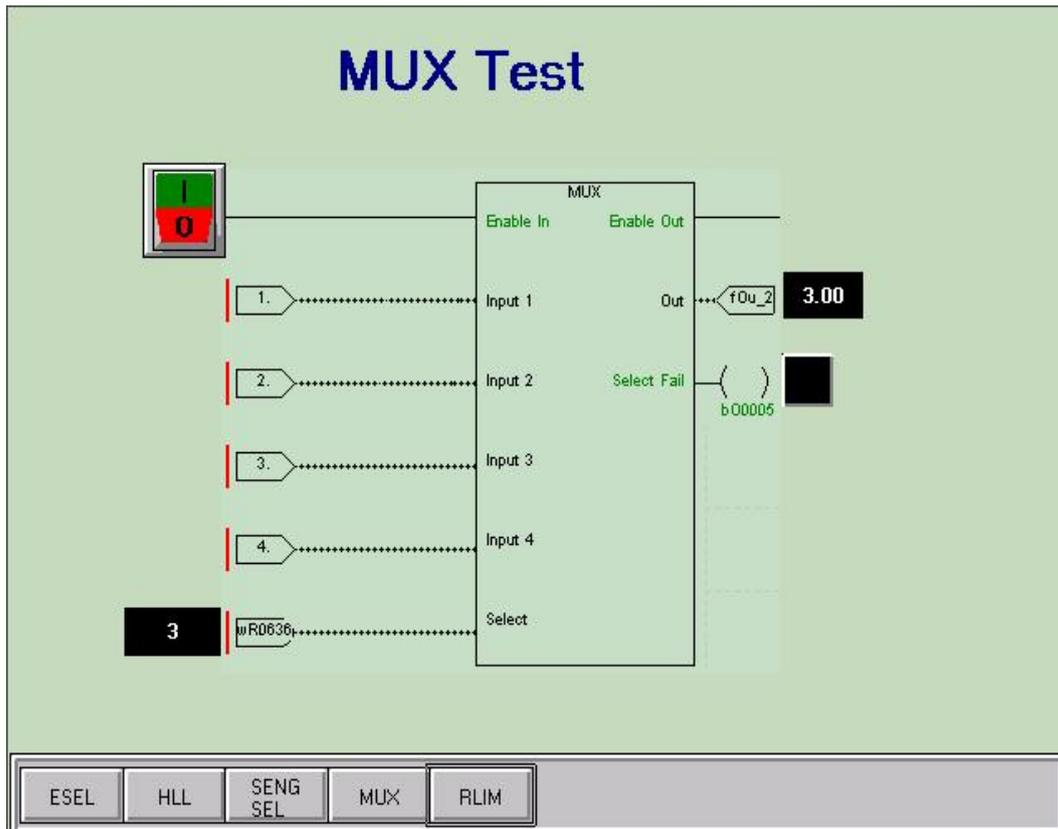


(iv) MUX 指令操作說明

(1) 切換到 MUX 畫頁，並且按下 Enable In 按鈕

(2) 輸入 Select=3，則 Output=Input 3=3

(3) 輸入 Select=9，則 Select Fail 燈亮

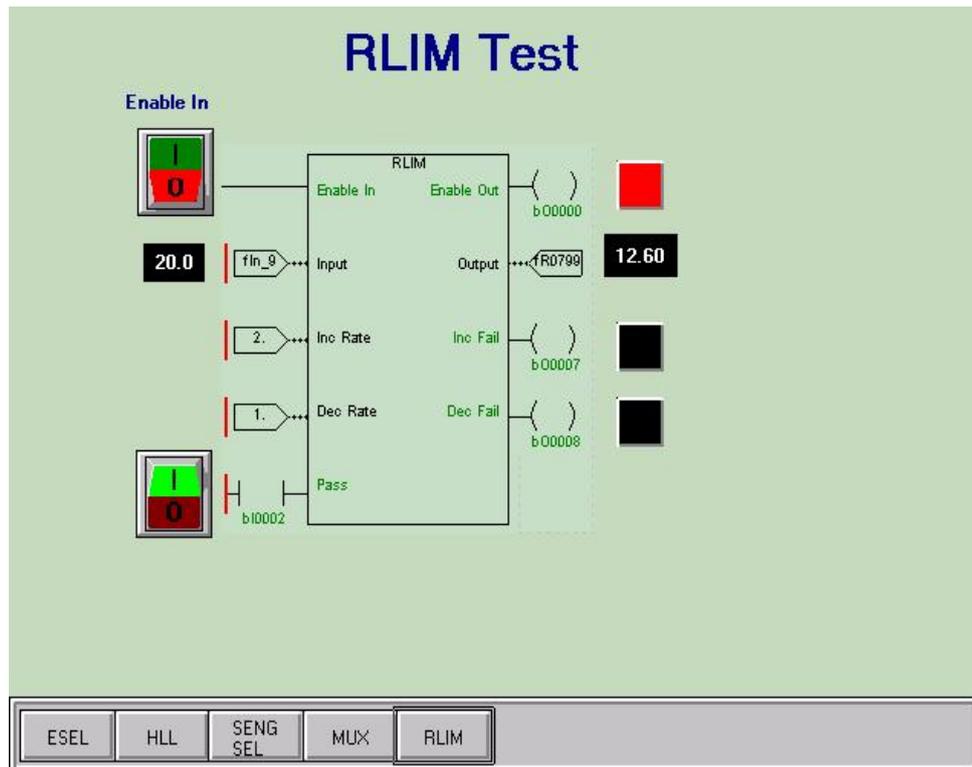


(v)RLIM 指令操作說明

(1)切換到 RLIM 畫頁，並且按下 Enable In

(2)輸入 Input=20。根據 RLIM 公式，因為 $20 / 0.1 > 2$ (Inc Rate)。因此 Output 值會以 $\text{Inc Rate} \times \text{Delta T} = 2 \times 0.1 = 0.2$ 等加遞增，直到 Output=Input 為止。

(3)當 Output=Input=20 時，輸入 Input=0。則此時根據 RLIM 公式，變化率為 $-20 / 0.1 < -1$ (Dec Rate)。因此 Output 值會以 $\text{Dec Rate} \times \text{Delta T} = 1 \times 0.1 = 0.1$ 等減數遞減，直到 Output=Input 為止。



4.4.5 統計指令(Statistical Instructions)

統計指令可分為下列指令：取最小值(Minimum Capture, MINC)、移動式平均值(Moving Average, MAVE)、移動式偏差值(Moving Standard Deviation, MSTD)、取最大值(Maximum Capture, MAXC)等以及統計指令驗證和操作說明。

1.取最小值(Minimum Capture , MINC)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
Reset	BOOL	若訊息為 1 時；將指令運算值重新設定為 Reset Value 值。
Reset Value	REAL	為一輸入值若 Reset 值一直為 1 時，則 Out=Reset Value 輸入值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當 Reset 為 1 時 $Out_{n-1} = \text{Reset Value}$
 $Out = \text{Reset Value}$

當 Reset 為 0 時 $Out = \text{Input}$ (當 $\text{Input} < Out_{n-1}$)

$Out = Out_{n-1}$ (當 $Out \geq Out_{n-1}$)

2.移動式平均值(Moving Average , MAVE)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
Initialize	BOOL	若訊息為 1 時；將指令運算值 Out 重新設定為 Input 值。
Sample Enable	BOOL	若訊息為 1 時；將進行 Input 值抓取計算
Sample Number	DINT	欲抓取計算之樣品數，若輸入值無效時將會出現 Sample Number Fail 以及指令運算輸出值保持原值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

MAVE為一重量組合或無重量組合之移動平均值指令，若Sample

Enable成立時將進行Input資料抓取運算而抓取樣品數以Sample

Number值來決定。

組合運算公式：

$$Out = \frac{\sum_{n=1}^{SampleNumber} In_n}{SampleNumber}$$

範例

當 Sample Number =3

Scan 1: $Out = In_n / 1$

Scan 2: $Out = (In_n + In_{n-1}) / 2$

Scan 3: $Out = (In_n + In_{n-1} + In_{n-2}) / Sample\ Number$

3. 移動式偏差值(Moving Standard Deviation, MSTD)

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
Initialize	BOOL	若訊息為 1 時；將指令運算值 Out 重新設定為 Input 值。
Sample Enable	BOOL	若訊息為 1 時；將進行 Input 值抓取計算
Sample Number	DINT	欲抓取計算之樣品數，若輸入值無效時將會出現 Sample Number Fail 以及指令運算輸出值保持原值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Average	REAL	指令計算後輸出平均值。
SDev	REAL	指令計算後輸出值。

說明：

Condition Action

Average

$$Average = \frac{\sum_{n=1}^{NumberOfSamples} In_n}{NumberOfSamples}$$

Out

$$Out = \sqrt{\frac{\sum_{n=1}^{NumberOfSamples} (In_n - Average)^2}{NumberOfSamples}}$$

4.取最大值(Maximum Capture , MAXC)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	REAL	類比訊息輸入至指令，輸入值為任一浮點數，其預設值為 0。
Reset	BOOL	若訊息為 1 時；將指令運算值重新設定為 Reset Value 值。
Reset Value	REAL	為一輸入值，若 Reset 值一直為 1 時，則 Out=Reset Value 輸入值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當 Reset 為 1 時 $Out_{n-1} = Reset\ Value$

$Out = Reset\ Value$

當 Reset 為 0 時 $Out = Input$ (當 $Input > Out_{n-1}$)

$Out = Out_{n-1}$ (當 $Out \leq Out_{n-1}$)

$Out_{n-1} = Out$

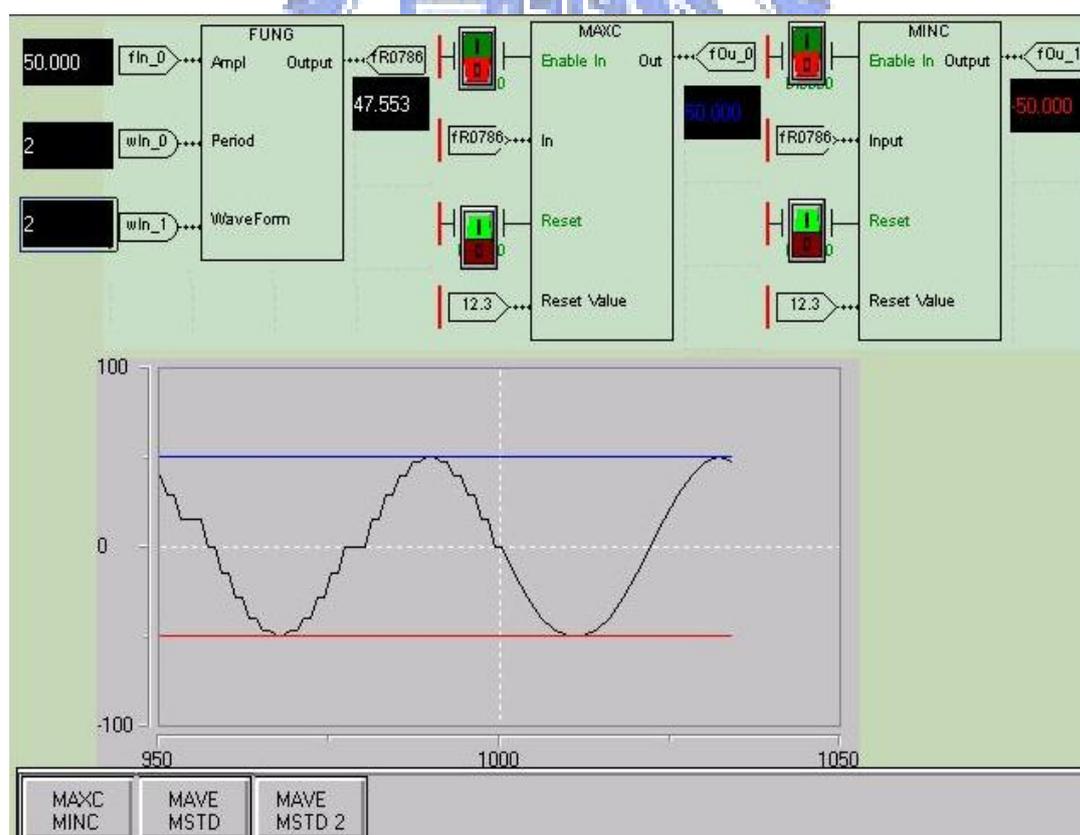
5. 統計指令範例操作說明

(i) 此範例是使用正弦波作為輸入訊號源，並觀察各指令的輸出波形，來驗證指令是否正確。

(ii) MINC, MAXC 操作指令說明

(1) 切換到 MAXC/MINC 畫頁，在 FUNG 函式中輸入 $Ampl=50$, $Period=2$, $Wave Form=2$ ，則輸出為週期是 2，振幅為 50 的正弦波(黑色線)。

(2) 由下圖可以看到最大值(藍色線)是 50，最小值(紅色線)是 -50



(iii) MAVE / MSTD 指令操作說明

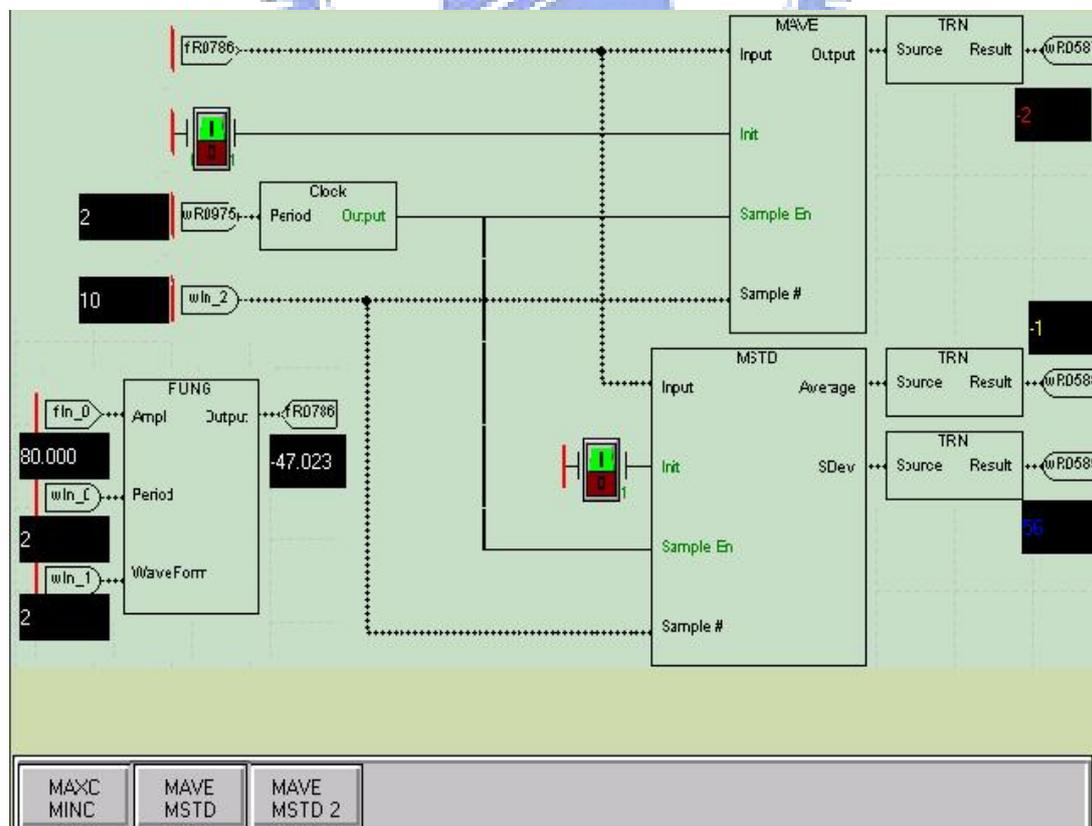
(1) 切換到 MAVE / MSTD 畫頁

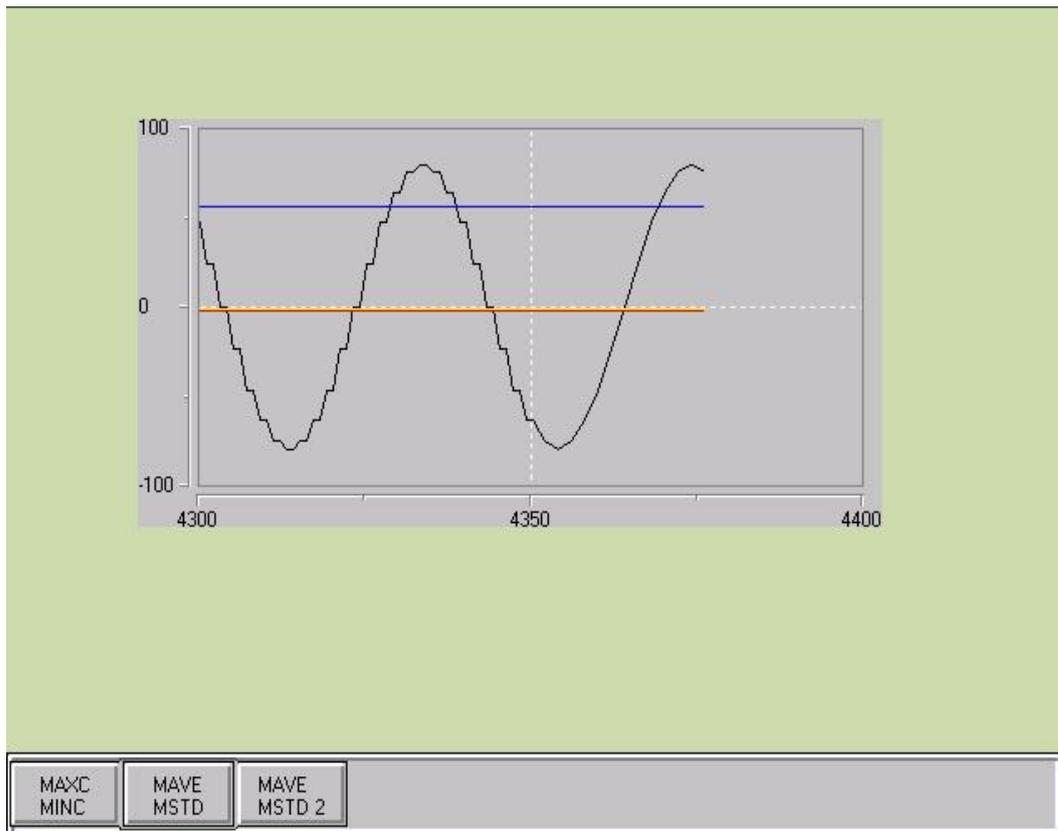
(2) 在 FUNG 函式中輸入 FUNG=80，Period=2，WaveForm=2，輸出週期是 2 秒，振幅為 80 的正弦波

(3) 在 Clock 函式的 Period 輸入 2，表示每 200ms 輸出 ON/OFF 一次

(4) 在 MAVE 及 MSTD 共用的 Sample # 欄位輸入 10，表示每 10 個 Input 點做一次 MAVE 及 MSTD 的計算

(5) 切換到 MAVE / MSTD 2 畫頁可以看到黑色正弦波為輸入訊號，紅色及黃色線分為為 MAVE 及 MSTD 所計算的平均值應為 0 或接近 0 (因為跟取樣到的點有關係)。藍色線為 MSTD 所計算出來的標準差，值為 56





4.4.6 位元觸發指令(Bit Instructions)

位元觸發指令可分為下列指令：正緣觸發(One Shot Rising with Input, OSRI)、負緣觸發(One Shot Falling with Input, OSFI)等指令以及位元觸發指令驗證和操作說明。

1. 正緣觸發(One Shot Rising with Input, OSRI)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。

Input bit	BOOL	數位訊息輸入至指令，輸入值為布林值，其預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Output bit	BOOL	指令計算後輸出值。

說明：

當 Input bit = 1 及 Input bit_{n-1} = 0 時 則 Output bit 輸出為 1

當 Input bit = 0 及 Input bit_{n-1} = 1 時 則 Output bit 輸出為 0

2. 負緣觸發(One Shot Falling with Input, OSFI)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input bit	BOOL	數位訊息輸入至指令，輸入值為布林值，其預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Output bit	BOOL	指令計算後輸出值。

說明：

當 Input bit = 0 及 Input bit_{n-1} = 1 時 則 Output bit 輸出為 1

當 Input bit = 1 及 Input bit_{n-1} = 0 時 則 Output bit 輸出為 0

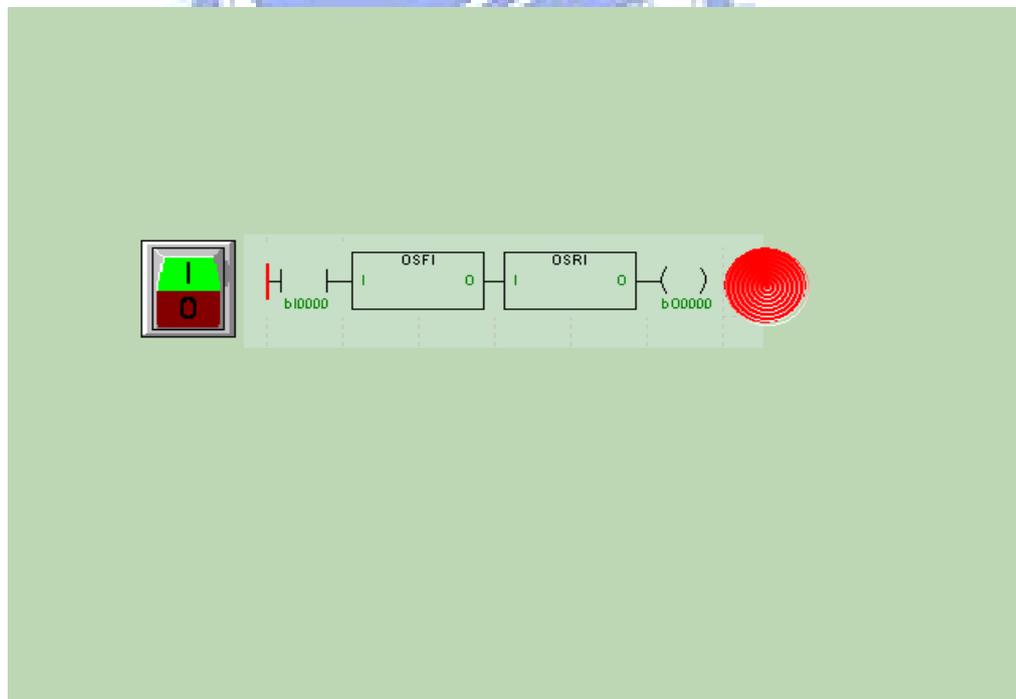
3.位元觸發指令驗證和操作說明

(i)此範例是使用 OSFI 連接 OSRI 觀察輸入與輸出值，同時驗證兩個指令是否正確。

(ii)OSFI 及 OSRI 函式操作說明

(1)按下 OSFI 函數的 I，使按鈕的狀態為 ON。此時 OSFI 函數不會作動，OSFI 的 O 為 FALSE，因此 OSRI 函數也不會作動。

(2)再按下 OSFI 函數的 I，使按鈕狀態為 OFF。此時 OSFI 函式作動，O 為 TRUE。因此 OSRI 函數也作動，O 為 TRUE。此時 bO0000 燈亮。



4.4.7 計時/計數指令(Timer/Counter Instructions)

計時/計數指令可分為下列指令：正向延遲計時重置器(Timer On with Reset, TONR)、負向計時重置器(Timer Off with Reset, TOFR)、自保持計時重置器(Retentive Timer On with Reset, RTOR)、上/下計數器(Count Up/Down, CTUD)

等指令以及各指令驗證和操作說明。

1.正向延遲計時重置器(Timer On with Reset, TONR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Timer Enable	BOOL	輸入值為布林值，若值為 1 時表示 Timer 運作計時，其預設值為 0。
Preset	DINT	Timer 預計時之時間；若輸入值無效時 Timer 將無法運作。
Reset	BOOL	為一布林值，若值為 1 時，則 Timer 計時將被重置歸零，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Value	DINT	Timer 計時值。
Timer On	BOOL	Timer 運轉狀態，值為 1 時表示 Timer 計時中，值為 0 時表示 Timer 已停止計時。

Done	BOOL	Timer 計時到達 Preset 值時，輸出值 =1。
------	------	------------------------------

說明：

當 Timer Enable =1 時 Timer On = 1；ACC 值開始計時，直到 ACC 值到達 Preset 設定值($ACC \geq Preset$)時；則 Done 輸出為 1。

當 Reset 值=1 時，則 ACC=0,Done=0。

當 Reset 值=0 時，則 ACC 重新計時。

2.負向計時重置器(Timer Off with Reset，TOFR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Timer Enable	BOOL	輸入值為布林值，若值為 1 時表示 Timer 停止運作計時，其預設值為 0。
Preset	DINT	Timer 預計時之時間；若輸入值無效時 Timer 將無法運作。
Reset	BOOL	為一布林值，若值為 1 時，則 Timer 計時將被重置歸零，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Value	DINT	Timer 計時值。
Timer On	BOOL	Timer 運轉狀態，值為 0 時表示 Timer 計時中，值為 1 時表示 Timer 已停止計時。

Done	BOOL	Timer 計時到達 Preset 值時，輸出值 =0。Timer 重置後其輸出值=1。
------	------	--

說明：

當 Timer Enable =0 時 Timer On = 0；ACC 值開始計時，直到 ACC 值到達 Preset 設定值($ACC \geq Preset$)時；則 Done = 0 (指令運作時 Done = 1)。

當 Reset 值=1 時，則 ACC=0,Done=1。

當 Reset 值=0 時，則 ACC 重新計時。

3.自保持計時重置器(Retentive Timer On with Reset，RTOR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Timer Enable	BOOL	輸入值為布林值，若值為 1 時表示 Timer 運作計時，其預設值為 0。
Preset	DINT	Timer 預計時之時間；若輸入值無效時 Timer 將無法運作。
Reset	BOOL	為一布林值，若值為 1 時，則 Timer 計時將被重置歸零，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Value	DINT	Timer 計時值。
Timer On	BOOL	Timer 運轉狀態，值為 1 時表示 Timer 計時中，值為 0 時表示 Timer 已停止計時。

Done	BOOL	Timer 計時到達 Preset 值時，輸出值=1。Timer 重置後其輸出值=0。

說明：

當 Timer Enable =1 時 Timer On = 1，ACC 值開始計時，若 Timer Enable =0 時 Timer On = 0，ACC 值會保持最後計時值，再將 Timer Enable =1 時 Timer 計時會從前一次計時值開始計時，直到 ACC 值到達 Preset 設定值($ACC \geq Preset$)時；則 Done 輸出為 1。

當 Reset 值=1 時，則 ACC=0,Done=0。

當 Reset 值=0 時，則 ACC 重新計時。

4.上/下計數器(Count Up/Down，CTUD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Up Enable	BOOL	上計數啟動，若值為 1 時表示上計數器運作計數，其預設值為 0。
Dn Enable	BOOL	下計數啟動，若值為 1 時表示下計數器運作計數，其預設值為 0。
Preset	DINT	Counter 欲計數之次數，有效值為任一整數其預設值為 0。
Reset	BOOL	為一布林值，若值為 1 時，則 Counter 計數值將被重置歸零，預設值為 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)

Enable In	BOOL	指令可運作執行輸出值。
ACC	DINT	Counter 計數值。
Counter On	BOOL	Counter 運轉狀態，值為 1 時表示 Counter 計數中，值為 0 時表示 Counter 已停止計數。
Done	BOOL	Counter 計數到達 Preset 值時，輸出值=1。Counter 重置後其輸出值=0。

說明：

當 Up Enable =1 時 CU On = 1；ACC 值開始計數，直到 ACC 值到達 Preset 設定值($ACC \geq \text{Preset}$)時；則 Done =1 (指令運作時 Done = 0)。

當 Reset 值=1 時，則 ACC=0,Done=0。

當 Reset 值=0 時，則 ACC 重新計數。



5.計時/計數指令驗證和操作說明

(i)TONR/TOFR 指令操作說明

(1)切換到 TONR 畫面，按下 Enable In

(2)設定 Preset 值為 100

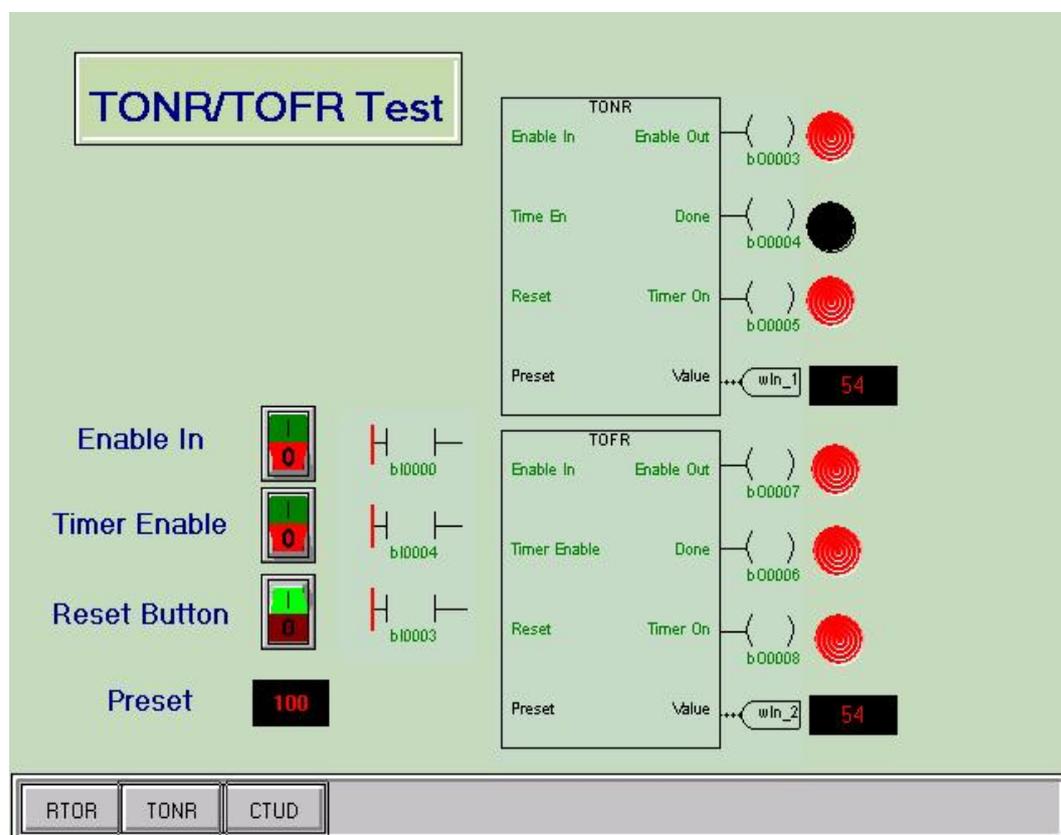
(3)按下 Timer Enable，此時 TONR, TOFR 的 Value 時開始遞增。在 Value 未達到目標值 100 前，TONR, TOFR 的 Timer On 燈都亮起。

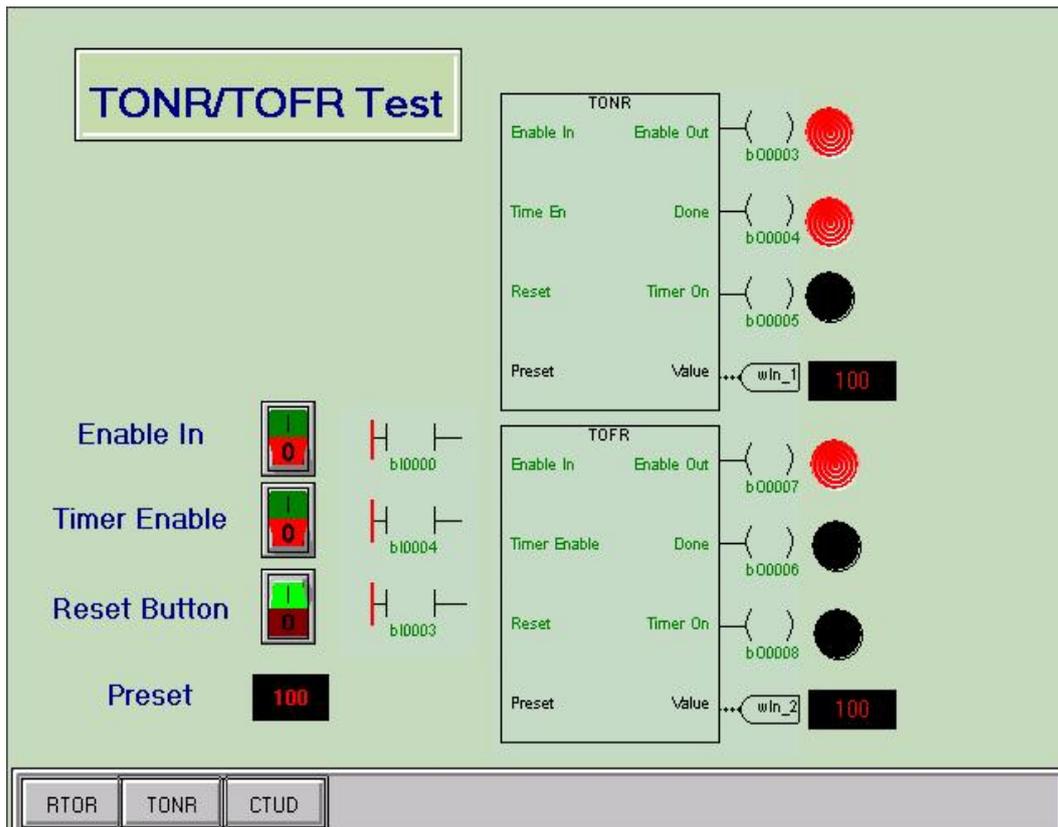
TONR 的 Done 燈號維持不亮，TOFR 的 Done 燈號亮起。

(4)當 Value 值達到 100 時,TONR, TOFR 的 Timer On 燈號熄滅,TONR 的 Done 燈號亮起,TOFR 的 Done 燈號熄滅

(5)按下 Reset 鈕,則 TONR, TOFR 的 Value 值歸 0

(6)再按下 Reset 鈕使 Reset 為 OFF 狀態,則 TONR, TOFR 的 Value 再度開始計數





(ii) RTOR 指令操作說明

(1) 切換到 RTOR 畫頁，並且按下 Enable In 按鈕

(2) 輸入 Preset=100，並且按下 Timer Enable 按鈕，使得 Timer Enable 按鈕狀態為 ON

(3) 此時 Timer On 燈號亮起，Value 數值開始遞增

(4) 在 Value 未達到 100 前，按下 Timer Enable 按鈕，使 Timer Enable 按鈕的狀態為 OFF。

(5) 此時 Value 數值不會歸 0，會維持目前的數值。

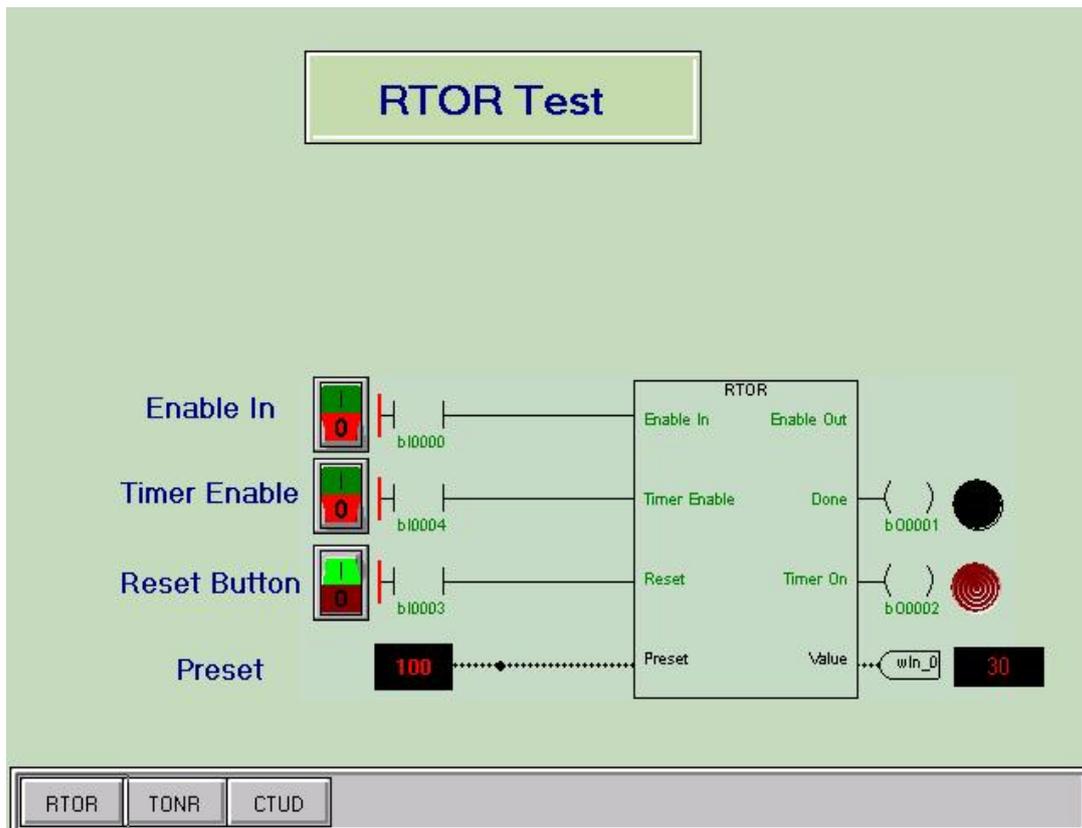
(6) 再按下 Timer Enable 按鈕，使 Timer Enable 按鈕的狀態為 ON

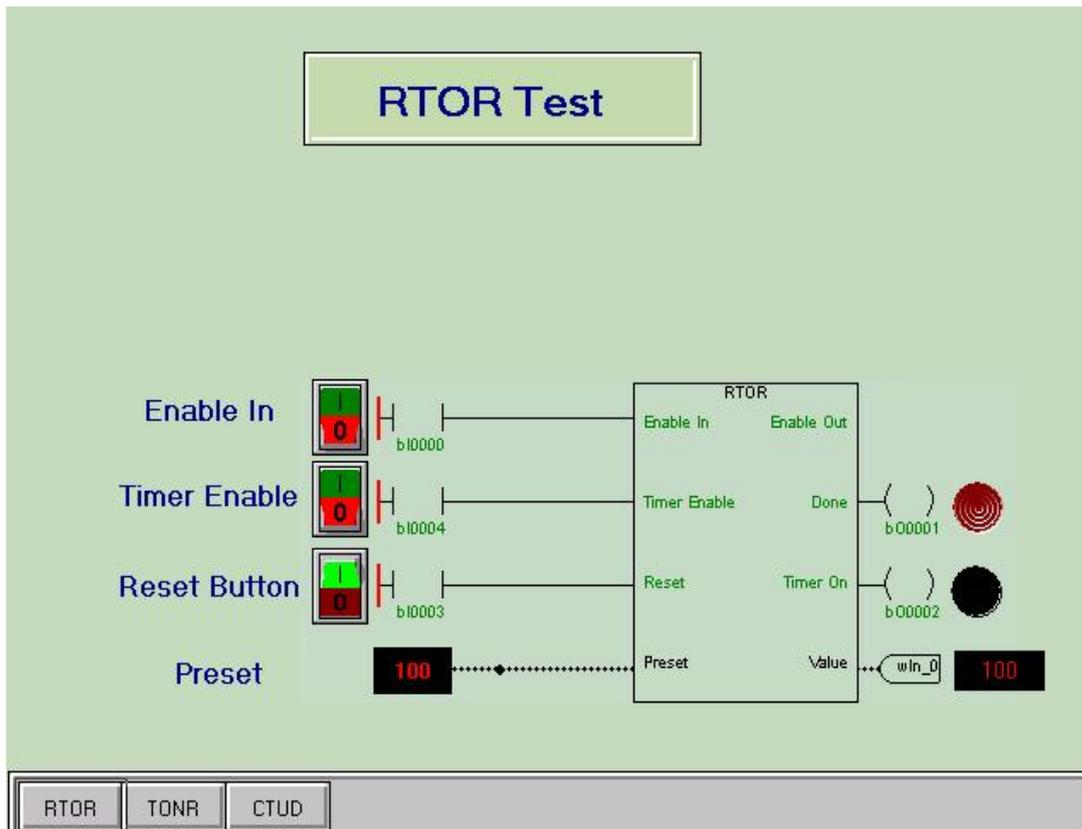
(7)此時 Value 的數值會從目前的數值開始繼續遞增，直到目標值 100 為止。

(8)當 Value 時達到 100 時，Timer On 燈號熄滅，Done 燈號亮起。

(9)按下 Reset 按鈕，此時 Value=0，Done 燈號熄滅

(10)再按下 Reset 按鈕，使按鈕狀態為 OFF，則此時 Value 又開始遞。





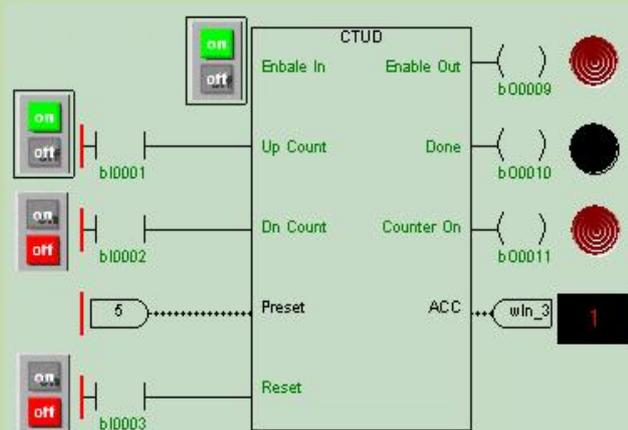
(iii) CTUD 指令操作說明

(1) 切換到 CTUD 畫面，並按下 Enable In 按鈕，使得 Enable In 按鈕的狀態為 ON。在此函式中 Up Count 與 Dn Count，在未達到 Preset 值之前不可切換使用。意即，同一時間只能使用 Up Count 或是 Dn Count。若要切換使用，必須先按下 Reset 鈕。

(2) 按 Up Count 按鈕，使 Up Count 按鈕 ON, OFF 一次。此時 ACC 數值為 1。此時 Counter On 燈號亮起。

(3) 再 Up Count 按鈕，使 Up Count 按鈕 ON, OFF 一次。此時 ACC 數值為 2。每 ON, OFF 一次 Up Count 按鈕，ACC 數值就加 1，直到 Preset 值為止。此 Sample 中 Preset 值為 5。

CTUD Test



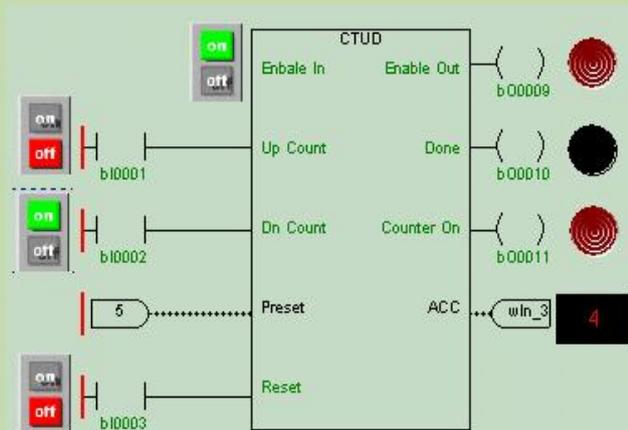
RTOR

TONR

CTUD

- (4)當 ACC 值為 5 時，Done 燈號亮起，Counter On 燈號熄滅。
- (5)當 ACC 值為 5 時，使 Up Count 按鈕狀態為 OFF。此時按下 Reset 按鈕，則 ACC 值歸 0。Done 燈號熄滅。
- (6)使 Reset 鈕為 OFF 狀態。按下 Dn Count 按鈕，使該按鈕 ON, OFF 一次。此時 Counter On 燈號亮起，ACC 數值會從 Preset 值下向減 1。也就是 $ACC=5-1=4$ 。

CTUD Test



RTOR

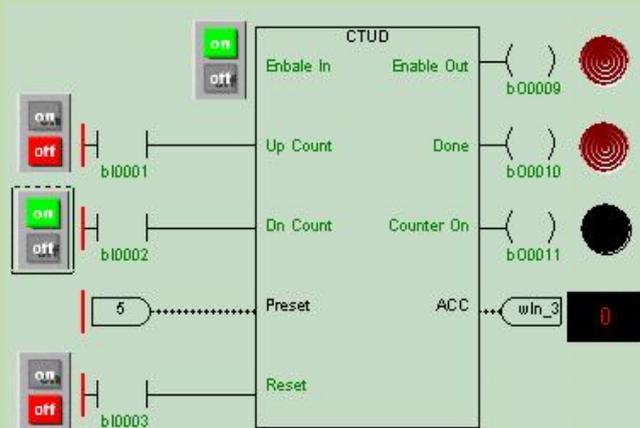
TONR

CTUD

(7)每 ON, OFF 一次 Dn Count 按鈕，則 ACC 值減 1。直到 ACC=0 為止。

(8)當 ACC = 0 時，Done 燈號亮起，Counter On 燈號熄滅。

CTUD Test

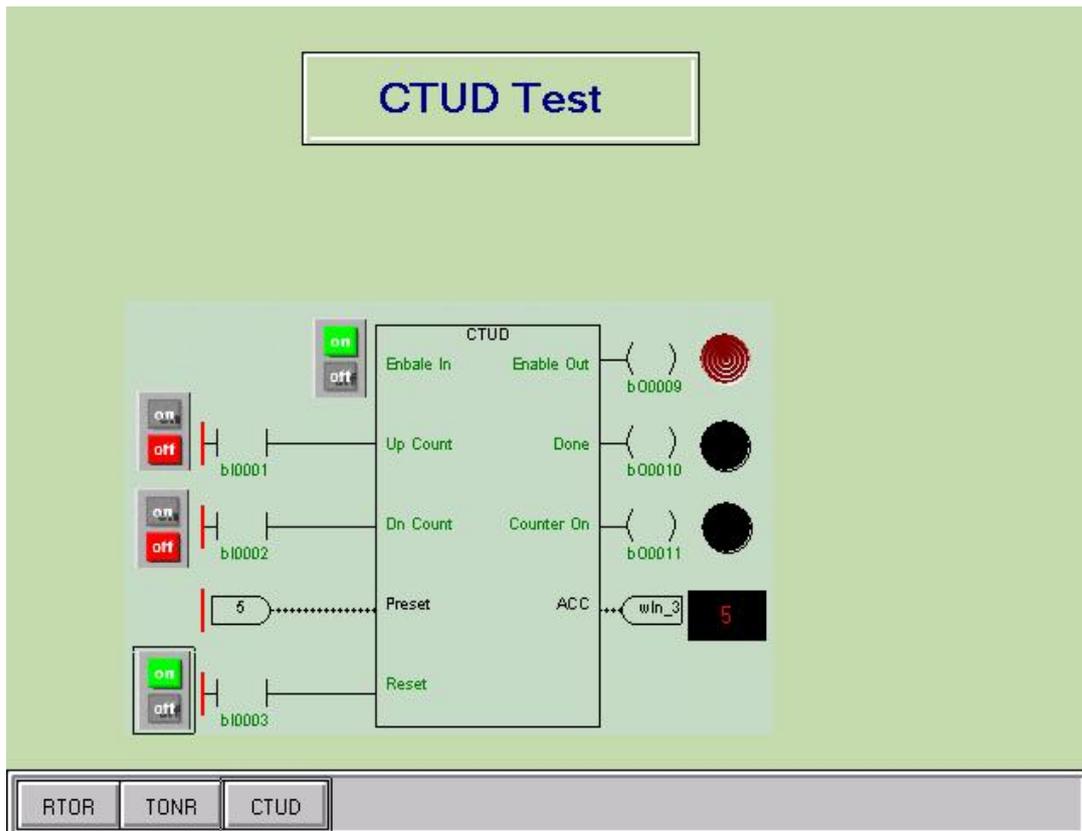


RTOR

TONR

CTUD

(9)使 Dn Count 按鈕為 OFF 狀態，並按下 Reset 按鈕。則 ACC 值 =Preset=5。



(10)使 Reset 按鈕為 OFF 狀態。

(11)測試操作完成。

4.4.8 比較指令(Compare Instructions)

比較指令可分為下列指令：限制(Limit ，LIM)、遮罩比對(Mask Equal to ，MEQ)、等於(Equal to ，EQU)、不等於(Not Equal to ，NEQ)、大於或等於(Greater than or Equal to ，GEQ)、大於(Greater Than ，GRT)、小於或等於(Less Than or Equal to ，LEQ)、小於(Less Than ，LES)等指令以及各指令驗證和操作說明。

1.限制(Limit ，LIM)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	欲進行計算之輸入值，輸入值為任一浮點數。
Low Limit	REAL	低限輸入值，輸入值為任一浮點數。
High Limit	REAL	高限輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

條件 1： $Low\ Limit \leq High\ Limit$

當 $0 \leq value \leq 100$ 時，則 $Result=1$ 。

若 $value < 0$ 或 $value > 100$ 時，則 $Result=0$ 。

條件 2： $Low\ Limit \geq High\ Limit$

當 $value \geq 0$ 或 $value \leq -100$ 時，則 $Result=1$ 。

若 $value < 0$ 或 $value > -100$ 時，則 $Result=0$ 。

2. 遮罩比對(Mask Equal to ，MEQ)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1

		時則指令可執行運作，預設值為 1。
Source	DWORD	欲進行比對計算之輸入值，輸入值為任一整數。
Mask	DWORD	定義單位位元值是否進行比對動作，當值為 1 時進行 Source 所對應之單位位元值將與 Compare 對應位置值進行比對，當值為 0 時則不執行，輸入值為任一整數。
Compare	DWORD	比對輸入值，將與 Source 值進行單位位元值比對，輸入值為任一整數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

首先計算出 Mask1 值

Source	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1						
Mask	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
Mask1	0	1	0	1	x	x	x	x	0	1	0	1	x	x	x	x	0	1	0	1	x	x	x	x	0	1	0	1

再計算出 Mask2 值

Compare	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1
Mask	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
Mask2	0	1	0	1	x	x	x	x	0	1	0	1	x	x	x	x	0	1	0	1	x	x	x	x	0	1	0	1

進行 Mask1 與 Mask2 紅框內之單位位元值比對，

若相同時，則 Result=1；反之則 Result=0。

3. 等於(Equal to，EQU)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

當 Source A 值等於 Source B 值時，則 Result = 1。

當 Source A 值不等於 Source B 值時，則 Result = 0。

4. 不等於(Not Equal to，NEQ)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)

Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

當 Source A 值不等於 Source B 值時，則 Result = 1。

反之則 Result = 0。

5.大於或等於(Greater than or Equal to，GEQ)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

當 Source A 值大於或等於 Source B 值時，則 Result = 1。

反之則 Result = 0。

6.大於(Greater Than，GRT)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

當 Source A 值大於 Source B 值時，則 Result = 1。

反之則 Result = 0。

7. 小於或等於(Less Than or Equal to，LEQ)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。

Result	BOOL	指令計算後輸出值。
--------	------	-----------

說明：

當 Source A 值小於或等於 Source B 值時，則 Result = 1。

反之則 Result = 0。

8.小於(Less Than ，LES)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
Source B	REAL	欲進行比較計算之輸入值，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	BOOL	指令計算後輸出值。

說明：

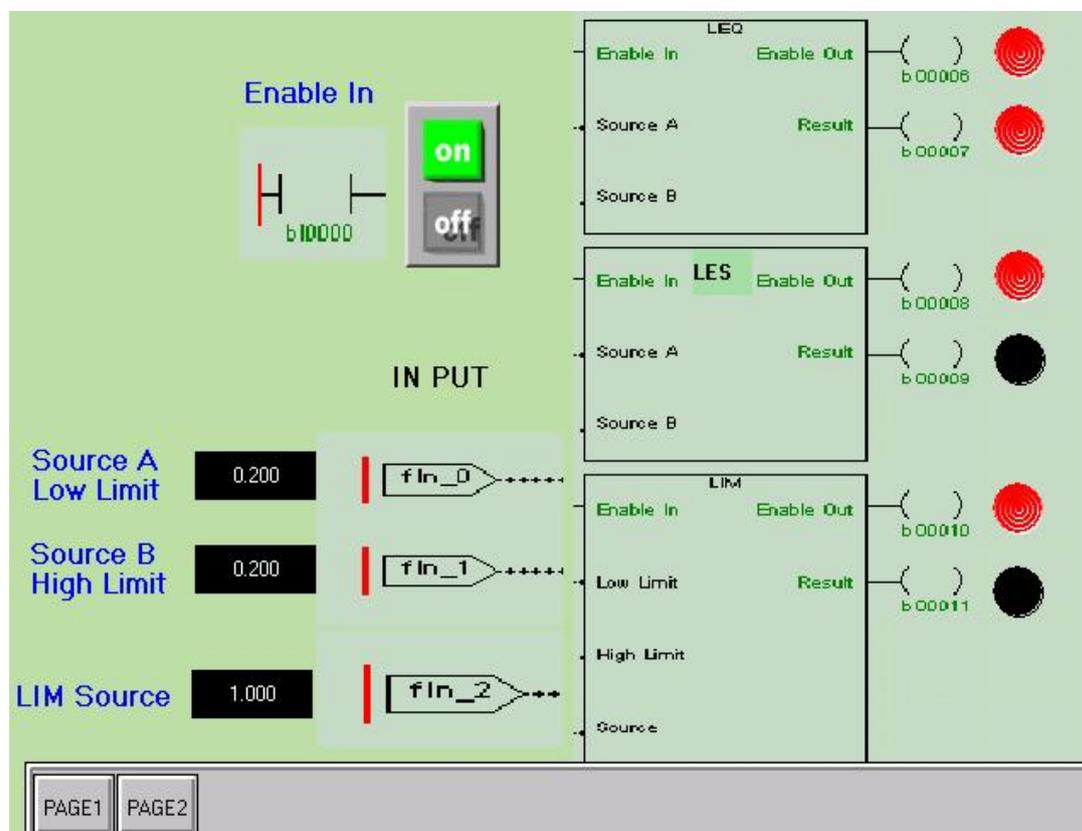
當 Source A 值小於 Source B 值時，則 Result = 1。

反之則 Result = 0。

9.比較指令驗證和操作說明：

(i)LEQ, LES, LIM 指令操作說明

(1)切到 PAGE1 畫頁，並且按下 Enable In 按鈕。操作介面如下



(2)操作 LEQ 函數：

- 輸入 Source A=0.2，Source B=0.2，此時 Source A = Source B，Result 為 TURE，因此 Result 燈亮。
- 輸入 Source A=5，此時 Source A>Source B，Result 為 FALSE，因此 Result 燈滅。
- 輸入 Source B=200。此時 Source A < Source B，Result 為 TRUE，因此 Result 燈亮。

(3)操作 LES 函數：

- a) 輸入 Source A=0.2，Source B=0.2，此時 Source A = Source B，Result 為 FALSE，因此 Result 燈滅。
- b) 輸入 Source A=5，此時 Source A>Source B，Result 為 FALSE，因此 Result 燈滅。
- c) 輸入 Source B=200。此時 Source A < Source B，Result 為 TRUE，因此 Result 燈亮。

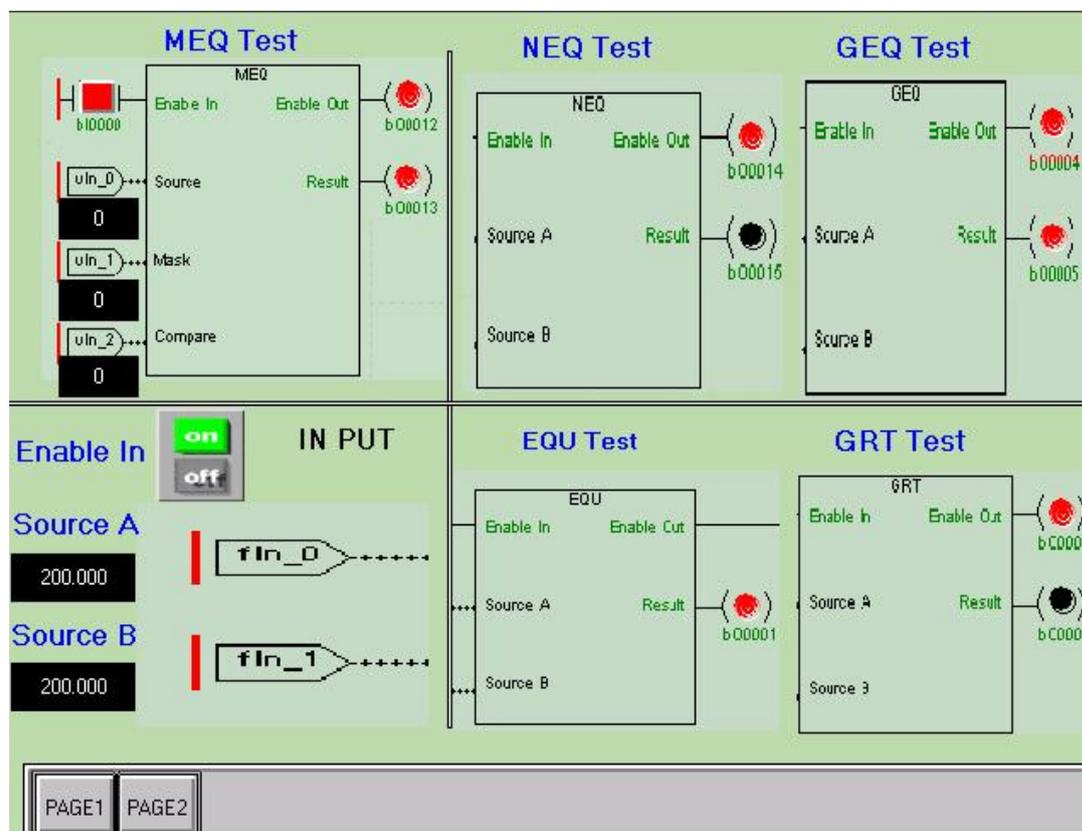
(4)操作 LIM 函數：

- a) 輸入 Low Limit=-3，High Limit=100。使得 Low Limit<High Limit。
- b) 輸入 Source=5。此時 Low Limit<Source<High Limit，因此 Result 為 TRUE，Result 燈亮。
- c) 輸入 Source=-10。此時 Source<Low Limit，因此 Result 為 FALSE，Result 燈滅。
- d) 輸入 Low Limit = 200。使得 Low Limit>High Limit。
- e) 此時 Source=-10，不落在 Low Limit > Source > High Limit 的範圍中。根據 LIM 函數說明，此時 Result 為 TRUE，Result 燈亮。
- f) 輸入 Source=105。此時 Low Limit > Source > High Limit。根據 LIM 函數說明，此時 Result 為 FALSE，Result 燈滅。



(5) 切換到 PAGE2 畫頁，維持 Enable In 按鈕為 ON 的狀態。操作介面

如下：



(6) 操作 EQU 函數：

- 輸入 Source A = 200，Source B = 200。此時 Source A = Source B，Result 為 TRUE，Result 燈亮。
- 輸入 Source A = 300。此時 Source A 不等於 Source B。Result 為 FALSE。Result 燈滅。

(7) 操作 GRT 函數

a) 輸入 Source A=300，Source B=200。此時 Source A>Source B。

Result 為 FALSE。Result 燈滅。

b) 輸入 Source B=300。此時 Source A = Source B。Result 為

FALSE，Result 燈滅。

c) 輸入 Source B=500。此時 Source A < Source B。Result 為

FALSE，Result 燈滅。

(8)操作 NEQ 函數：

a) 輸入 Source A = 200，Source B = 200。此時 Source A = Source

B，Result 為 FALSE，Result 燈滅。

b) 輸入 Source A=300。此時 Source A 不等於 Source B。Result

為 TRUE。Result 燈亮。

(9)操作 GEQ 函數：

a) 輸入 Source A=300，Source B=200。此時 Source A>Source B。

Result 為 TRUE。Result 燈亮。

b) 輸入 Source B=300。此時 Source A = Source B。Result 為

TRUE，Result 燈亮。

c) 輸入 Source B=500。此時 Source A < Source B。Result 為

FALSE，Result 燈滅。

(10)操作 MEQ 函數：

- a) 輸入 Source = 15，Mask = 3，Compare = 7。根據 MEQ 的函數說明 15 與 3 作&運算的結果和 7 與 3 作&運算的結果相同。因此 Result 為 TRUE。Result 燈亮。
- b) 輸入 Compare=1。說明 15 與 3 作&運算的結果和 1 與 3 作&運算的結果不相等。因此 Result 為 FALSE。Result 燈滅。

(11)Sample 操作完成

4.4.9 算數指令(Compute/math Instructions)

算數指令可分為下列指令：加法(Add，ADD)、減法(Subtract，SUB)、乘法(Multiply，MUL)、除法(Divide，DIV)、取餘數(Modulo，MOD)、平方根(Square Root，SQR)、反相(Negate，NEG)、絕對值(Absolute Value，ABS)、可指定加總(Selected Summer，SSUM)等指令以及各指令驗證和操作說明。

1.加法(Add，ADD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將與 Source B 進行加法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A 進行加法計算，輸入值為任一浮點數。

輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

$$\text{Source A} + \text{Source B} = \text{Dest}$$

2.減法(Subtract ，SUB)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將與 Source B 進行減法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A 進行減法計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

$$\text{Source A} - \text{Source B} = \text{Dest}$$

3.乘法(Multiply ，MUL)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將被 Source B(被乘數)進行乘法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A(乘數)進行乘法計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

$$\text{Source A} \times \text{Source B} = \text{Dest}$$

4. 除法(Divide, DIV)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將被 Source B(被除數)進行除法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A(除數)進行除法計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

$$\text{Source A} \div \text{Source B} = \text{Dest}$$

5.取餘數(Modulo ，MOD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值將被 Source B(被除數)進行除法計算，輸入值為任一浮點數。
Source B	REAL	其輸入值將與 Source A(除數)進行除法計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

說明：

當 Source A = 9 Source B = 2 則 商數=4 餘數=1

所以 Result = 1

6.平方根(Square Root ，SQR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1

		時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行開根號計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

計算式為 $\sqrt{\text{Source}} = \text{Dest}$

當 Source = 9 時 則 Dest = 3

7. 反相(Negate , NEG)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被加上負號計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

說明：

當 Source = 5 時 則 Result = -5

當 Source = -9 時 則 Result = 9

8.絕對值(Absolute Value , ABS)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被以絕對值方式計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Dest	REAL	指令計算後輸出值。

說明：

當 Source = 5 時 則 Dest = 5

當 Source = -5 時 則 Dest = 5

9.可指定加總(Selected Summer , SSUM)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1	REAL	將被加總之類比訊息輸入值 1，輸入值為任一浮點數，其預設值為 0。
Gain1	REAL	增益輸入值 1，輸入值為任一浮點數，預設值 1。

Select1	BOOL	輸入值 1 之計算訊號器，預設值為 0
---------	------	---------------------

⋮
⋮
⋮

Input 8	REAL	將被加總之類比訊息輸入值 8，輸入值為任一浮點數，其預設值為 0。
Gain8	REAL	增益輸入值 8，輸入值為任一浮點數，預設值 1。
Select8	BOOL	輸入值 8 之計算訊號器，預設值為 0
Bias	REAL	補償訊號輸入值，輸入值為任一浮點數，預設值 0。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

說明：

當全部 Input Select 值為 0 時；則 $Out = Bias$ 。

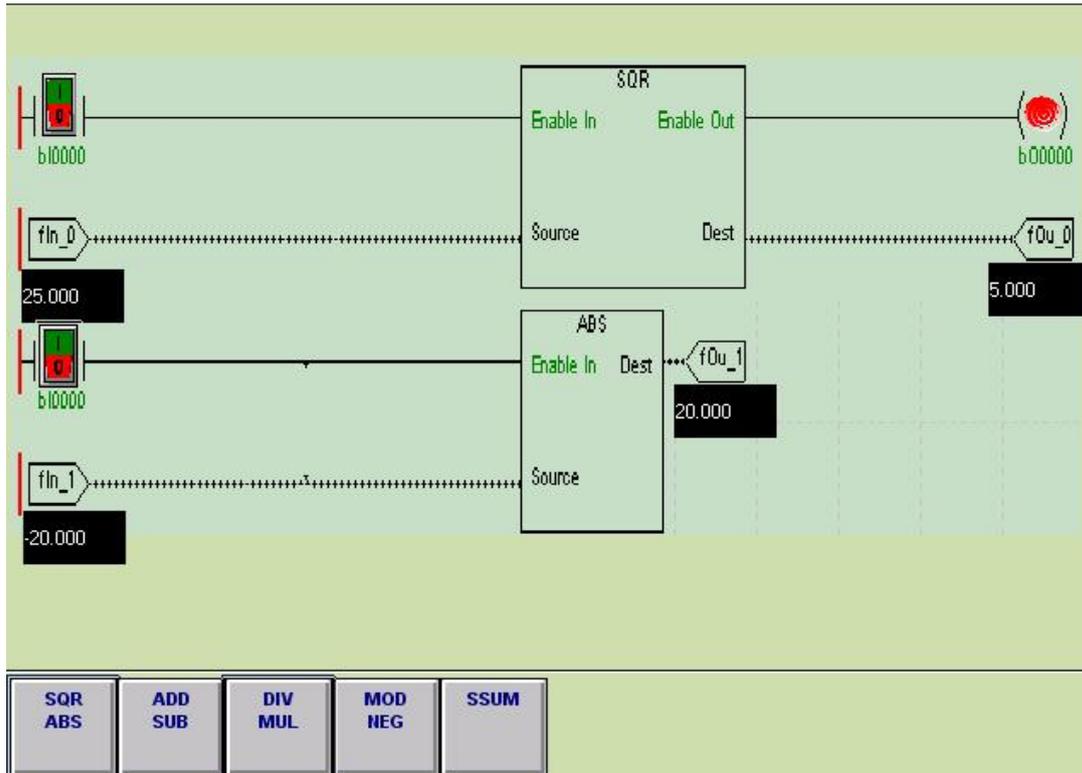
若 Input Select 值為 1 時；則

$$Out = \sum_{n=1}^8 (In_n \times Gain_n) + Bias$$

10. 算數指令驗證和操作說明

(i) 操作 SQR 及 ABS 指令

(1) 切換到 SQR/ABS 畫面並且使 Enable In 按鈕為 ON 的狀態，操作介面如下：



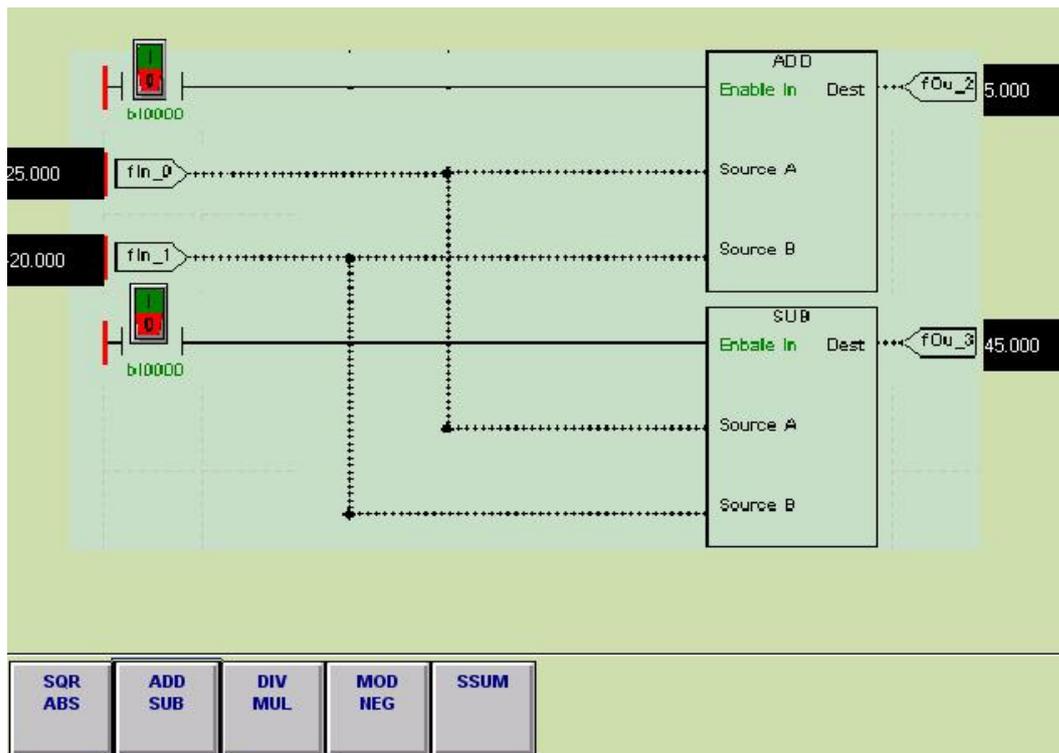
(2)輸入 SQR 函數的 Source=25，則 SQR 的 Dest=5

(3)輸入 ABS 函數的 Source=-20，則 ABS 的 Dest=20

(ii)操作 ADD 及 SUB 指令

(1)切換到 ADD/SUB 畫面，並使 Enable In 按鈕為 ON 的狀態。操作

介面如下：

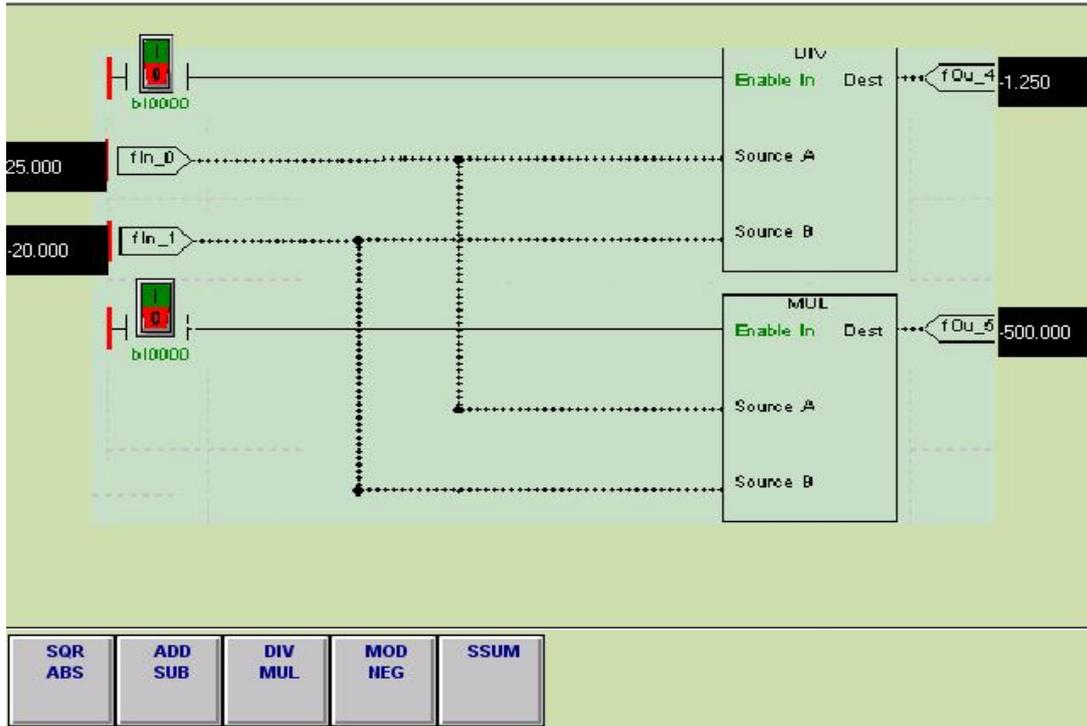


(2)輸入 ADD 函數的 Source A=25，Source B=-20 則 ADD 的 Dest=5

(3)輸入 SUB 函數的 Source A=25，Source B=-20 則 SUB 的 Dest=45

(iii)操作 DIV 及 MUL 函數：

(1)切換到 DIV/MUL 畫頁，使 Enable In 按鈕為 ON 的狀態。操作介面如下：



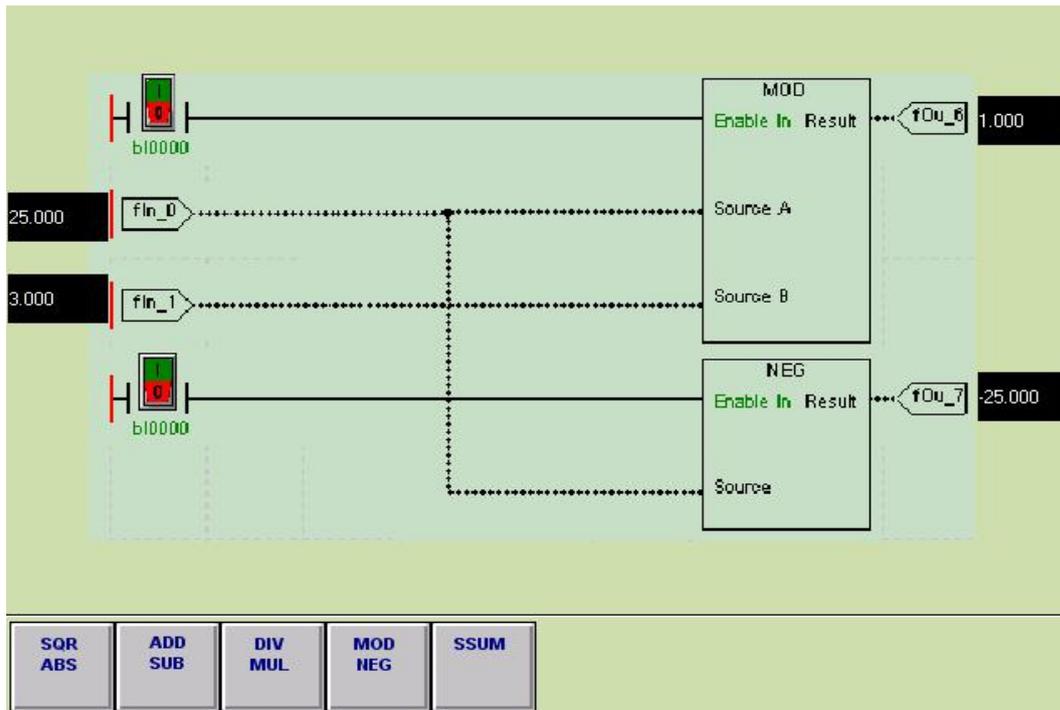
(2)輸入 DIV 函數的 Source A=25，Source B=-20 則 DIV 的 Dest=-1.25

(3)輸入 MUL 函數的 Source A=25，Source B=-20 則 MUL 的 Dest=-500

(iv)操作 MOD 及 NEG 函數

(1)切換到 MOD/NEG 畫頁，使 Enable In 按鈕為 ON 的狀態。操作介

面如下：

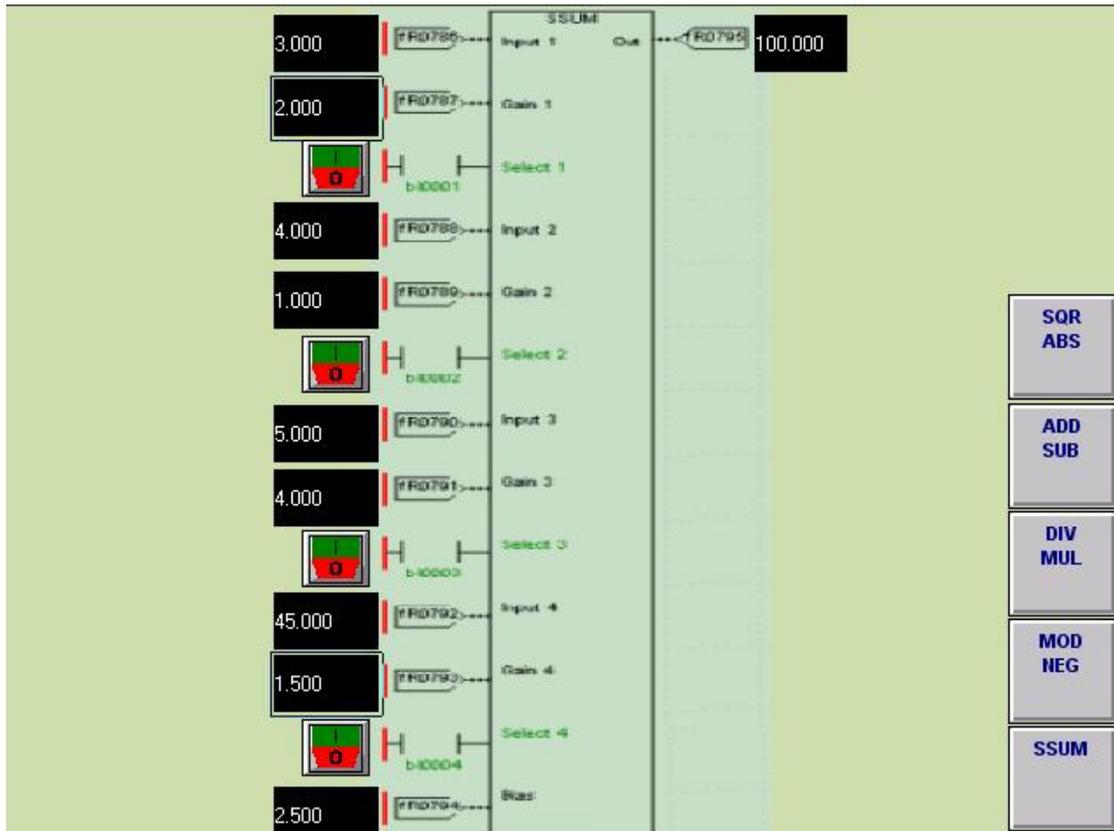


(2)輸入 MOD 函數的 Source A=25，Source B=3 則 MOD 的 Dest=1

(3)輸入 NEG 函數的 Source =25，則 NEG 的 Dest=-25

(v)操作 SSUM 函數

(1)切換到 SSUM 畫頁，使 Enable In 按鈕為 ON 的狀態。操作介面如下：



(2) 按下 Select 1~Select 4 按鈕，使按鈕狀態為 ON。表示要加總 4 個輸入。

(3) 輸入 Input 1=3，Gain 1=2，Input 2=4，Gain 2=1，Input 3=5，Gain 3=4，Input 4=45，Gain 4=1.5，Bias=2.5。則 Out =

$$3 \times 2 + 4 \times 1 + 5 \times 4 + 45 \times 1.5 + 2.5 = 100$$

4.4.10 搬移/邏輯指令(Move/Logical Instructions)

搬移/邏輯指令可分為下列指令：及閘(Bitwise And ，AND&)、遮罩搬移(Masked Move with Target ，MVMT)、或閘(Bitwise Or ，OR)、互斥或閘(Bitwise Exclusive Or ，XOR)、反閘 (Bitwise Not ，NOT)、位元及閘(Boolean AND ，BAND)、位元或閘(Boolean OR ，BOR)、位元互斥或閘(Boolean Exclusive OR ，BXOR)、位元反閘(Boolean NOT ，BNOT)、D 型正反器(D Latch Flip-Flop ，DFF)、JK 正反器(JK Flip-Flop ，JKFF)、設定型支配器(Set Dominant ，SETD)、重置型支配器(Reset Dominant ，RES D)等指令以及各指令驗證和操作說明。

1.及閘(Bitwise And ，AND&)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	DINT	其輸入值將與 Source B 進行及閘方式計算，輸入值為任一整數值。
Source B	DINT	其輸入值將與 Source A 進行及閘方式計算，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	DINT	指令計算後輸出值。

說明：

步驟二：若Mask位元值為1之Source資料將被傳送至所對應之Out位元位置，若Mask位元值為0所對應之Out位元位置之值將保持不變。

Source	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
Mask	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
Result	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	0	1	0	1	1	1	1	1	1

3. 或閘(Bitwise Or, OR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	DINT	其輸入值將與 Source B 進行或閘方式計算，輸入值為任一整數值。
Source B	DINT	其輸入值將與 Source A 進行或閘方式計算，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	DINT	指令計算後輸出值。

說明：

當指令開始運作時其計算輸出方式如下表所示：

Source A	Source B	Out
0	0	0
0	1	1
1	0	1
1	1	1

4. 互斥或閘(Bitwise Exclusive Or, XOR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	DINT	其輸入值將與 Source B 進行互斥或閘方式計算，輸入值為任一整數值。
Source B	DINT	其輸入值將與 Source A 進行互斥或閘方式計算，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	DINT	指令計算後輸出值。

範例說明：

當指令開始運作時其計算輸出方式如下表所示：

Source A	Source B	Out
0	0	0
0	1	1
1	0	1
1	1	0

5.反閘 (Bitwise Not, NOT)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
In	DINT	其輸入值將被反相輸出，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	DINT	指令計算後輸出值。

範例說明：

當指令開始運作時其計算輸出方式如下表所示：

In	Out
0	1
1	0

In	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	1	1
Out	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0	0

6.位元及閘(Boolean AND , BAND)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1~8	BOOL	輸入值 1~8 至指令進行位元及閘運算，輸入值為任一布林值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。

範例說明：

BAND指令有8個輸入值進行位元及閘計算，若任一輸入值未使用時其預設值為1，若其中任一輸入值為0時，則Out = 0。

指令計算公式如下：

$$\text{Out} = \text{Input1 AND Input 2 AND Input 3 AND Input 4 AND Input 5 AND Input 6 AND Input 7 AND Input 8}$$

範例：Input1及Input 2進行位元及閘計算輸出值，如下列所列

Input 1	Input 2	Out
0	0	0
0	1	0
1	0	0
1	1	1

7.位元或閘(Boolean OR ，BOR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1~8	BOOL	輸入值 1~8 至指令進行位元或閘運算，輸入值為任一布林值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。

範例說明：

BOR指令有8個輸入值進行位元或閘計算，若任一輸入值未使用時其預設值為0，若其中任一輸入值為1時，則Out = 1。

指令計算公式如下：

$$\text{Out} = \text{Input 1 OR Input 2 OR Input 3 OR Input 4 OR Input 5 OR Input 6 OR Input 7 OR Input 8}$$

範例：Input1及Input 2進行位元及閘計算輸出值，如下列所示

Input 1	Input 2	Out
0	0	0
0	1	1
1	0	1
1	1	1

8.位元互斥或閘(Boolean Exclusive OR , BXOR)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input 1	BOOL	第一輸入值至指令進行位元互斥或閘運算，輸入值為任一布林值。
Input 2	BOOL	第二輸入值至指令進行位元互斥或閘運算，輸入值為任一布林值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。

範例說明：

BXOR指令為在Input 1及Input 2兩者值進行位元互斥或閘計算。

指令計算公式： $Out = Input 1 \text{ XOR } Input 2$

範例：Input1及Input 2進行位元互斥或閘計算輸出值，如下列所示

Input 1	Input 2	Out
0	0	0
0	1	1
1	0	1
1	1	0

9.位元反閘(Boolean NOT ，BNOT)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Input	BOOL	其輸入值將被反相輸出，輸入值為任一布林值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。

範例說明：

當指令開始運作時其計算輸出方式如下表所示：

Source	Out
0	1
1	0

10.D型正反器(D Latch Flip-Flop, DFF)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
D	BOOL	為一布林輸入值，預設值為 0。
Clear	BOOL	若為 1 時則 Q=1 及 Not Q=1
Clock	BOOL	時脈輸入點，預設值為 0。
L/E Sel	BOOL	值為 1 時為 Latch 型，值為 0 時為 edge Trigger 型，預設值為 1。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Q	BOOL	指令計算後輸出值。
Q Not	BOOL	與 Q 值反相輸出

範例說明：

當 Clear = 1 時，則 Q = 0 及 Q Not = 1

若 Clock = 1 及 Clock_{n-1} = 0 時，則 Q = D 及 Q Not = Not(D)

11. JK正反器(JK Flip-Flop, JKFF)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Clear	BOOL	若為 1 時則 Q=1 及 Not Q=1
Clock	BOOL	時脈輸入點，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Q	BOOL	指令計算後輸出值。
Q Not	BOOL	與 Q 值反相輸出

說明：

當 Clear = 1 時，則 Q = 0 及 Q Not = 1

若 Clock = 1 及 Clock_{n-1} = 0 時，則 Q = 1 及 Q Not = 0

12. 設定型支配器(Set Dominant, SETD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Set	BOOL	設定輸入值，預設值為 0
Reset	BOOL	重置輸入值，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)

Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。
Out Not	BOOL	與 Out 值反相輸出

範例說明：

當 Set = 1 時，則 Out = 1 及 Out Not = 0

當 Reset = 1 及 Set = 0 時，則 Out = 0 及 Out Not = 1

13. 重置型支配器 (Reset Dominant, RESD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Set	BOOL	設定輸入值，預設值為 0
Reset	BOOL	重置輸入值，預設值為 0
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	BOOL	指令計算後輸出值。
Out Not	BOOL	與 Out 值反相輸出

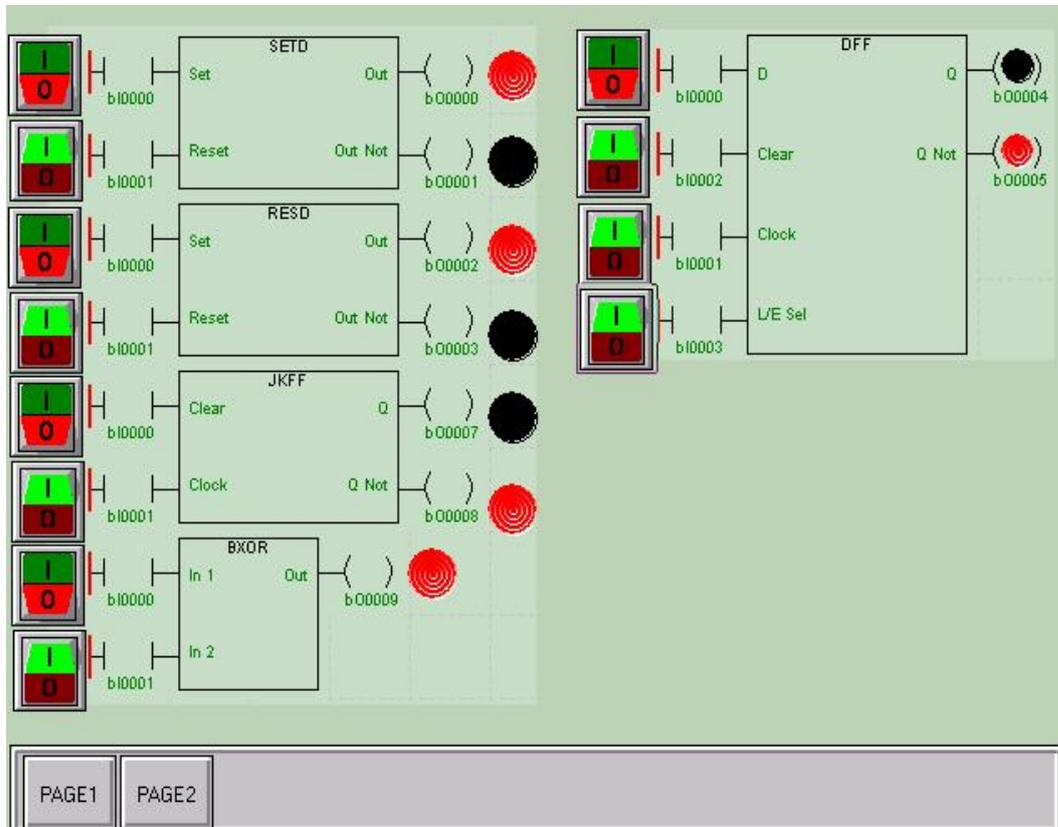
範例說明：

當 Set = 1 及 Reset = 0 時，則 Out = 1 及 Out Not = 0

當 Reset = 1 時，則 Out = 0 及 Out Not = 1

14.搬移/邏輯指令驗證和操作說明

(i) 切換到 PAGE1 畫頁，操作畫面如下



(1) 操作 SETD 指令：

- 按下 Set 按鈕，使 Set 按鈕狀態為 ON，此時 Out 為 TRUE，Out 燈亮，Out Not 為 FALSE，Out Not 燈滅。
- 使 Set 按鈕狀態為 OFF，使 Reset 按鈕狀態為 ON，則此時 Out 被 Reset 為 FALSE，Out 燈滅。Out Not 為 TRUE，Out Not 燈亮。

(2) 操作 RESD 指令：

- a) 按下 Set 按鈕，使 Set 按鈕狀態為 ON，此時 Out 為 TRUE，Out 燈亮，Out Not 為 FALSE，Out Not 燈滅。
- b) 不改變 Set 按鈕狀態，使 Reset 按鈕狀態為 ON，則此時 Out 被 Reset 為 FALSE，Out 燈滅。Out Not 為 TRUE，Out Not 燈亮。

(3)操作 JKFF 指令：

- a) 使 Clock 按鈕為 OFF 狀態。按下 Clock 按鈕，使 Clock 按鈕狀態為 ON，則此時 Q 為 TRUE，Q 燈號亮。Q Not 為 FALSE，Q Not 燈號滅。
- b) 使 Clear 按鈕為 ON 狀態，則 Q 為 FALSE，Q 燈號滅。Q Not 為 TRUE，Q Not 燈號亮。
- c) 使 Clear 按鈕為 OFF 狀態，則 Q 為 FALSE，Q 燈號滅。Q Not 為 TRUE，Q Not 燈號亮。

(4)操作 BXOR 指令：

- a) 輸入 In 1 與 In 2 按鈕狀態與輸出結果如下表

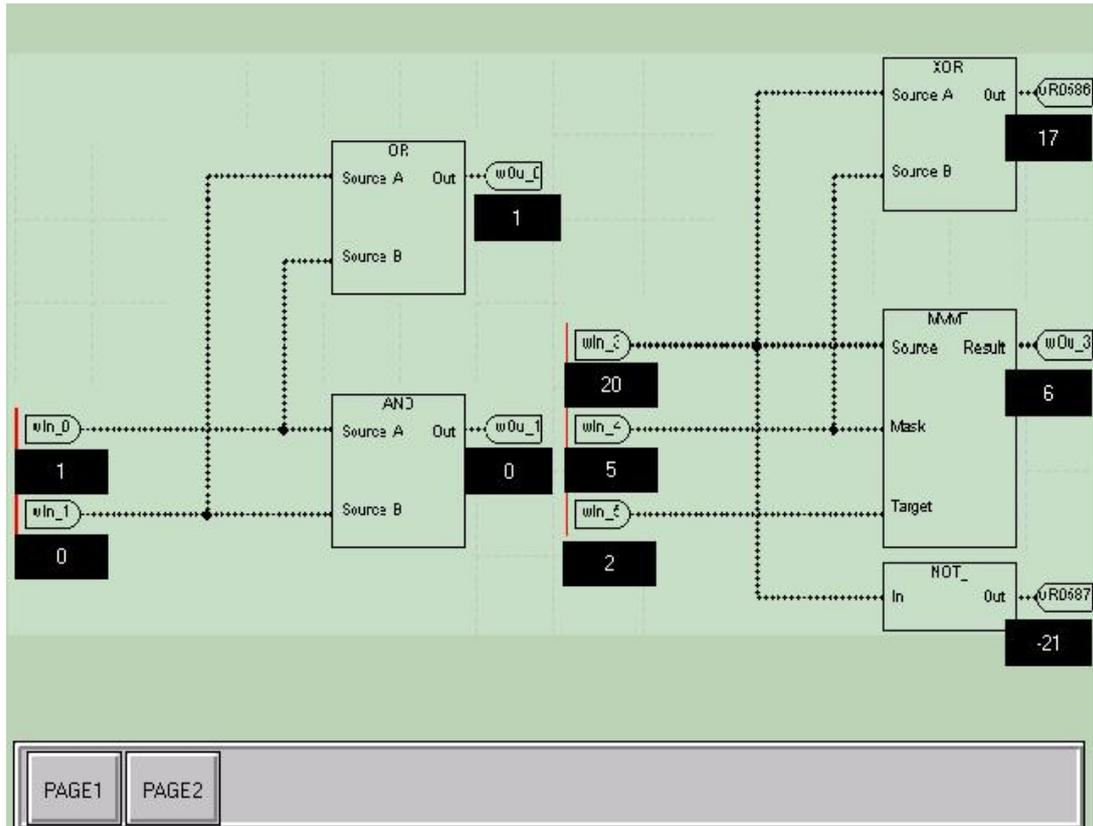
In 1 按鈕	In 2 按鈕	Out
OFF	OFF	FALSE(燈滅)
OFF	ON	TRUE(燈亮)
ON	OFF	TRUE(燈亮)

ON	ON	FALSE(燈滅)
----	----	-----------

(5)操作 DFF 指令

- a) 使 L/E Sel 按鈕狀態為 OFF，進入上緣觸發模式。
- b) 使 D 按鈕的狀態為 ON
- c) 使 Clock 按鈕的狀態為 ON，則此時 $Q = D$ 。Q 燈亮。Q Not 燈滅。
- d) 使 Clear 按鈕 ON, OFF 一次，此時 Q 燈號滅，Q Not 燈亮。
- e) 使 D 按鈕狀態為 OFF。使 Clock 按鈕狀態為 OFF。
- f) 使 Clock 按鈕的狀態為 ON，則此時 $Q = D$ 。Q 燈滅。Q Not 燈亮。
- g) 使 Clear 按鈕 ON, OFF 一次，此時 Q 燈號滅，Q Not 燈亮。
- h) 使 Clock 按鈕狀態為 OFF。使 L/E Sel 按鈕狀態為 ON。
- i) 使 Clock 按鈕狀態為 ON，此時 $Q = D$ ，若 D 按鈕狀態為 ON 則 Q 燈亮，若切換 D 按鈕狀態為 OFF，則 Q 燈滅。

(ii)切換到 PAGE2，介面如下



(1)操作 OR 指令

- a) Out 值為 Source A 與 Source B 作 bit 腳位的 OR 運算結果。
- b) 運算舉例如下

Source A	Source B	Out
2	1	3
15	0	15
8	7	15

(2)操作 AND 指令

- a) Out 值為 Source A 與 Source B 作 bit 腳位的 AND 運算結果。
- b) 運算舉例如下

Source A	Source B	Out
2	1	0

15	3	3
8	15	8

(3)操作 XOR 指令

a) Out 值為 Source A 與 Source B 作 bit 腳位的 XOR 運算結果。

b) 運算舉例如下

Source A	Source B	Out
2	1	3
15	1	14
9	7	14

(4)操作 MVMT 指令

a) 根據 MVMT 指令說明運算與例如下

Source	Mask	Target	Result
8	15	3	8
7	14	2	6

(5)操作 NOT 指令

a) Out 為 In 的 bit 腳位作 NOT 運算後的結果

b) 運算舉例如下

In	Out
65535(0xffff)	-65536 (0xffff0000)
0	-1(0xffffffff)

4.4.11 三角函數指令(Trig Functions Instructions)

三角函數指令可分為下列指令：正弦(Sine ，SIN)、餘弦(Cosine ，COS)、正切(Tangent ，TAN)、反正弦(Arc Sine ，ASN)、反餘弦(Arc Cosine ，ACS)、反正切(Arc Tangent ，ATN)等指令以及各指令驗證和操作說明

1. 正弦(Sine ，SIN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

範例說明：

其輸入值Source需在 $\geq -205887.4 (-2\pi \times 2^{15})$ 或 $\leq 205887.4 (2\pi \times 2^{15})$

範圍內，以及經數學運算後輸出值Result需為 ≥ -1 或 ≤ 1 範圍內

值。當Source = 0.78539819 時 則 Result = 0.70710677

2.餘弦(Cosine ，COS)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

範例說明：

其輸入值Source需在 ≥ -205887.4 ($-2\pi \times 2^{15}$) 或 ≤ 205887.4 ($2\pi \times 2^{15}$)

範圍內，以及經數學運算後輸出值Result需為 ≥ -1 或 ≤ 1 範圍內

值。當 **Source = 1.0471976** 時 則 **Result = 0.5**

3.正切(Tangent ，TAN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數	資料型態	說明

(Output Parameter)	(Data Type)	(Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

範例說明：

其輸入值Source需在 $\geq -102943.7(-2\pi \times 2^{14})$ 或 $\leq -102943.7(2\pi \times 2^{14})$

範圍內。當Source = 2.0343552 時 則Result = -1.7320507

4.反正弦(Arc Sine ，ASN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

範例說明：

其輸入值Source需在 ≥ -1 或 ≤ 1 範圍內，以及經數學運算後輸出值

Result需為 $\geq -\pi/2$ 或 $\leq \pi/2$ 範圍內值(當 $\pi = 3.141593$)。

當Source = 0.2 時 則 Result = 0.20135795

5.反餘弦(Arc Cosine ， ACS)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令計算後輸出值。

範例說明：

其輸入值Source需在 ≥ -1 或 ≤ 1 範圍內，以及經數學運算後輸出值Result需為 ≥ 0 或 $\leq \pi$ 範圍內值(當 $\pi = 3.141593$)。

當 Source = -0.6000002 時 則 Result = 2.2142975

6.反正切(Arc Tangent ， ATN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數	資料型態	說明

(Output Parameter)	(Data Type)	(Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

範例說明：

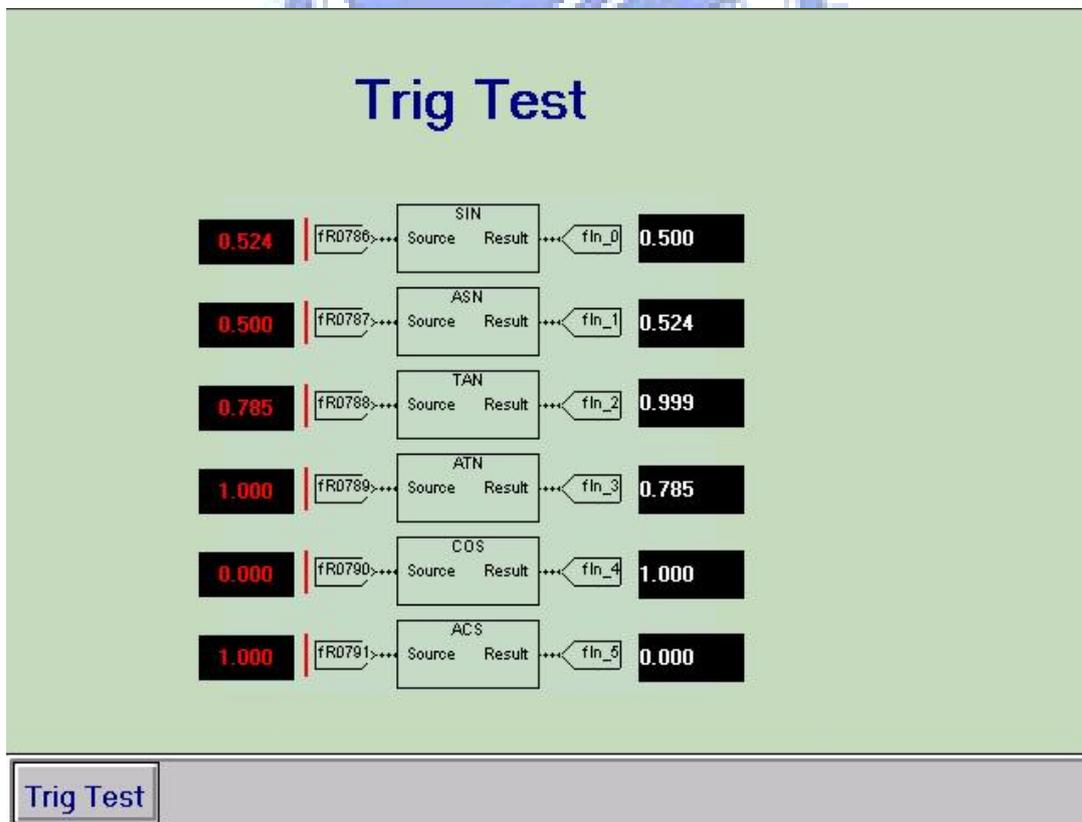
經數學運算後輸出值Result需為 $\geq -\pi/2$ 或 $\leq \pi/2$ 範圍內值

(當 $\pi = 3.141593$)。

當Source = 0.86000001 時 則 Out = 0.71027106

7.三角函數指令驗證和操作說明

(i)操作介面如下



(1)操作各函數舉例如下表

函數名稱	Source	Result
SIN	0.524(30度)	0.5
ASIN	0.5	0.524
COS	0(0度)	1
ACOS	1	0
TAN	0.785(45度)	1
ATAN	1	0.785

4.4.12 高等數學指令(Advanced Math Instructions)

高等數學指令可分為下列指令：自然對數(Natural Log ，LN)、對數(Log Base 10 ，LOG)、冪次方(X to the Power of Y ，XPY)等指令以及各指令驗證測試。

1.自然對數(Natural Log ，LN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

範例說明：

其輸入值需 >0 ，經數學運算後輸出值Out需為 ≥ -87.33655 或 ≤ 88.72284 範圍內值。

當 Source = 5.0 時 則 Out = 1.6094378

2. 對數(Log Base 10，LOG)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行數學計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

範例說明：

其輸入值需 >0 ，經數學運算後輸出值Out需為 ≥ -37.92978 或 ≤ 38.53184 範圍內值。

當 Source = 3.4000001 時 則 Out = 0.53147888

3. 冪次方(X to the Power of Y，XPY)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source A	REAL	其輸入值為基數(X)，輸入值為任一浮點數。
Source B	REAL	其輸入值為指數(Y)，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令計算後輸出值。

範例說明：

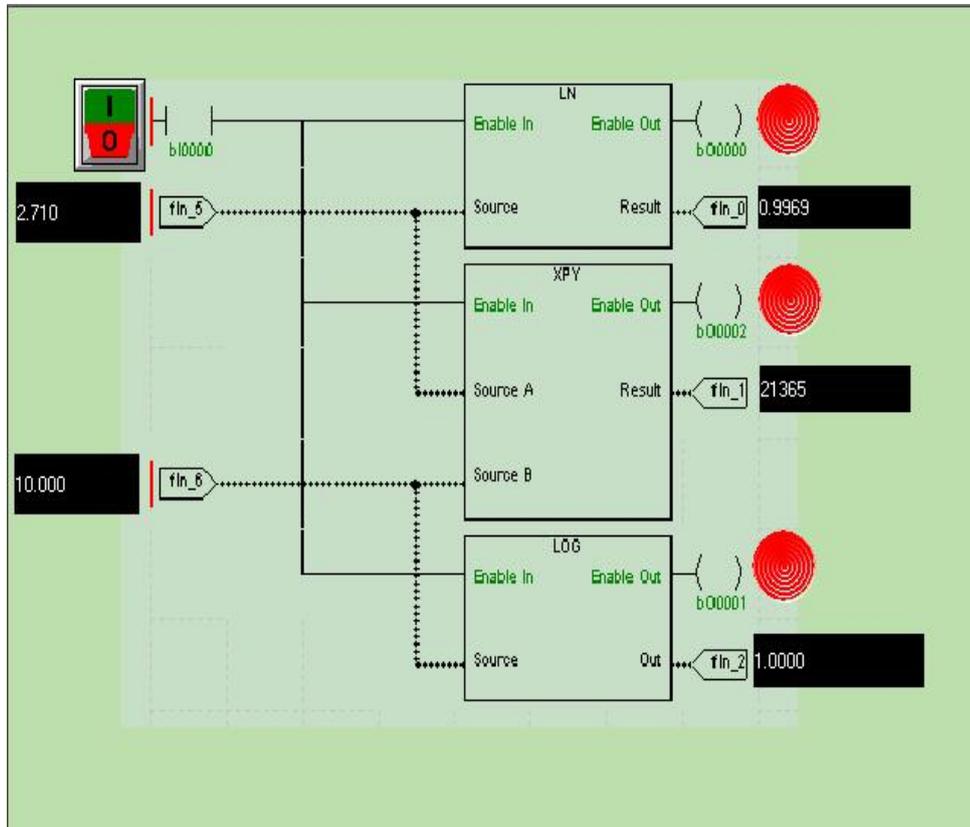
當 Source A(X) 為一負值則 Source B(Y) 必須為一整數。

指令計算方式： $X^0 = 1$ 和 $0^X = 0$ 。

當 Source A = 3 及 Source B = 2 時則 $3^2 = 9$

4. 高等數學指令驗證測試

(i) 操作介面如下



(1) 使 Enable In 按鈕狀態為 ON

(2) 操作 LN 指令

a) 輸入 Source=2.71，Result=0.9969

(3) 操作 XPY 指令

a) 輸入 Source A=3，Source B=2，則 Result=9

(4) 操作 LOG 指令

a) 輸入 Source=10，則 Result=1

4.4.13 數學轉換指令(Math Conversions Instructions)

數學轉換指令可分為下列指令：弧/角轉換(Degrees ，DEG)、角/弧轉換(Radians ，RAD)、BCD 轉換(Convert to BCD ，BCD)、Integer 轉換(Convert to Integer ，FRD)、去除小數(Truncate ，TRN)等指令以及各指令驗證和操作說明。

1.弧/角轉換(Degrees ，DEG)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行轉換計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Out	REAL	指令轉換計算後輸出值。

說明：

將弧度值轉換為角度值，其指令計算公式：

$$\text{Source} \times 180/\pi \text{ (當}\pi= 3.141593)$$

2.角/弧轉換(Radians ，RAD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行轉換計算，輸入值為任一浮點數。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	REAL	指令轉換計算後輸出值。

範例說明：

將角度值轉換為弧度值，其指令計算公式：

Source $\times \pi / 180$ (當 $\pi = 3.141593$)

3.BCD轉換(Convert to BCD ，BCD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
I	DINT	其輸入值將被進行二進制編碼的十進制轉換計算，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)

Enable In	BOOL	指令可運作執行輸出值。
O	DINT	指令轉換計算後輸出值。

說明：

將十進制值轉換為二進制編碼值。

4.Integer轉換(Convert to Integer ，FRD)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
I	DINT	其輸入值將被進行十進制轉換計算，輸入值為任一整數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
O	DINT	指令轉換計算後輸出值。

說明：

將二進制編碼值轉換為十進制值。

5. 去除小數(Truncate , TRN)

結構及功能說明：

輸入參數 (Input Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	如果值為 0，則指令將無法執行而且指令運作輸出值將為 0，如果值為 1 時則指令可執行運作，預設值為 1。
Source	REAL	其輸入值將被進行整數值轉換計算，輸入值為任一浮點數值。
輸出參數 (Output Parameter)	資料型態 (Data Type)	說明 (Description)
Enable In	BOOL	指令可運作執行輸出值。
Result	DINT	指令轉換計算後輸出值。

範例說明：

將浮點數值轉換為整數值，當 Source = 564.234，則 Result = 564。

6. 數學轉換指令驗證和操作說明

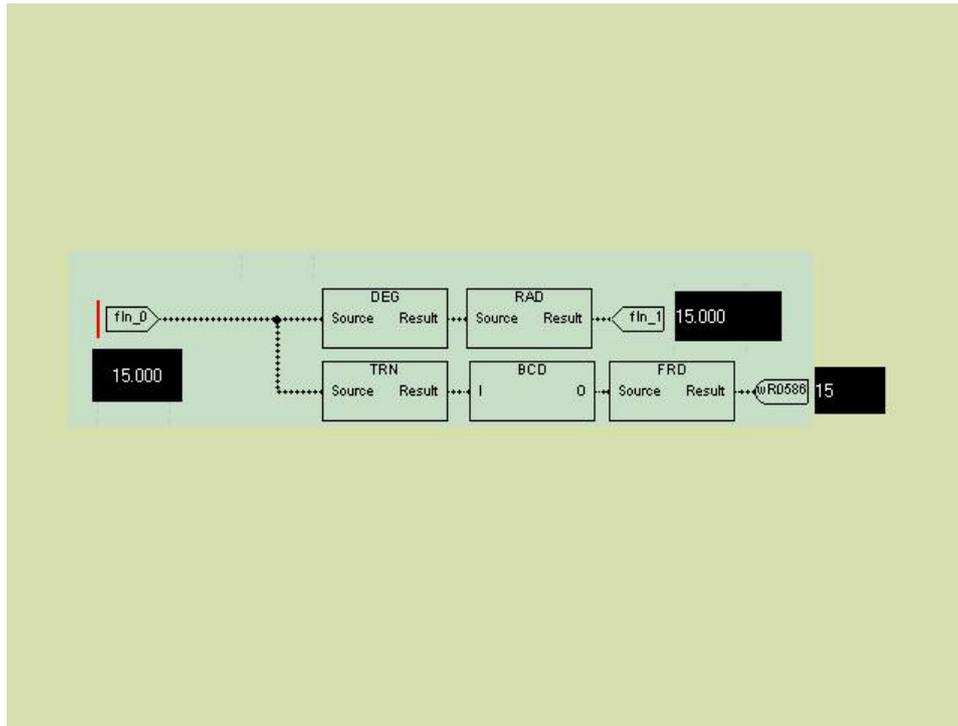
(i) 範例說明：

此範例利用以下轉換來驗證轉換函數是否正確。

(1) 將角度輸入值轉換成弧度再轉換成角度輸出，輸出值必須與輸入值相同

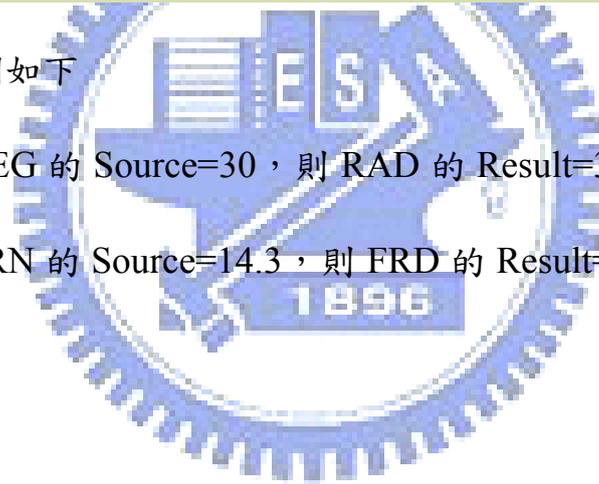
(2) 將浮點數輸入值無條件捨去成整數後轉換成 BCD 格式，再轉換成浮點數，輸出值必須與輸入值經無條件捨去後的整數值相同

(3) 操作介面如下：



(4)操作舉例如下

- a) 輸入 DEG 的 Source=30，則 RAD 的 Result=30
- b) 輸入 TRN 的 Source=14.3，則 FRD 的 Result=14



五.結論及未來展望

5.1 結論

本論文WINPC32 PLC功能方塊指令之製作以參考他廠之運作模式並加入既有系統架構進行轉換而建立完成，功能方塊指令可成為12種功能屬性之功能方塊指令，再依屬性種類共計完成80個指令，每一功能方塊指令皆已完成測試及運算並可獲得預期所得之結果，在WINPC32 PLC軟體編輯平台上完成功能方塊指令語言之建立且在程式編譯上亦可正確性地反應出程式編撰者所想要之結果出來，使得WINPC32 PLC軟體編輯器增加了使用功能方塊語言編輯程式之功能並增加了其市場上之競爭性。

5.2 未來展望

目前WINPC32-PLC功能方塊指令未能完善地全部建立起來，因會有因應使用者之需求不在預設建置內，但系統上仍有建置因應使用需求可自行建立新增功能方塊指令的平台，此部份需由使用者自行增設，未來日後仍有改善空間。

參考文獻

1. Hurco, WINPC32 Technical Manual, 1999
2. Allen-Bradley Logix5000™ Controller Process Control and Drives
Instruction user manual
3. Allen-Bradley Logix5000™ Controllers General Instructions
Reference Manual
- 4.游武璋，工控器指令集之語法結構分析，國立交通大學機械工程研
究所碩士論文，民 88 年。
5. Allen-Bradley Logix5000 Controllers Motion Instructions Reference
Manual
6. Allen-Bradley Logix5000™ Controller Function Block Instruction
User manual
- 7.謝瑞成，以 Windows 為平台之可程式邏輯控制系統架構之研究



表目錄

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to add to SourceB. Valid = any float
SourceB	REAL	Value to add to SourceA. Valid = any float
Output Parameter	Data Type	Description
EnableOut	BOOL	Enable output.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

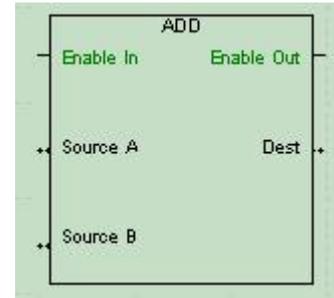


表 2.1 加法器(ADD)功能方塊指令屬性表

MINIMUM_CAPTURE Structure		
Input Parameter	Data Type	Description
EnableIn	BOOL	Function Block: Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set. Structured Text: No effect. The instruction executes.
In	REAL	The analog signal input to the instruction. Valid = any float Default = 0.0
Reset	BOOL	Request to reset control algorithm. The instruction sets Out = ResetValue as long as Reset is set. Default is cleared.
ResetValue	REAL	The reset value for the instruction. The instruction sets Out = ResetValue as long as Reset is set. Valid = any float Default = 0.0
Output Parameter	Data Type	Description
EnableOut	BOOL	Enable output.
Out	REAL	The calculated output of the algorithm. Arithmetic status flags are set for this output.

表 2.2 取最小值(MINC)功能方塊指令屬性表

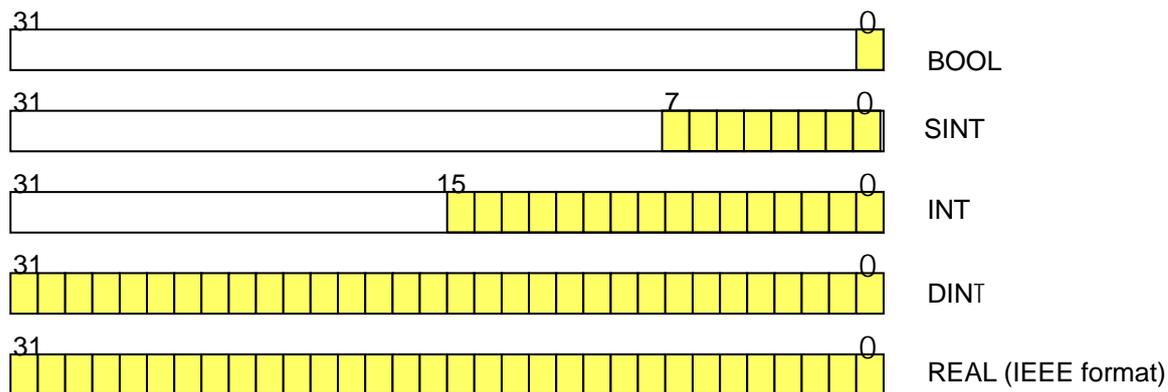


表 3.1 內部資料型態表

Select	Caption	Value	Data Type	Description
<input type="checkbox"/>	Enable In	TRUE	BOOLEAN	Enable input
<input checked="" type="checkbox"/>	Init	FALSE	BOOLEAN	The initialize input req...
<input checked="" type="checkbox"/>	InitValue	0	REAL	The initialize value of ...
<input checked="" type="checkbox"/>	Input+	0	REAL	The input added to the...
<input checked="" type="checkbox"/>	Input-	0	REAL	The input subtracted f...
<input checked="" type="checkbox"/>	Hold	FALSE	BOOLEAN	The hold input request...
<input type="checkbox"/>	Enable Out	FALSE	BOOLEAN	Enable output
<input checked="" type="checkbox"/>	Output	0	REAL	The output of the instr...

表 3.2 功能方塊指令參數表

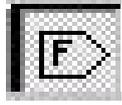
Reference Type	Description	Mnemonic	Symbol
Input Reference	Corresponds to a tag and brings a value from memory into the function block routine	IREF	
Output Reference	Corresponds to a tag and sends a value from the function block routine into memory	OREF	

表 3.3 輸入/輸出參考點表



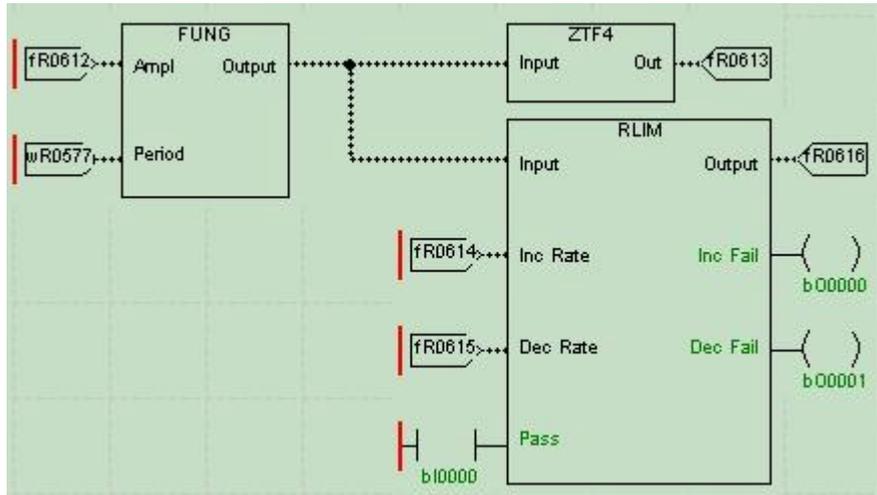


圖 1.1 功能方塊圖程式

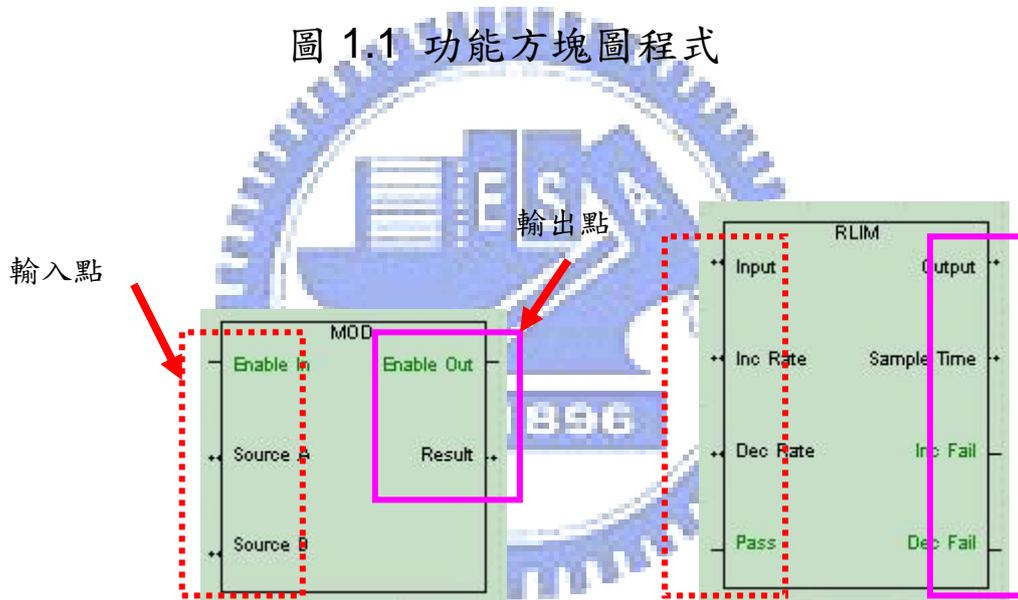


圖 1.2 功能方塊指令輸入及輸出點

當 Reset=1 時, 則 Out=Reset Value.
 當 Reset=0 時, 若 $In < Out_{n-1}$ 時, $Out = In$
 若 $In \geq Out_{n-1}$ 時, $Out = Out_{n-1}$

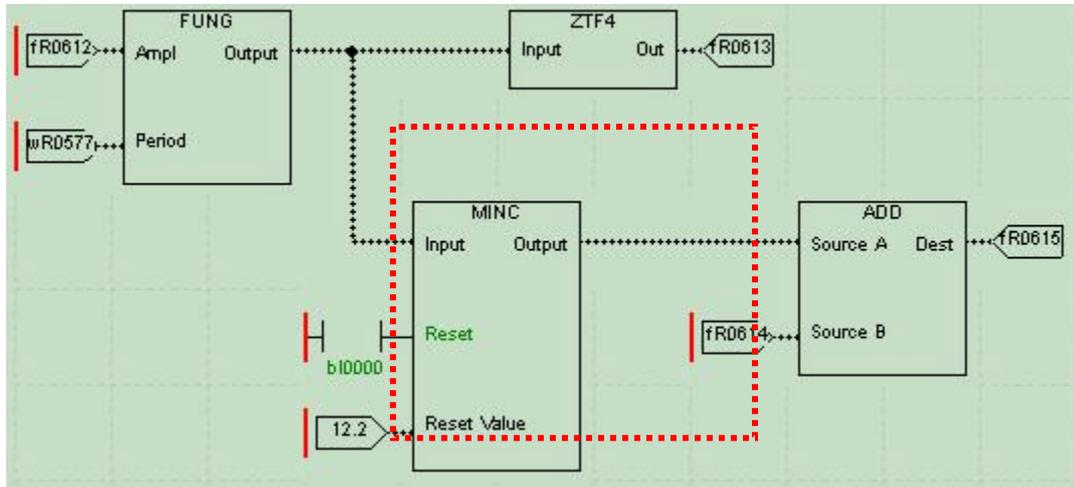


圖 2.1 取最小值(MINC)功能方塊指令運算條件



圖 2.2 加法器(ADD)功能方塊指令轉換成 A 廠階梯圖型態

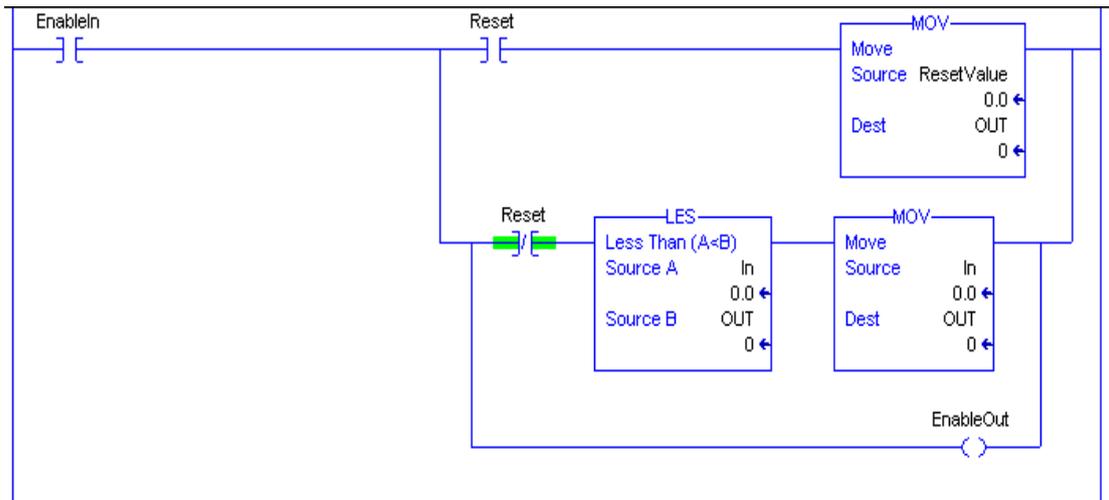


圖 2.3 取最小值(MINC)功能方塊指令轉換成 A 廠階梯圖型態

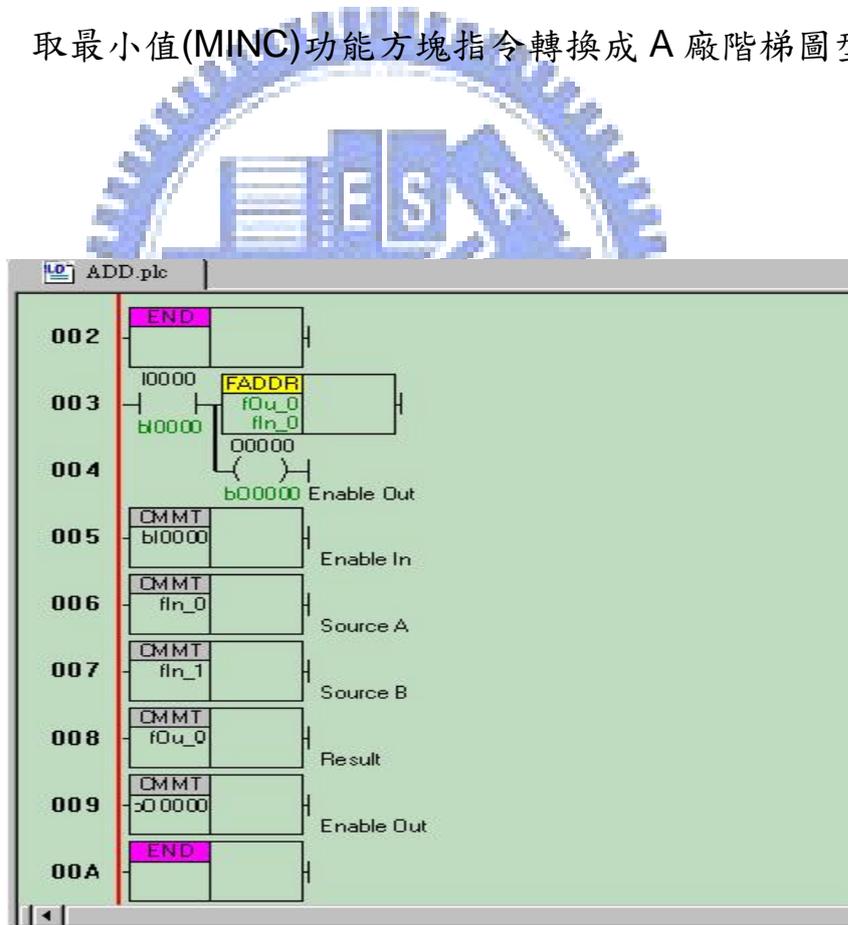


圖 2.4 加法器(ADD)功能方塊指令轉換成 WINPC32 階梯圖型態

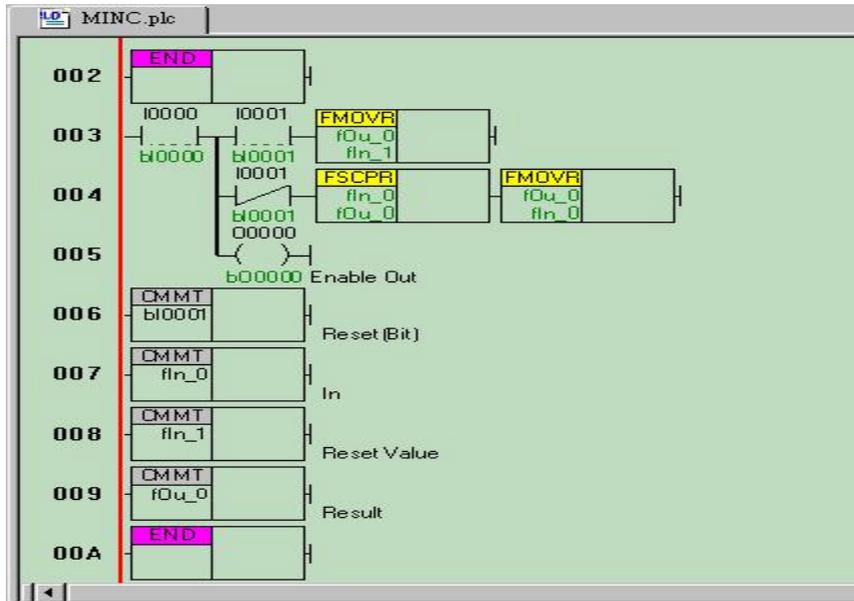


圖 2.5 取最小值(MINC)功能方塊指令轉換成 WINPC32 階梯圖型態

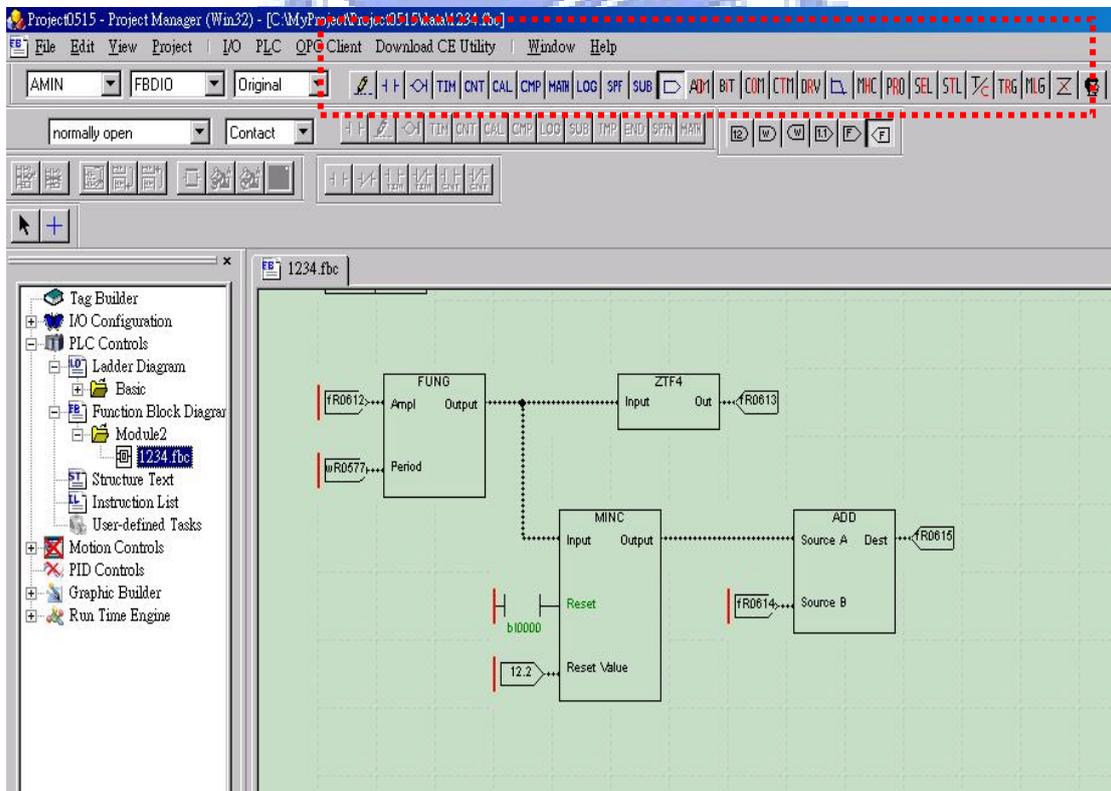


圖 3.1 功能方塊指令工具列

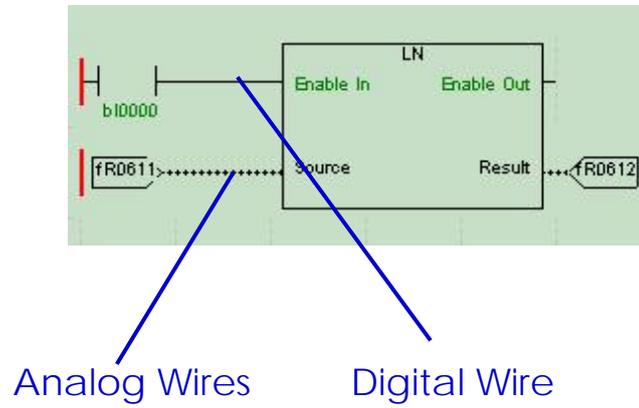


圖 3.2 類比/數位連接線

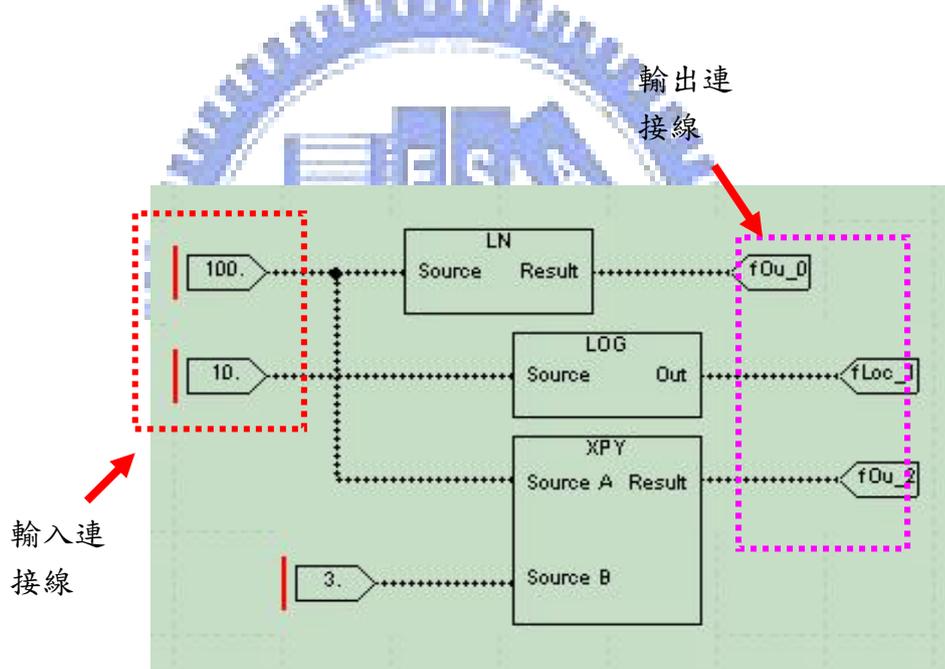


圖 3.3 輸入及輸出連接線連接器

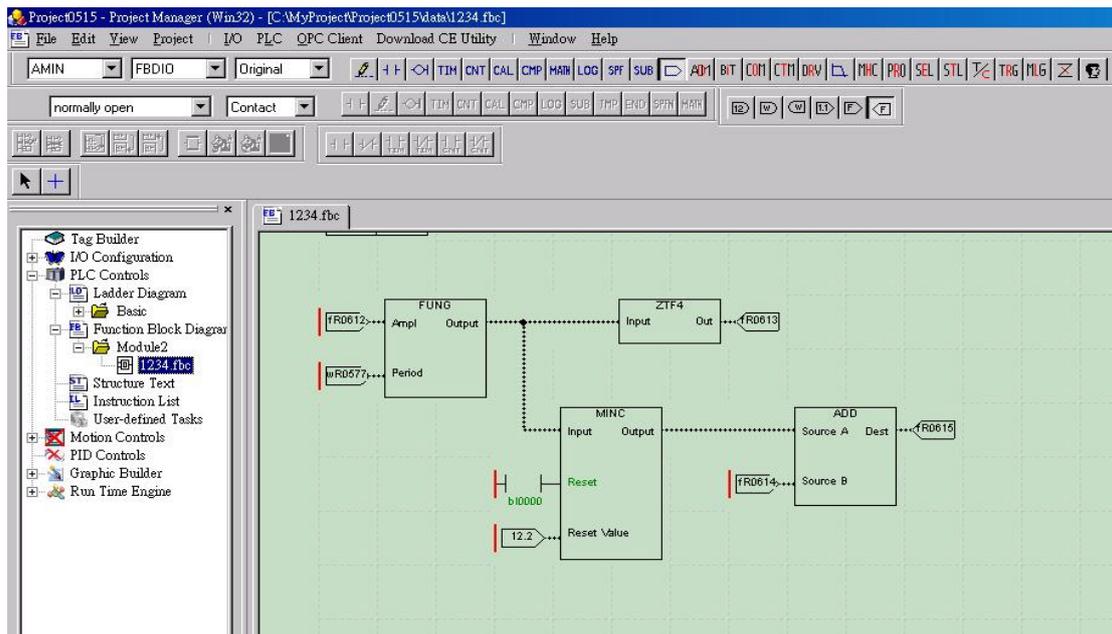


圖 3.4 A 廠程式執行架構

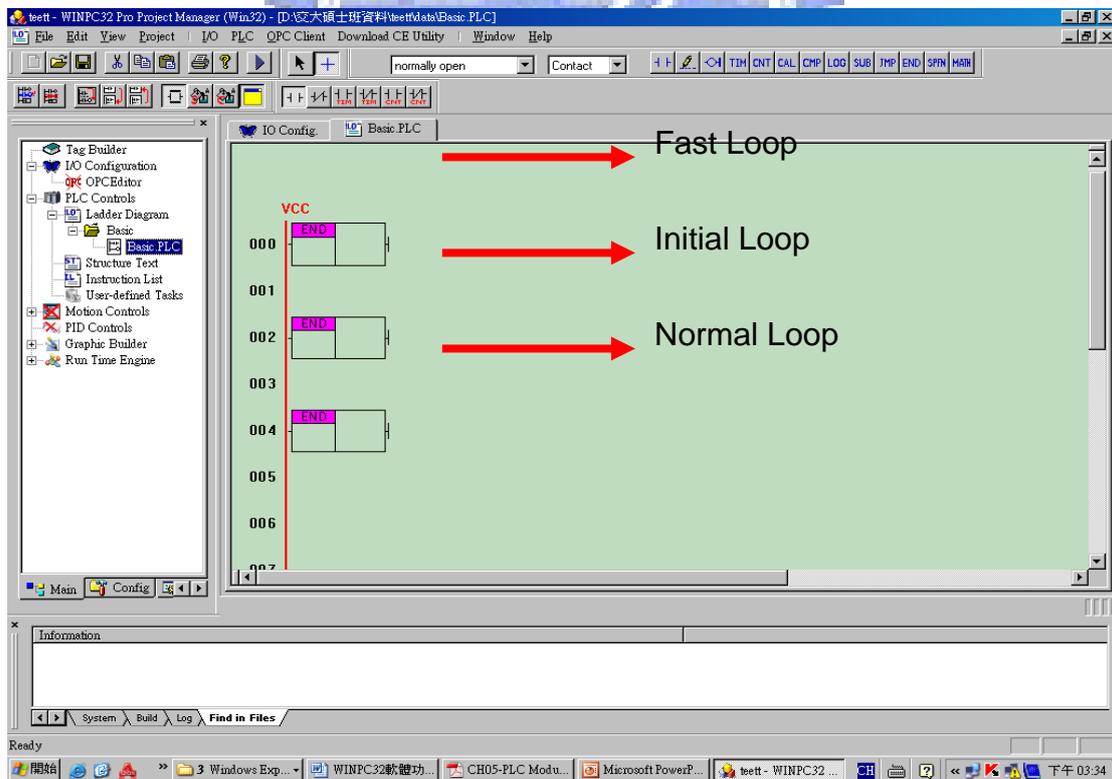


圖 3.5 WINPC32 階梯圖迴圈種類

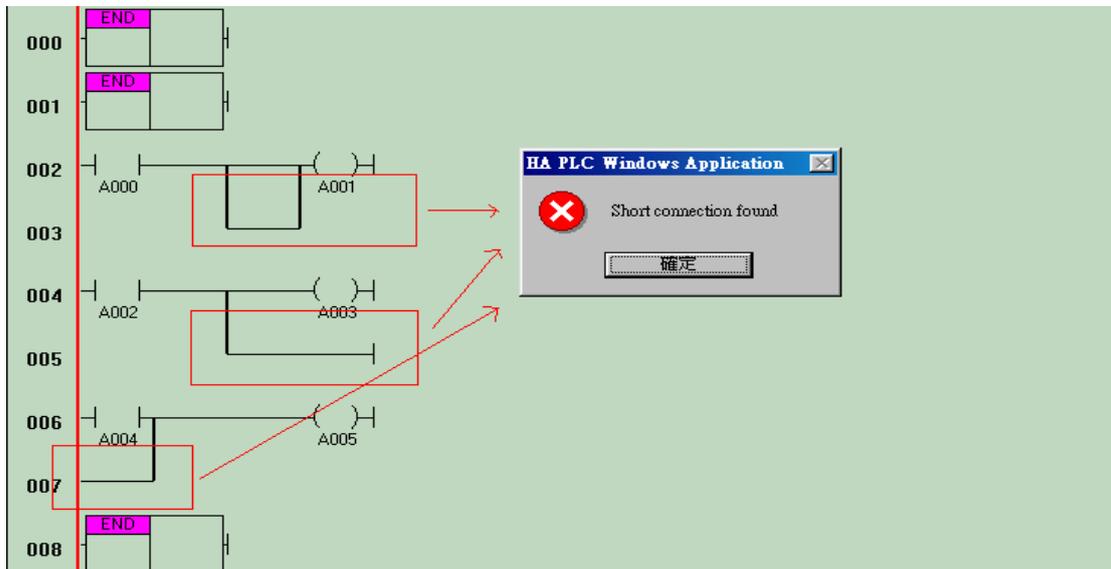


圖3.6 不應該有無作用的回路或線段部分

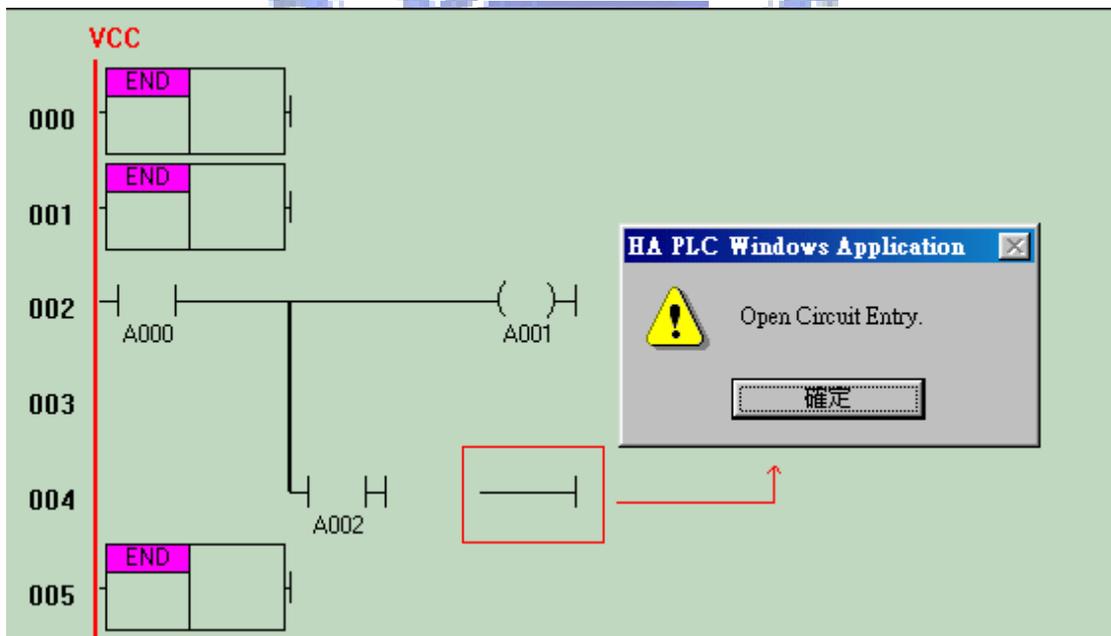


圖3.7 不應該有開路或斷線的部分

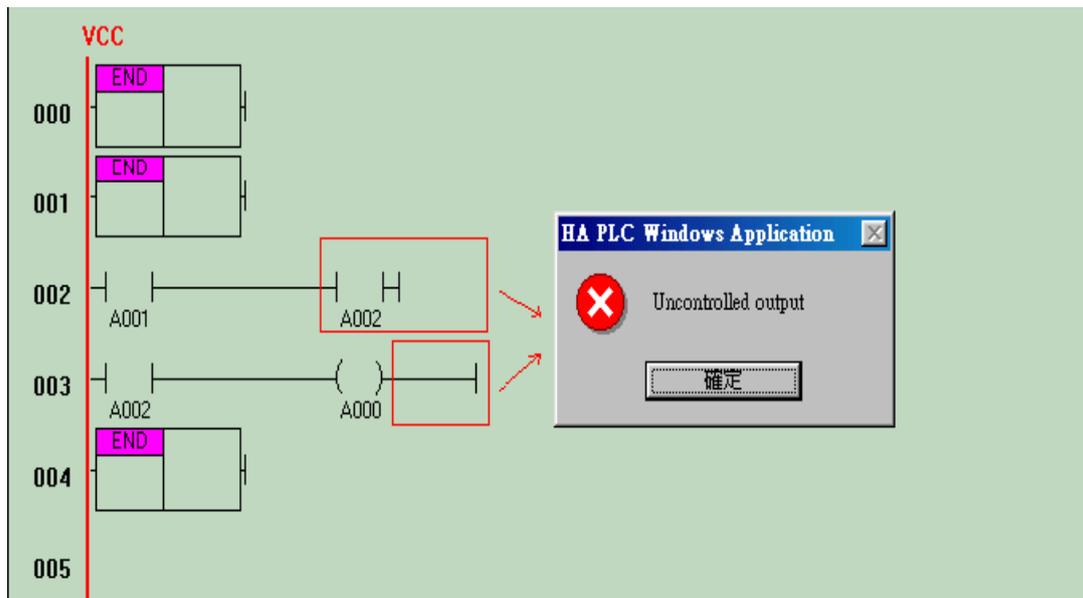


圖3.8 一個回路的最後一個節點不應該沒有輸出型式的節點

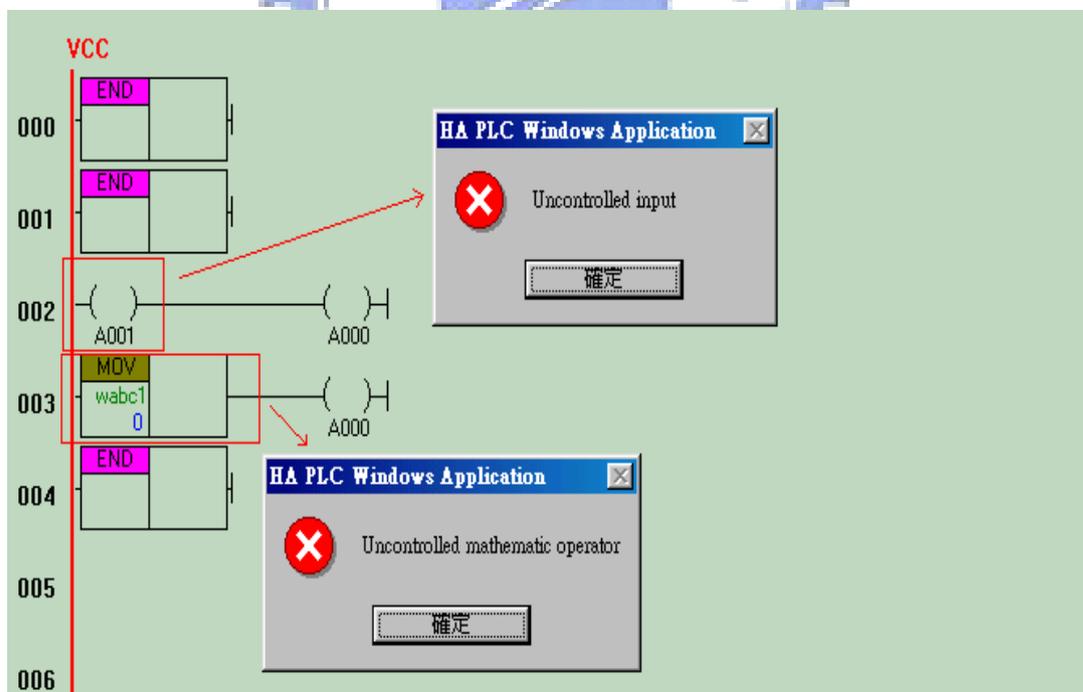


圖3.9 一行的第一個節點不應該是輸出型式的節點

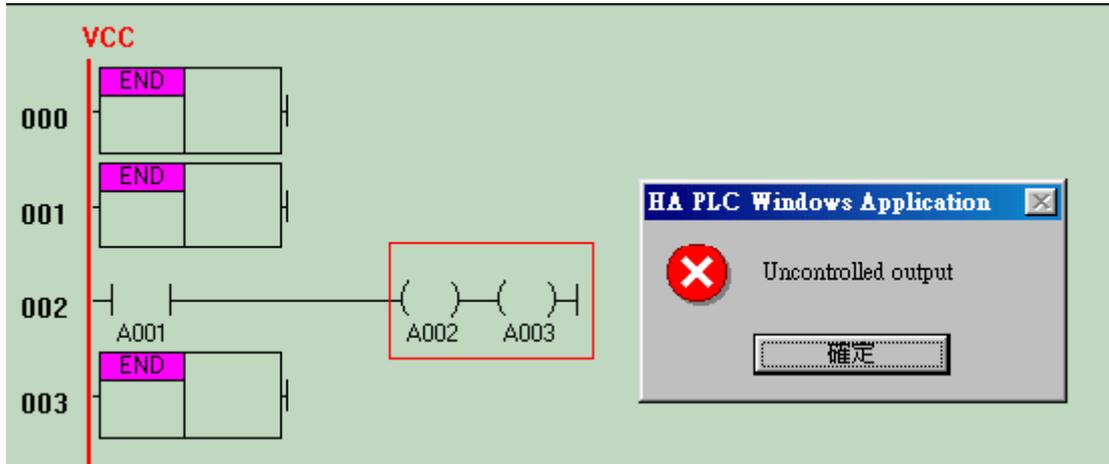


圖3.10 輸出線圈不能是串聯的型式

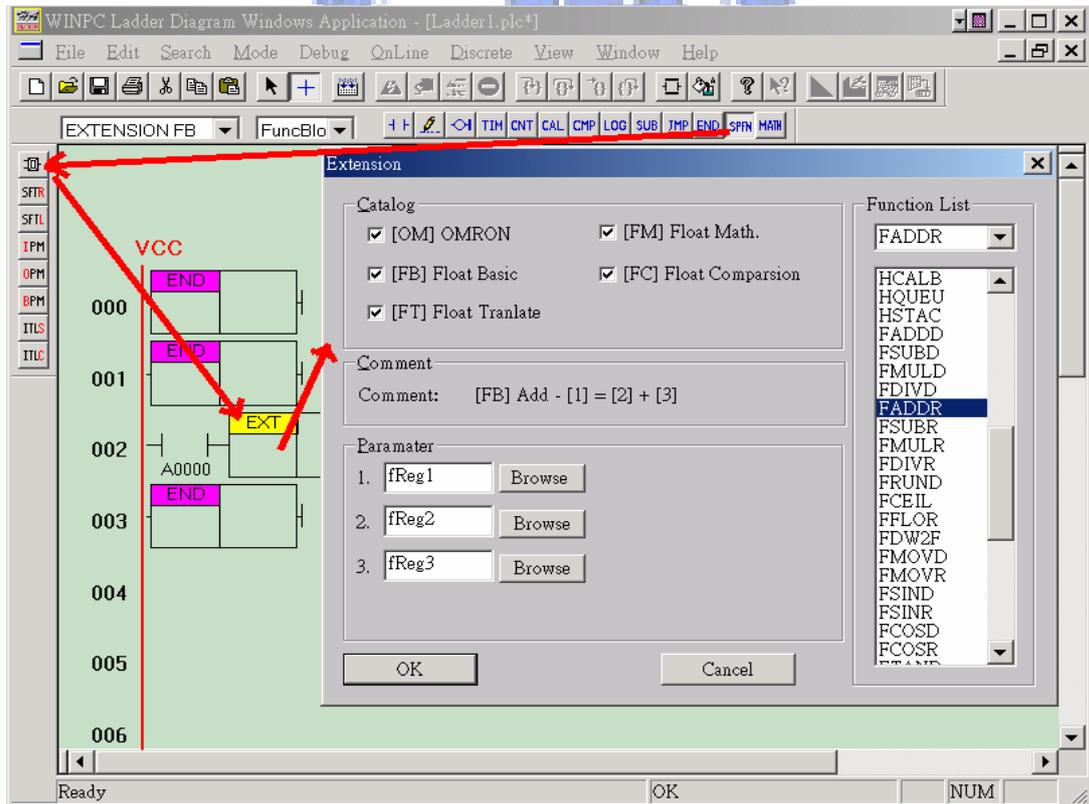


圖 3.11 擴充功能的對話窗

