

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	1 / 11

## 如何在 WP-5xx7 , VP-25W7 與 WP-8xx7 內使用 Micro\_SD 內的 file 來保存變數資料 ?

本文件與範例程式 ( wpdmo56.pia , wpdmo56a.pia , wpdmo56b.pia , wpdmo56c.pia , wpdmo56d.pia , wpdmo56e.pia ) 可於 [http://www.icpdas.com/faq/isagraf\\_c.htm](http://www.icpdas.com/faq/isagraf_c.htm) > FAQ-155 下載取得, 說明如下.

- wpdmo56 : 保存 17 個 實數 變數於 \Micro\_SD\data56.txt 內 (最多可保存 255 個 實數 )
- wpdmo56a : 保存 17 個 實數 與 2 個 BOOL 變數於 \Micro\_SD\data56.txt 內 (最多可保存 255 個 )
- wpdmo56b : 保存 25 個 整數 變數於 \Micro\_SD\data56.txt 內 (最多可保存 255 個 整數 )
- wpdmo56c : 保存 25 個 整數 與 2 個 BOOL 變數於 \Micro\_SD\data56.txt 內 (最多可保存 255 個 )
- wpdmo56d : 保存 17 個 實數 與 2 個 BOOL 變數 與 10 個整數 於 \Micro\_SD\data56F.txt 與 \Micro\_SD\data56.txt 內 (最多可保存 255 個 實數 與 “整數+BOOL” 255 個 )
- wpdmo56e : 最多可保存 1024 個 實數 與 最多 “整數+BOOL” 1024 個 .

本文件是說明如何將 WP-5xx7 , VP-25W7 與 WP-8xx7 這些 PAC 內 ISaGRAF 變數的值儲存於 Micro\_SD 內, 每次 PAC 一開機, 它就會自動讀出變數的值, 當變數的值有變更時, 也會自動把新的值存起來. 特別適用於 WP-5xx7 (比如 WP-5147), 因為它的標準品並沒有 “電池供應的 SRAM”, 若沒有購買 XW-608 板卡則 WP-5xx7 會無法使用 “新的可保留變數”, 此時就改使用本文件的方法.

- 注意:** 1. 請最好不要把資料 file 與 開發好的 .net 程式存放在 \System\_Disk 內, 因為 \System\_Disk 使用的是 Nor Flash 記憶體, 它主要是給 OS 與一些必要 utility 與 DLL 存放使用, size 不大. 而且 Nor Flash 記憶體 不適合常常去更新資料, 若常常在 \System\_Disk 內更新 file (比如每 1~5 秒就更新 file 一次, 那一天就更新了約幾萬次), 久了可能會損壞 \System\_Disk 內的資料. 所以自行開發的程式與要操作的 file 最好都存放在 \Micro\_SD 內 .
2. 在 PAC 的 \System\_Disk\ 或 \Micro\_SD\ 內 寫資料到檔案內 很費 CPU 時間, 必需避免每個 PLC Scan 都在 Read / Write 檔案 , 不然 PLC Scan Time 會變很大, 所以要保存的變數的值若一直常常變更, 則不適合用本文件的方法. 此類變數值會常常變動又需要保存下來的應用請參考 [http://www.icpdas.com/faq/isagraf\\_c.htm](http://www.icpdas.com/faq/isagraf_c.htm) 內的 FAQ-074 來改成使用 “新的可保留變數” 來快速保存資料 (WP-5xx7 必需額外購買一張 XW-608 卡片來插在 WP-5xx7 內才有支持 新的可保留變數” ).

範例程式 wpdmo56.pia , wpdmo56a.pia ~ wpdmo56e.pia 都會使用到變數陣列. 啟用它的方法如下, 一般是先把 PC 上的所有 ISaGRAF 畫面都先關閉, 然後在你的 PC / ISaGRAF 所安裝的路徑內, 比如 C:\ISAWIN\EXE\ 內開啟 isa.ini 這個 file, 然後在它的最上方加入以下 2 列, 儲存後, 再開啟 ISaGRAF 畫面, 之後就可使用變數陣列 , 更多說明,請參考 [http://www.icpdas.com/faq/isagraf\\_c.htm](http://www.icpdas.com/faq/isagraf_c.htm) > FAQ-039.

```
[DEBUG]
arrays=1
```

若您的 PC 上裝的 ISaGRAF 找不到 Msg\_F , Msg\_N , ARY\_F\_R , AFY\_F\_W 等函式, 請訪問 <http://www.icpdas.com/products/PAC/i-8000/isagraf.htm> > Driver 來下載 “ICP DAS utilities For ISaGRAF”, 解壓縮後, 執行其內的 setup.exe 來安裝新的 函式進去 ISaGRAF 內 (安裝大約花費 10 分鐘, 所以若已經有以上的函式, 就不需重新 install)

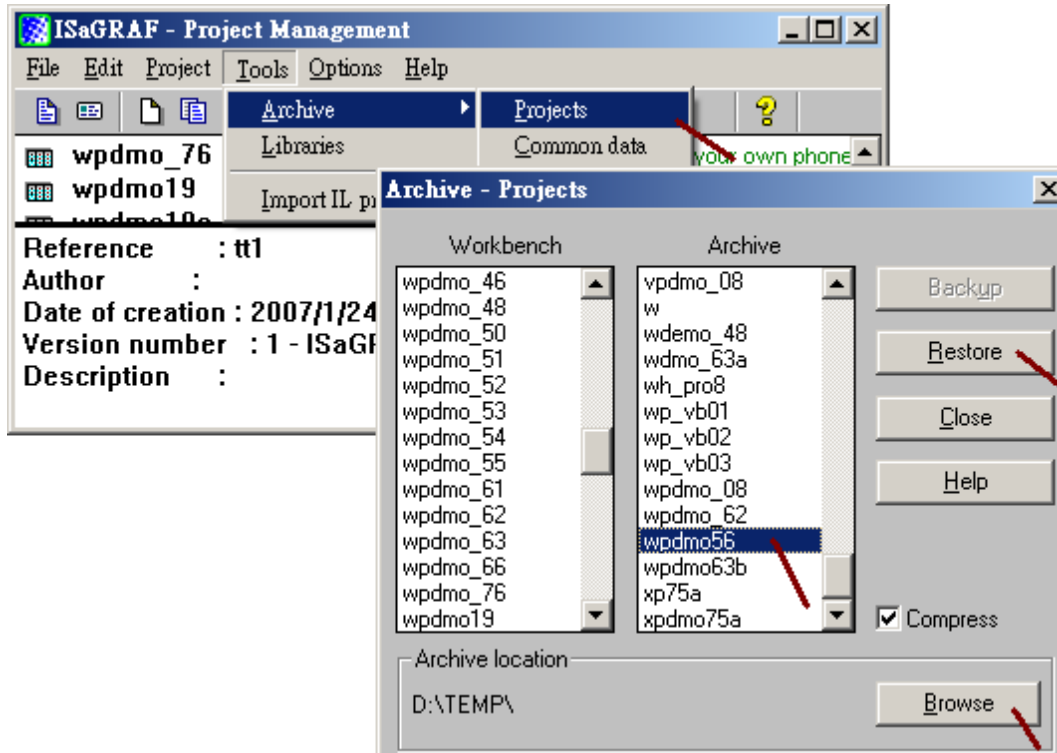
本文件是對 wpdmo56.pia 這個範例程式來說明, 一開機它會從 \Micro\_SD\data56.txt 內讀取 1 ~ 255

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	2 / 11

個實數變數的最終值，若該檔案不存在，則會指定每個變數的初值為 0.0. 之後只要任何一個數值有變，就會將全部的 1 ~ 255 個新的值存入 \Micro\_SD\data56.txt 內. 檔案若不存在，本範例程式會自動建一個新的 \Micro\_SD\data56.txt 檔案.

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	3 / 11

回存 範例程式 wpdmo56.pia 到 PC / ISaGRAF 內：

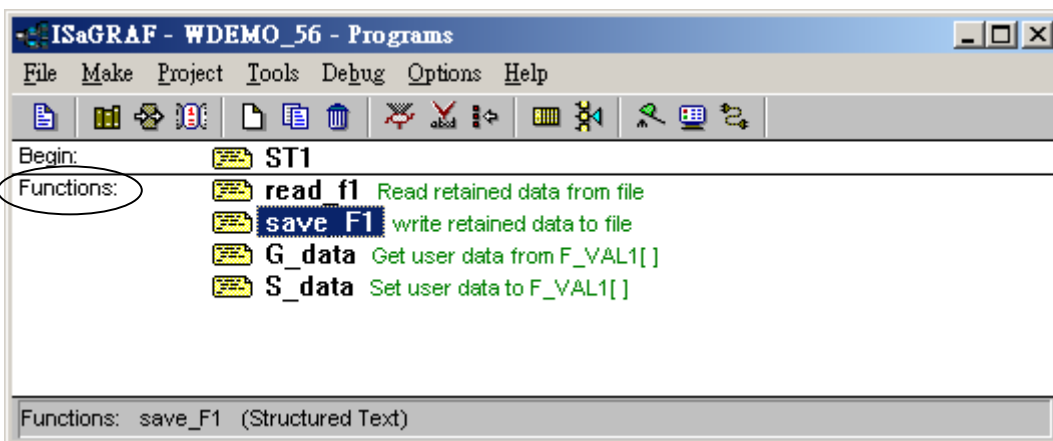


wpdmo56.pia 專案程式架構:

共有 5 個 ST 程式, 其中 reaf\_f1 , save\_f1 , G\_data 與 S\_data 為 functions

重要:

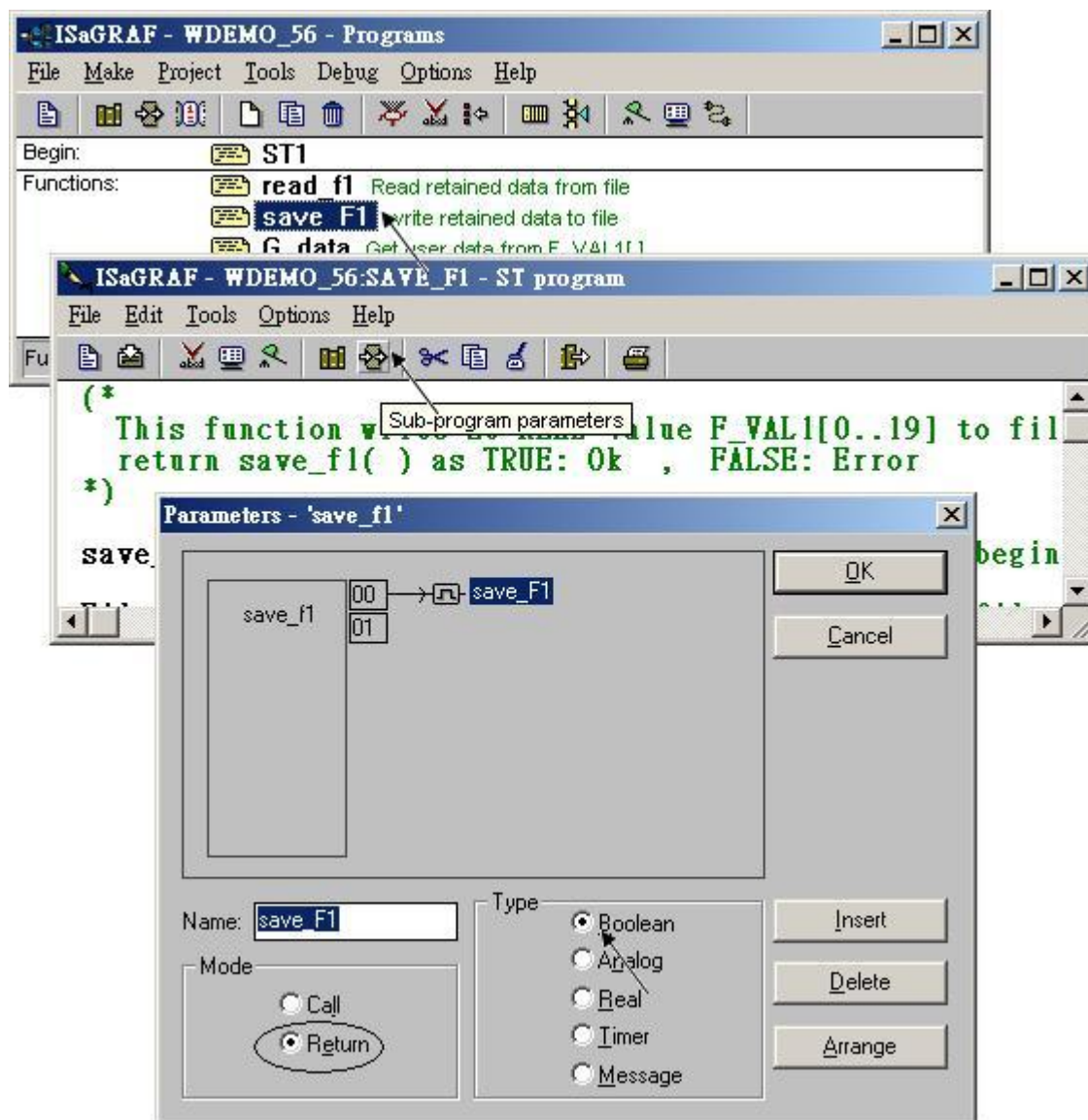
1. 本範例程式 可以依照實際應用狀況來修改 常數變數 **SIZE1** 之值為 1 到 255 之間的任一個值
2. 當 **SIZE1** 值有更改時, **F\_VAL1[]** 與 **Old\_F\_VAL1[]** 的 "Dim" 欄位也要改成同一個值, 同時, "G\_data" 與 "S\_data" 內的程式也需改成符合 User 需要的.
3. 使用 \Micro\_SD 來保存資料有個優點為,這些資料的 File 可以預先在 PC 上編輯好, 再用 ftp 丟到 PAC 內, 本例為 \Micro\_SD\data56.txt. 之後將 RE\_LOAD 變數設為 TRUE 一次, 它就會自動更新到 USER 的應用變數內.



Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	4 / 11

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	5 / 11

本例, read\_f1 , save\_f1 , G\_data 與 S\_data 皆為 functions, 其回傳值皆定義為 Boolean 型態, 要宣告 functions 回傳值的型態, 如下 ( “ISaGRAF 進階使用手冊” 的第 15 章有詳細說明)



Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	6 / 11

read\_f1 與 save\_f1 內有使用 區域變數 (Local variable):

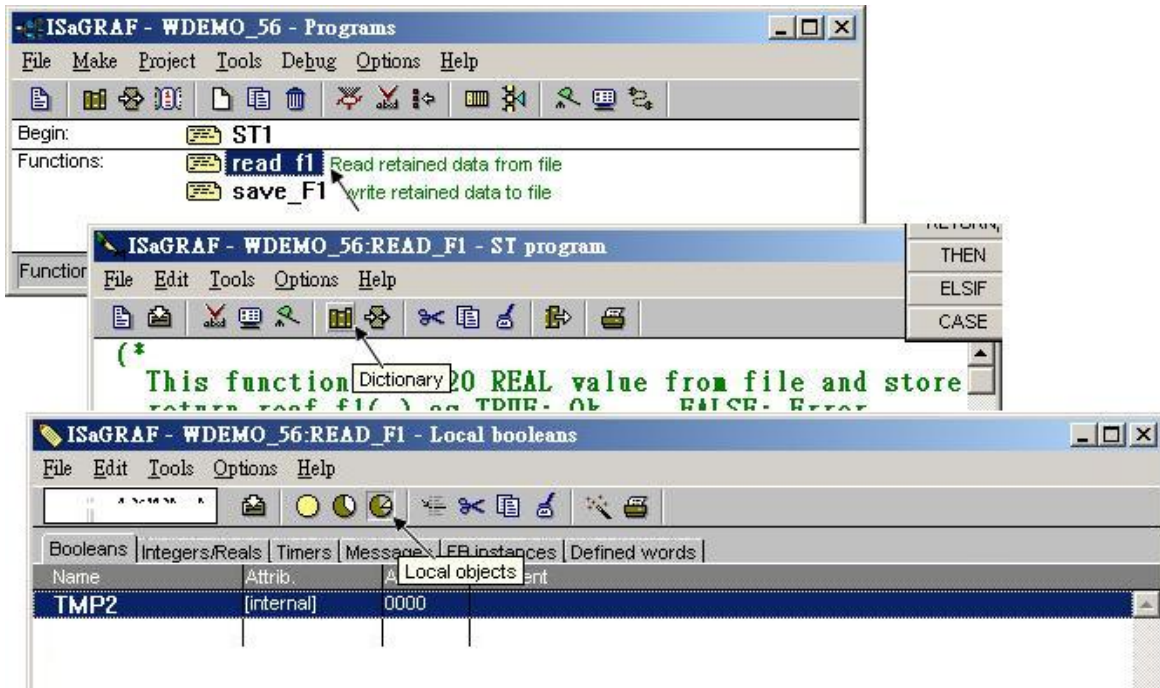
read\_f1 程式的 區域變數 (Local variable) 如下:

<i>Name</i>	<i>Type</i>	<i>Attribute</i>	<i>Description</i>
TMP2	Bool	Internal	暫時使用的 Boolean 變數
ii2	Integer	Internal	給 for 迴圈使用的 index
jj2	Integer	Internal	給 for 迴圈使用的 index
num2	Integer	Internal	暫時使用的 Integer 變數

save\_f1 程式的 區域變數 (Local variable) 如下:

<i>Name</i>	<i>Type</i>	<i>Attribute</i>	<i>Description</i>
TMP2	Bool	Internal	暫時使用的 Boolean 變數
ii2	Integer	Internal	給 for 迴圈使用的 index
jj2	Integer	Internal	給 for 迴圈使用的 index
num2	Integer	Internal	暫時使用的 Integer 變數

要宣告區域變數, 先雙擊 read\_f1 程式進入, 之後 進入 Dictionary 內, 之後點選 “Local objects” 開始宣告 區域變數



Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	7 / 11

全域變數 (Global variable)

<i>Name</i>	<i>Type</i>	<i>Attribute</i>	<i>Description</i>
SIZE1	Integer	Constant 常數	User 要保存的資料數量, 可以是 1 ~ 255 若改變此值, 需一併更改 F_VAL1[ ] 與 Old_F_VAL1[ ] 內宣告的 Dim 欄位為相同值 <b>本例設 SIZE1 為 17 (請改成你要使用的數量)</b>
num_row1	Integer	Internal	檔案內資料共幾行, 此值由程式用 SIZE1 自動算出 本範例一行可放 10 個實數資料
Last_num1	Integer	Internal	檔案內最後一行資料的 資料有幾個 此值由程式用 SIZE1 自動算出
RE_LOAD	Bool	Internal	宣告初值為 TRUE, 所以一開機會讀 File 一次 若程式運行中 又設此值為 TRUE, 會再去讀 File 一次
TMP	Bool	Internal	暫時使用的 Boolean 變數
Data_Ok1	Bool	Internal	TRUE 表是讀 File Ok
Flag_to_save	Bool	Internal	Controller 要存 File 時, 會自動設它為 TRUE
File_name1	Message	Internal	長度 64, 初值為 \Micro_SD\data56.txt
Msg1	Message	Internal	長度 128, 會顯示 File 處理狀態
str1	Message	Internal	長度 255, 操作 File 字串時會用到
F_VAL1[0..19]	REAL	Internal	為變數陣列, Dim 欄位需設成與 SIZE1 之值相同
Old_F_VAL1 [0..19]	REAL	Internal	為變數陣列, Dim 欄位需設成與 SIZE1 之值相同, 為 F_VAL1[ ] 的舊值
NUM1	Integer	Internal	接收 Msg_F( ) 的回傳值, 若為 -1 表示格式錯誤
File1	Integer	Internal	File ID
ii	Integer	Internal	給 for 迴圈使用的 index
jj	Integer	Internal	給 另一個 for 迴圈使用的 index
Data1 ~ Data5 與 Data06 ~ Data17	REAL	Internal	用來模擬 (仿真) 為 User Data 變數, 本例因為 SIZE1 值為 17, 所以共有 17 個資料變數 User 實際的應用可以每個都使用不同的變數名稱. 若有更改名稱, G_data 與 S_data 內程式也要修改

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	8 / 11

ST 程式 ST1:

```

if RE_LOAD then (* 一開機, RE_LOAD 初值為 TRUE, 會去讀 File 一次 *)

    RE_LOAD := FALSE ; (* 進來之後需馬上將 RE_LOAD 設為 FALSE *)

    (* 根據 SIZE1 之值, 計算出 共有幾行資料, 與最後一行有幾個 REAL 資料 *)
    num_row1 := SIZE1 / 10 ;
    last_num1 := SIZE1 - 10 * num_row1 ;
    if last_num1 <> 0 then
        num_row1 := num_row1 + 1 ;
    else
        last_num1 := 10 ;
    end_if ;

    TMP := read_F1( ) ; (* 呼叫 read_f1( ) 來讀 SIZE1 個 資料到 F_VAL1[ ] 內 *)

    if TMP = FALSE then (* read_f1( ) 若回傳 FALSE, 表示讀取失敗 *)

        for ii := 0 to SIZE1 - 1 do
            F_VAL1[ii] := 0.0 ; (* 讀取失敗將 SIZE1 個初值設為 0.0 *)
        end_for ;

        Data_Ok1 := FALSE ; (* 將 Data_Ok1 設為 FALSE 來表示 “讀取失敗” *)
        Msg1 := 'File : ' + File_name1 + ' not exist or data error ! or File is open now' ;

    Else (* 若讀取 File 成功 *)

        Data_Ok1 := TRUE ; (* 將 Data_Ok1 設為 TRUE 來表示 “讀取成功” *)
        Msg1 := 'Get Retained data from file Ok' ;

    end_if ;

    (* 一開機時 要更新 Old_F_VAL1[ ] 之值 與 F_VAL1[ ] 相同 *)
    for ii := 0 to SIZE1 - 1 do
        Old_F_VAL1[ii] := F_VAL1[ii] ;
    end_for ;

    TMP := G_data( ) ; (* 每次從 File 內讀資料後 , 需更新 值 到 User Data 變數內*)

end_if ;

```



Classification	ISaGRAF Chinese FAQ-155							
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	9 / 11	

(\* 每個 PLC Scan 都需 把 User Data 變數值 更新到 F\_VAL1[ ] 內 \*)

TMP := S\_data( ) ;

(\* 在每個 PLC scan 內判斷 這 SIZE1 個資料是否有任一個值有更改 \*)

for ii := 0 to SIZE1 - 1 do

if Old\_F\_VAL1[ii] <> F\_VAL1[ii] then (\* 新/舊值不同, 表示有更改過 \*)

Flag\_to\_save := TRUE ; (\* 將 Flag 設為 TRUE 來準備要 寫資料到 File 內\*)

Old\_F\_VAL1[ii] := F\_VAL1[ii]; (\* 值不同時, 要更新 舊值 \*)

end\_if ;

end\_for ;

if Flag\_to\_save then (\* 若 Flag 被設為 TRUE, 寫 SIZE1 個資料到 File 內 \*)

TMP := save\_fl( ) ; (\* 呼叫 save\_fl() 來寫 資料 \*)

if TMP = FALSE then (\* save\_fl() 回傳 FALSE, 表示寫 File 失敗,可能 File 被打開,沒關 \*)

Msg1 := 'Can not save data to file. May be file is open now by Wincon screen !' ;

Else (\* 寫 File 成功, 將 Flag 清除為 FALSE \*)

Flag\_to\_save := FALSE ;

end\_if ;

end\_if ;

Classification	ISaGRAF Chinese FAQ-155							
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	10 / 11	

ST functions 程式 - G\_data :

(\* 更新 User Data 變數之值 ,若 SIZE1 值有改, 或 User Data 變數名稱有改, 以下的 Code 也要修改 \*)

```

Data1 := F_VAL1[0];
Data2 := F_VAL1[1];
Data3 := F_VAL1[2];
Data4 := F_VAL1[3];
Data5 := F_VAL1[4];
Data06 := F_VAL1[5];
Data07 := F_VAL1[6];
Data08 := F_VAL1[7];
Data09 := F_VAL1[8];
Data10 := F_VAL1[9];
Data11 := F_VAL1[10];
Data12 := F_VAL1[11];
Data13 := F_VAL1[12];
Data14 := F_VAL1[13];
Data15 := F_VAL1[14];
Data16 := F_VAL1[15];
Data17 := F_VAL1[16];
G_data := TRUE ; (* function 回傳 TRUE *)

```

ST functions 程式 - S\_data :

(\* 更新 F\_VAL1[ ] 之值 , 若 SIZE1 值有改, 或 User Data 變數名稱有改, 以下的 Code 也要修改 \*)

```

F_VAL1[0] := Data1;
F_VAL1[1] := Data2;
F_VAL1[2] := Data3;
F_VAL1[3] := Data4;
F_VAL1[4] := Data5;
F_VAL1[5] := Data06;
F_VAL1[6] := Data07;
F_VAL1[7] := Data08;
F_VAL1[8] := Data09;
F_VAL1[9] := Data10;
F_VAL1[10] := Data11;
F_VAL1[11] := Data12;
F_VAL1[12] := Data13;
F_VAL1[13] := Data14;
F_VAL1[14] := Data15;
F_VAL1[15] := Data16;
F_VAL1[16] := Data17;
S_data := TRUE ; (* function 回傳 TRUE *)

```

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	11 / 11

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	12 / 11

ST functions 程式 read\_f1 :

```

-----
(* 這個 function 從 File 內讀出 SIZE1 個實數並存入 F_VAL1[ ]內
   成功: read_f1() 回傳 TRUE , 失敗回傳 FALSE *)
read_f1 := FALSE ; (* 一開始先預設為失敗 *)
File1 := f_wopen( File_name1 ) ; (* 開啟 File 為 可讀 / 可寫 模式 *)

if File1 = 0 then (* 開啟失敗, 通常為 File 不存在 *)
  return ; (* 離開此程式 *)
end_if ;

(* File 開啟成功 , 讀資料 *)
for ii2 := 0 to num_row1 - 1 do (*共 num_row1 行, 每行 10 個 REAL 值 *)

  if f_eof( File1 ) = TRUE then (* 偵測是否抵達檔案尾端 *)
    exit ; (* 若是則離開 for 迴圈 *)
  end_if ;

  str1 := fm_read( File1 ) ; (* 從 File 內讀出一行字串 *)
  NUM1 := Msg_F( str1 , 1 ) ; (* 將該字串轉換為數個 REAL 值 , 並存放於 1 號 Float 陣列內 *)

  (* 若為 最後一行資料數量是否正確 ? 若非最後一行, 資料數量是否為 10 個 ? *)
  if ( ( ii2 = num_row1 - 1 ) and ( NUM1 <> last_num1 ) ) or
     ( ( ii2 <> num_row1 - 1 ) and ( NUM1 <> 10 ) ) then
    exit ; (* 數量不對, 離開 for 迴圈 *)
  end_if ;

  if ii2 = num_row1 - 1 then (* 若為最後一行, 設 num2 為 最後一行的資料數量 *)
    num2 := last_num1 ;
  else (* 若不是最後一行, 設資料數量 num2 為 10 個 *)
    num2 := 10 ;
  end_if ;

  (* 轉換成功, 將此行的 REAL 值 存入 F_VAL1[ ] 內 *)
  for jj2 := 0 to num2 - 1 do
    F_VAL1[ 10*ii2 + jj2 ] := ARY_F_R( 1 , jj2 + 1 ) ;
  end_for ;
end_for ;

TMP2 := f_close( File1 ) ; (* File 只要成功開啟過, 處理完就需 close *)
If ii2 = num_row1 then
  read_F1 := TRUE ; (* num_row1 行資料都已讀到, read_f1() 回傳 TRUE *)
end_if ;
-----

```

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	13 / 11

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	14 / 11

ST functions 程式 save\_f1 :

```

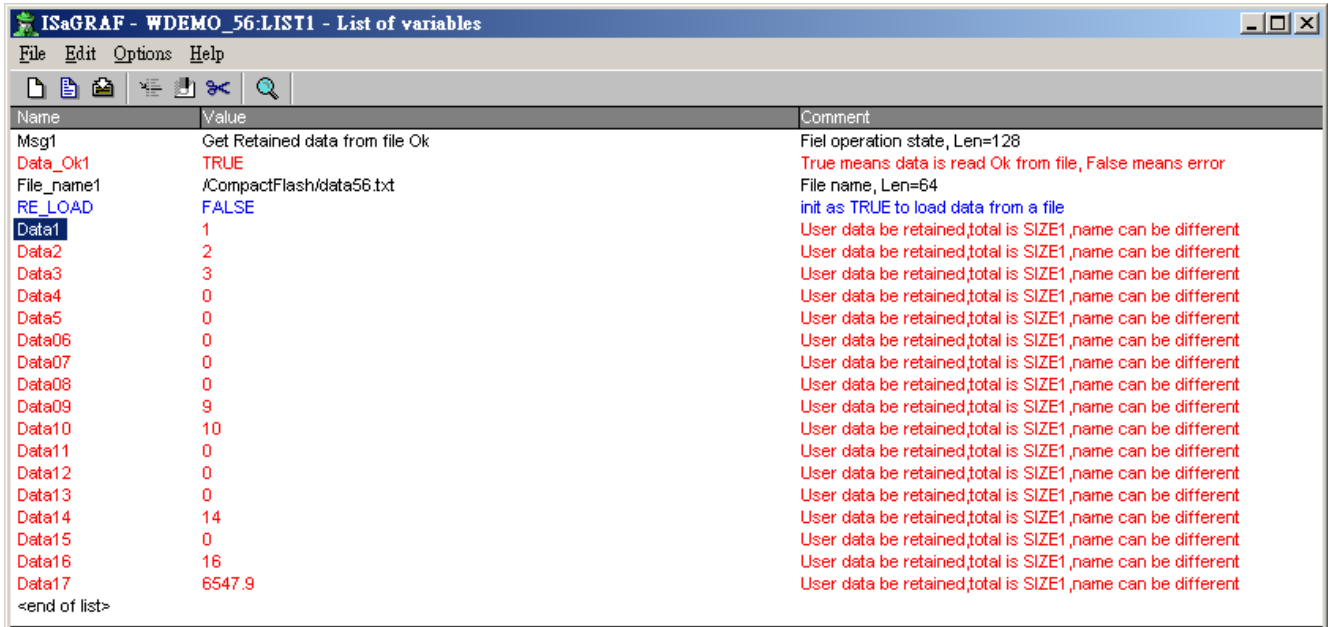
-----
(* 這個 function 寫 SIZE1 個實數 F_VAL1[ ] 到 File 內,
   成功: save_f1() 回傳 TRUE , 失敗回傳 FALSE *)
save_f1 := FALSE; (* 一開始先預設為失敗 *)
File1 := f_creat( File_name1 ) ; (* 建立一個新 File, 若檔案已存在, 資料會刪除 *)
if File1 = 0 then
    return ; (* 建立一個新 File 失敗, 離開此程式 *)
end_if ;
(*建立一個新 File 成功, 準備寫資料進去 *)
for ii2 := 0 to num_row1 - 1 do (* 共 num_row1 行資料要寫入 *)
    str1 := ' ' ; (* 設每行字串初值為 1 個空格 *)
    if ii2 = num_row1 - 1 then (* 若為最後一行, 設 num2 為 最後一行的資料數量 *)
        num2 := last_num1 ;
    else (* 若不是最後一行, 設資料數量 num2 為 10 個 *)
        num2 := 10 ;
    end_if ;
    (* 將 REAL 資料 寫成 字串格式 *)
    for jj2 := 0 to num2 - 2 do (* 不含最後一行 *)
        str1 := str1 + REAL_STR( F_VAL1[ 10 * ii2 + jj2 ] ) + ',' ;
    end_for ;
    (* 最後一行, 結尾需為 <CR><LF> 字元 *)
    str1 := str1 + REAL_STR( F_VAL1[ 10 * ii2 + 9 ] ) + '$0D$0A' ;
    TMP2 := f_writ_s( File1 , str1 ) ; (* 將此行字串寫入 File 內 *)
end_for ;
TMP2 := f_close( File1 ) ; (* File 只要成功開啟過, 處理完就需 close *)
save_f1 := TRUE; (* num_row1 行資料都已寫入, save_f1() 回傳 TRUE *)
-----

```

Classification	ISaGRAF Chinese FAQ-155						
Author	Chun Tsai	Version	1.2	Date	Jul.2012	Page	15 / 11

本 wpdmo\_56 範例如何測試 ?

1. 將 wpdmo\_56 用 ISaGRAF 下載到 WP-5xx7 後 (或 VP-25W7 , VP-23W7, WP-8xx7), 會出現 Spy list 視窗如下



請更改 任何一個 USER Data 變數之值, 改了後, 它就會自動存入 \Micro\_SD\data56.txt 內, 所以你可以在 WP-5xx7 連接的 VGA 的螢幕上開啟 此檔, 會發現它的資料有跟著變動. (請不要一直開著此檔, 要關掉它, 不然之後 資料有變它會寫不進去, 但只要有關掉, 資料就會更新)

2. 將 WP-5xx7 關機後 約 5 秒 再開機, 開完機後您會看到 這些 USER Data 變數之值 會是您最後輸入給它的值.
3. 在 PC 上用 NotePad (記事本) 編輯一個 data56.txt 檔類似如下,

1.1 , 2.2 , 3.3 , 4.4 , 5.5 , 6.66 , 7.77 , 8.88 , 9.99 , 10.01  
0.01 , 0.02 , 0.03 , 0.04 , 0.05 , 0.06 , 0.07

將此 data56.txt 檔用 ftp 傳到 WP-5xx7 的 \Micro\_SD\ 目錄 內. 之後在 ISaGRAF 的 Spy list 視窗, 雙擊 RE\_LOAD, 把它的值設為 TRUE. 您會發現 USER Data 變數之值 會更新成 上面的值.