

如何使用 ISaGRAF New Retain Variable 與 它的優點？

i-8417/8817/8437/8837 , I-7188EG/XG , Wincon-8xx7 從以下驅動程式版本起, 支持新的可保存變數方式.

I-7188EG + X607 / X608:	2.05 版起, 建議採用	2.17 版 或 更高的版本
I-7188XG + X607 / X608:	2.04 版起, 建議採用	2.15 版 或 更高的版本
I-8xx7+ S256 / S512 :	3.07 版起, 建議採用	3.19 版 或 更高的版本

W-8347/8747/8337/8737 + S256 / S512 : 3.18 版起, 建議採用 3.36 版 或 更高的版本
要使用以下的 Wincon 新底板(back-plane)

W-8346/8746/8336/8736 + S256 / S512 : 3.18 版起, 建議採用 3.36 版 或 更高的版本
要使用以下的 Wincon 新底板

WB-831 (3 槽): Rev 2.6 版起 (2006 年起出貨用的底板)

WB-871 (7 槽): Rev 2.8 版起 (2006 年起出貨用的底板)

假如有插 電池供應的 SRAM 於控制器的背板上 (I-8xx7: S256/S512, I-7188EG/XG: X607/X608, Wincon-8xx7: S256/S512), 最大的可保存變數數量為如下所示. 新的可保存變數方式是由以下的 ISaGRAF 函式來達成的

控制器 1 : I-7188EG/XG+X607/608, I-8417/8817/8437/8837+S256/512

控制器 2 : W-8xx7 + S256/512 (使用 Wincon 2006 年起出貨用的 新底板)

Retain_B : 設定 Boolean 可保存變數 控制器 1: 最多 256 個變數, 控制器 2: 最多 1024 個

Retain_N : 設定 Integer 可保存變數 控制器 1: 最多 1024 個變數, 控制器 2: 最多 4096 個

Retain_F : 設定 Real 可保存變數 控制器 1: 最多 1024 個變數, 控制器 2: 最多 4096 個

Retain_T : 設定 Timer 可保存變數. 控制器 1: 最多 256 個變數, 控制器 2: 最多 1024 個

Retain_X : 使用變數的 Network address 來設定可保存變數

Retain_A : 使用變數的 Network address 來設定可保存變數陣列 (請參考第 2.6.2 節)

新的可保存變數的優點:

1. 值會一直保存, 不論是 ” 控制器斷電 ” 或 ” 重新 compile ” 與 ” 重新下載 ISaGRAF 程式 ”, 除非是 S-256/512, X607/608 上面的電池沒電了. (舊方式碰到 ” 重新下載 ISaGRAF 程式 ”, 值會清為 0)
2. 新的可保存變數 可保存的變數量 較舊方式多.

請參考下一頁的範例程式來實作新的可保存變數.

注意:

1. 要使用新的可保存變數, 請 **不要** 選取 字典視窗內的變數宣告內的 “可保留” (Retain) 選項.
2. 如果是使用 **I-7188EG/XG**, 要先在 IO 連結視窗內連上 “**X607_608**”.



程式範例: (* 使用 變數名稱 來設定 為 可保存變數*)

(* 設定變數為 Retain variable 的動作, 一定要在第一個 PLC scan 內 只做一次, 如下方的 code *)

(* To_Retain 宣告為 Boolean, Internal, 初值設為 True *)

(* Tmp 宣告為 Boolean, Internal *)

(*

Check1 宣告為 Integer, Internal, 不可勾選 “可保留” (Retain), 此變數用來 當作 一個 標記.

比如可以定義 若值為 1357246, 表示 所有該指定初值的 Retain Variable, 初值都已經設定過了.

那就可讓 本應用的 控制程式開始動作 (因為該給的值已經設好了).

但若 Check1 的值不是 1357246, 表示 所有該指定初值的 Retain Variable, 還有一些沒設好初值.

那就 不能讓 本應用的 控制程式動作 (因為該給的值, 還沒設好, 若讓控制程式動作, 會出錯, 必需要等 User 把該設的 Retain 值設好後, 之候再設 Check1 為 1357246, 讓控制程式開始動作. 日後每次重新開機, 因為是 Retain Variable, 所以以前設過的值 都還在, 包含 Check1 也會是 1357246, 所以程式開機後就可動作, 不需再重新設 Retain 的初值, 因為以前已經至少設過一次了.)

*)

(* B1, B2 宣告為 Boolean, Internal, 不可勾選 “可保留” (Retain) *)

(* N1, N2 宣告為 Integer, Internal, 不可勾選 “可保留” (Retain) *)

(* F1, F2 宣告為 Real, Internal, 不可勾選 “可保留” (Retain) *)

(* T1, T2 宣告為 Timer, Internal, 不可勾選 “可保留” (Retain) *)

if To_Retain then (* 當控制器剛啟動時, 使用變數名稱來設定可保存變數 *)

To_Retain := False ; (* 設為 False, 如此只有第 1 個 PLC Scan 會執行到這一段 *)

Tmp := Retain_N(Check1, 1); (* 此變數用來 當作 一個 標記 *)

Tmp := Retain_B(B1, 1); **Tmp := Retain_B(B2, 2);** (* Boolean *)

Tmp := Retain_N(N1, 11); **Tmp := Retain_N(N2, 12);** (* Integer *)

Tmp := Retain_F(F1, 1); **Tmp := Retain_F(F2, 2);** (* Real *)

Tmp := Retain_T(T1, 1); **Tmp := Retain_T(T2, 2);** (* Timer *)

end_if ;

(* 如此之後 Check1, B1, B2, N1, N2, F1, F2, T1, T2 的值就會一值被保存下來 *)

接下來 不管是寫 Ladder program 或 ST 都可判斷 Check1 這個標記值是否正確, 再 開始做控制動作.

(用意再於讓 User 把 該設的 Retain 值, 至少設過一次初值, 然後才去設 Check1 為 1357246, 讓 以下的控制程式開始動作)

