

How to use ISaGRAF New Retain Variable ? What is its advantage ?

The I-8417/8817/8437/8837 , I-7188EG/XG , Wincon-8x37/8x36 , Wincon-8x47/8x46 , WP-8xx7, WP-8xx6, VP-2xW7, VP-2xW6, XP-8xx7-CE6, XP-8xx6-CE6, iP-8xx7, uPAC-7186EG, VP-2117 supports new retain function.

I-7188EG + X607 or X608: driver ver. 2.05 or later (better to be 2.17 or later)

I-7188XG + X607 or X608: driver ver. 2.04 or later (better to be 2.15 or later)

I-8xx7+ S256 or S512 : driver ver. 3.07 or later (better to be 3.19 or later)

uPAC-7186EG + X607 or X608: since released

iP-8xx7 (built-in battery backup SRAM) : since released

WP-8xx7 / 8xx6 (built-in battery backup SRAM) : since released

VP-2xW7/2xW6 (built-in battery backup SRAM) : since released

XP-8xx7-CE6/XP-8xx6-CE6 (built-in battery backup SRAM) : since released

VP-2117 (built-in battery backup SRAM) : since released

W-83x7/83x6+ S256 or S512 : driver ver. 3.18 or later (better to be 3.36 or later) with new back-plane of WB-831 (For 3-slot): Rev 2.6 (delivered since 2006)

W-87x7/87x6+ S256 or S512 : driver ver. 3.18 or later (better to be 3.36 or later) with new back-plane of WB-871 (For 7-slot): Rev 2.8 (delivered since 2006)

The iP-8xx7 , WP-8xx7, XP-8xx7-CE6,VP-25W7/23W7 and VP-2117 have built in 512K byte battery SRAM when shipped out (They don't need S-256/512).

If battery backup SRAM is found in the backplane of the controller (uPAC-7186EG:X607/608, I-8xx7: S256/S512, I-7188EG/XG: X607/X608, Wincon-83x7/87x7/83x6/87x6: S256/S512), the maximum number of retained variables for new retain function are listed as below. New retain variable is supported by below ISaGRAF “C-function”

Target 1 : uPAC-7186EG/I-7188EG/XG+X607/608, I-8417/8817/8437/8837+S256/512 and iP-8xx7

Target 2 : Wincon+S256/512(with new Wincon back-plane) and WP-8xx7, VP-2xW7, XP-8xx7-CE6

Retain_B : retain Boolean variable. Target 1: max. 256 variables, Target 2: max. 1024.

Retain_N : retain Integer variable. Target 1: max. 1024 variables, Target 2: max. 4096.

Retain_F : retain Real variable. Target 1: max. 1024 variables, Target 2: max. 4096.

Retain_T : retain Timer variable. Target 1: max. 256 variables, Target 2: max. 1024.

Retain_X : retain variable by using its Network address

The following ST program can read the state of the two batteries on the backplane of iP-8xx7, WP-8xx7, VP-25W7/23W7 and XP-8xx7-CE6. Please declare battery_state1 and battery_state2 as internal integer.

```
battery_state1 := R_MB_ADR(1 , 9992) ; (* 0: low power , 99: power Ok *)
```

```
battery_state2 := R_MB_ADR(1 , 9993) ; (* 0: low power , 99: power Ok *)
```


Sample program:

(* These Retain_? Functions should be called only once in the first PLC scan cycle as below code *)

(* To_Retain is declared as an internal boolean variable with initial value as TRUE *)

(* Tmp is declared as an internal boolean variable *)

(*

Check1 is declared as internal Integer variable, Do not check "Retain" in it in the dictionary windows. This "Check1" variable is for identification if the initial value of the retained variables were well set or not. For example, we can define if "Check1" value is 1357246, it means all initial value of retained variables are well set. Then the control program can go. However if "Check1" is not 1357246, it means some initial value of retained variables haven't been set. Then the control program CAN NOT go. Because it will generate error (some retained value are not set yet) .

So user has to set necessary initial value of the retained variables once at least . Then set "Check1" as 1357246.

*)

(* B1 , B2 is declared as internal Boolean variable, Do not check "Retain" *)

(* N1 , N2 is declared as internal Integer variable, Do not check "Retain" *)

(* F1 , F2 is declared as internal Real variable, Do not check "Retain" *)

(* T1 , T2 is declared as internal Timer variable, Do not check "Retain" *)

(* To set retained variables when controller is start running *)

if To_Retain then

To_Retain := False ; (* Only do it once *)

Tmp := Retain_N(Check1 , 1) ; (* for identification *)

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 ;

(* The following ST program can read the state of the two batteries on the backplane of iP-8xx7, WP-8xx7, VP-25W7/23W7 and XP-8xx7-CE6. Please declare battery_state1 and battery_state2 as internal integer. *)

battery_state1 := R_MB_ADR(1 , 9992) ; (* 0: low power , 99: power Ok *)

battery_state2 := R_MB_ADR(1 , 9993) ; (* 0: low power , 99: power Ok *)

(* After then Check1 , B1, B2, N1, N2, F1, F2, T1, T2 will be automatically retained in the program *)

