

Remanent variables

Remanent variables can retain their value throughout the usual program run period. These include Retain variables and Persistent variables.

Example:

```
VAR RETAIN
  iRem1:INT; (* 1. Retain variable*)
END_VAR
```

- Retain variables are identified by the keyword **RETAIN**. These variables maintain their value even after an uncontrolled shutdown of the controller as well as after a normal switch off and on of the controller (corresponding to command '[Online](#)' '[Reset](#)'). When the program is run again, the stored values will be processed further. A concrete example would be an piece-counter in a production line, that recommences counting after a power failure. All other variables are newly initialized, either with their initialized values or with the standard initializations. Retains are reinitialized at '[Reset \(cold\)](#)' and '[Reset \(original\)](#)' and - contrary to Persistent variables - at a new [download](#) of the program.
- Persistent variables are identified by the keyword **PERSISTENT**. Unlike Retain variables, these variables retain their value only after a re-[Download](#), but not after an '[Online](#)' '[Reset](#)', '[Online](#)' '[Reset \(original\)](#)' or '[Online](#)' '[Reset \(cold\)](#)', because they are not saved in the "retain area". If also persistent variables should maintain their values after a [uncontrolled shutdown](#) of the controller, then they have to be declared additionally as VAR RETAIN variables. A concrete example of "persistent Retain-Variables" would be a operations timer that recommences timing after a power failure.

x = value will be retained - = value gets reinitialized

after Online command	VAR	VAR RETAIN	VAR PERSISTENT	VAR RETAIN PERSISTENT VAR PERSISTENT RETAIN
Reset	-	x	-	x
Reset cold	-	-	-	-
Reset origin	-	-	-	-
Download	-	-	x	x
Online Change	x	x	x	x

Attention:

- If a local variable in a **program** is declared as VAR RETAIN, then exactly that variable will be saved in the retain area (like a global retain variable).

- If a local variable in a **function block** is declared as VAR RETAIN, then the complete instance of the function block will be saved in the retain area (all data of the POU), whereby only the declared retain variable will be handled as a retain.

- If a local variable in a **function** is declared as VAR RETAIN, then this will be without any effect. The variable will **not** be saved in the retain area ! If a local variable is declared as PERSISTENT in a function, then this will be without any effect also !