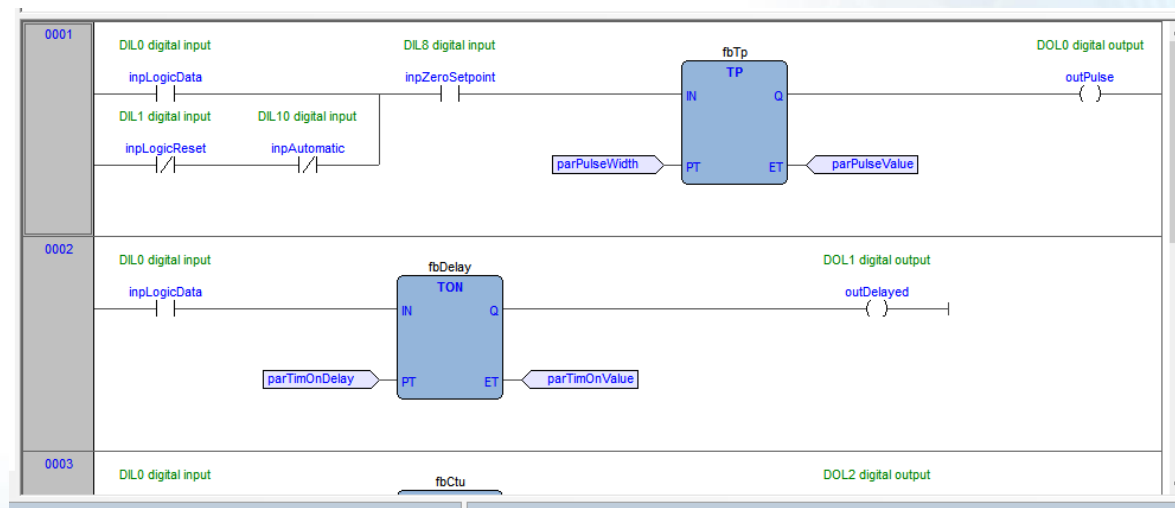


# LogicLab 4

**new features & enhancements**

- New Ladder Diagram (LD) editor
- Usage of expressions in graphic languages
- Customizable workspace tree
- Symbols cross reference
- New appearance for graphic languages (LD, FBD, SFC)
- Other improvements



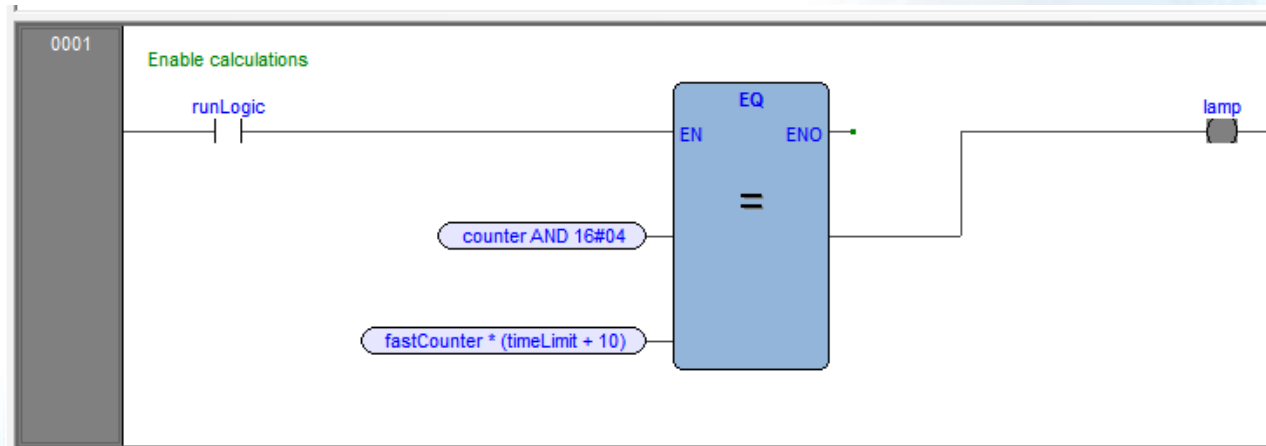
The LD editor has been completely re-designed in order to meet the typical LD programmers needing. Now LogicLab LD editor is in line with the most popular LD editors.

Major features are:

- Full auto routing and auto elements positioning
- Improved keyboard editing
- Blocks comments in addition to networks comments

LogicLab continues to support the old LD editor in order to allow a smooth migration from the old to the new LD way of coding.

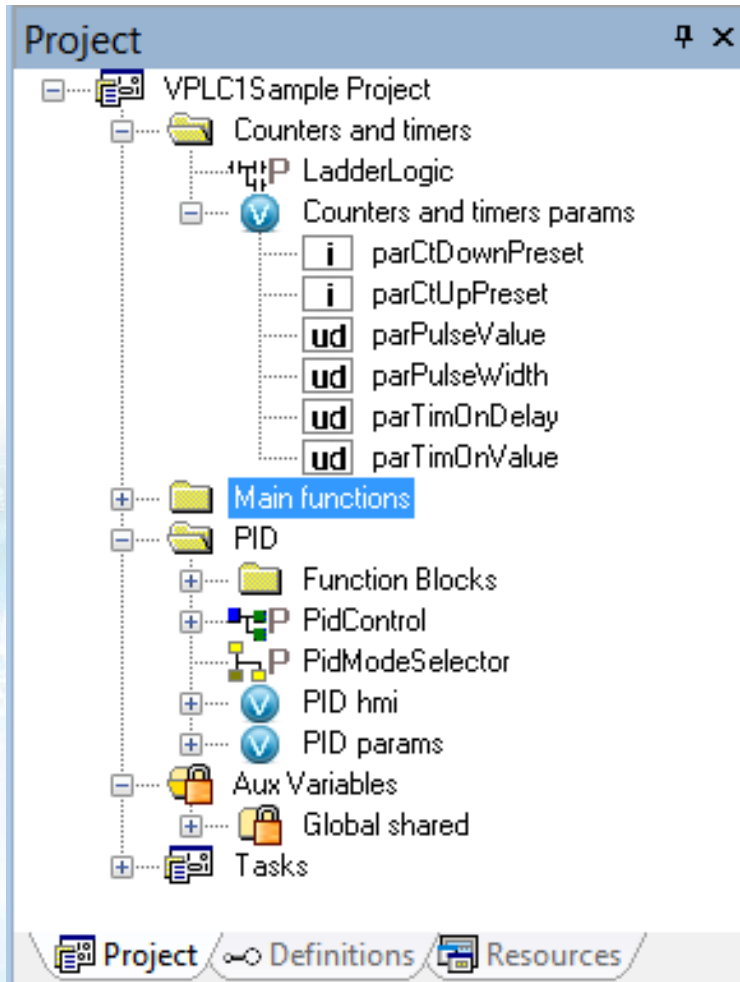
# Expressions in graphic languages



A new non-standard feature has been added in order to extremely simplify graphic code in LD, FBD and SFC languages.

LogicLab now allows to place ST expressions as input for operators and blocks.

Expressions extremely simplify LD, FBD and SFC coding, also the readability of the schemes is considerably improved.

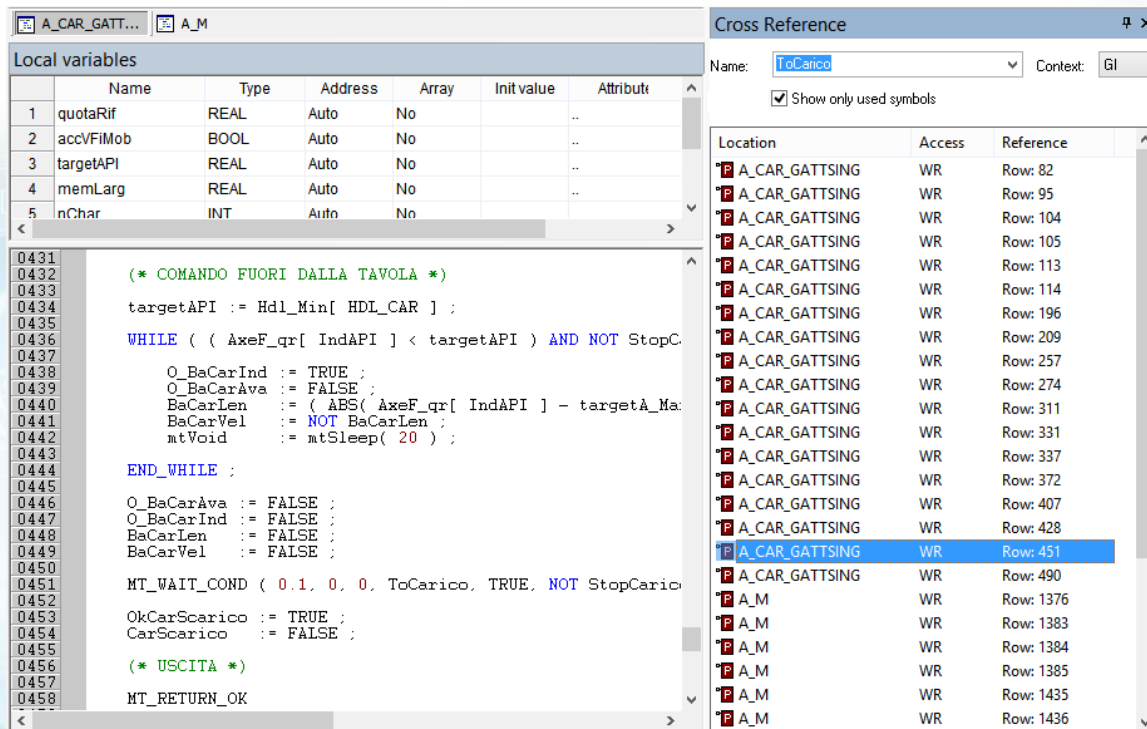


With the customizable workspace the user can organize its PLC application in folders in order to manage the elements of the project grouping them into a folder hierarchy that better reflect the logical organization of the application itself.

- Free folder definition and hierarchy similar to file systems organization
- Global variable grouping, each group with its own view in the editor
- Easy drag & drop placement of each element into the hierarchy tree
- Import and export of folders between PLC projects and libraries
- Auto-conversion of old fixed-format workspace trees into the new customizable format

The cross reference allows to keep track of each symbol (variable, function etc.) usage inside the PLC application.

Differently from a simple “Find in project” command, the cross reference will list only the places where the symbol is effectively used reporting also the kind of access executed (read, write, call, declaration).



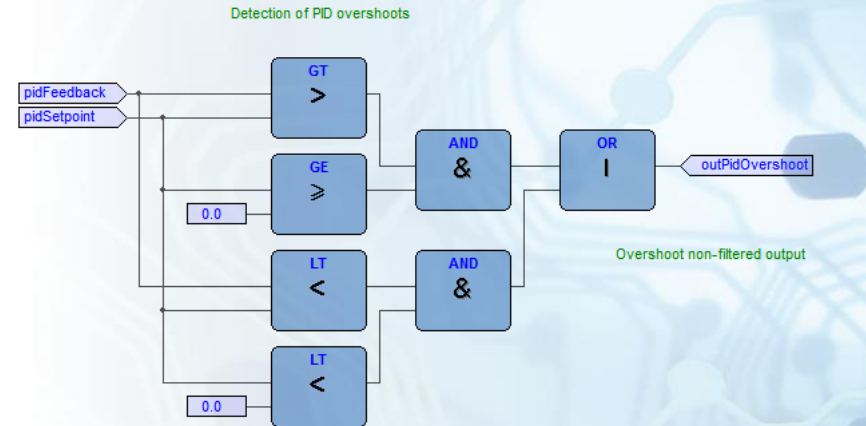
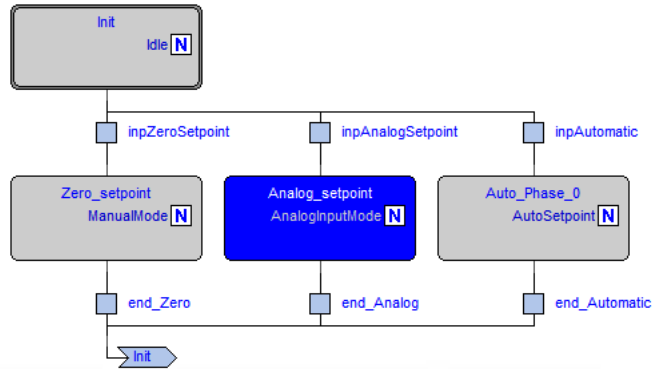
The screenshot shows a PLC development environment with a ladder logic program on the left and a 'Cross Reference' window on the right. The 'Cross Reference' window is titled 'Cross Reference' and has a search field containing 'ToCarico' and a context dropdown set to 'GI'. A checkbox 'Show only used symbols' is checked. The window displays a list of references for 'ToCarico' with columns for Location, Access, and Reference (Row).

Location	Access	Reference
A_CAR_GATTSING	WR	Row: 82
A_CAR_GATTSING	WR	Row: 95
A_CAR_GATTSING	WR	Row: 104
A_CAR_GATTSING	WR	Row: 105
A_CAR_GATTSING	WR	Row: 113
A_CAR_GATTSING	WR	Row: 114
A_CAR_GATTSING	WR	Row: 196
A_CAR_GATTSING	WR	Row: 209
A_CAR_GATTSING	WR	Row: 257
A_CAR_GATTSING	WR	Row: 274
A_CAR_GATTSING	WR	Row: 311
A_CAR_GATTSING	WR	Row: 331
A_CAR_GATTSING	WR	Row: 337
A_CAR_GATTSING	WR	Row: 372
A_CAR_GATTSING	WR	Row: 407
A_CAR_GATTSING	WR	Row: 428
A_CAR_GATTSING	WR	Row: 451
A_CAR_GATTSING	WR	Row: 490
A_M	WR	Row: 1376
A_M	WR	Row: 1383
A_M	WR	Row: 1384
A_M	WR	Row: 1385
A_M	WR	Row: 1435
A_M	WR	Row: 1436

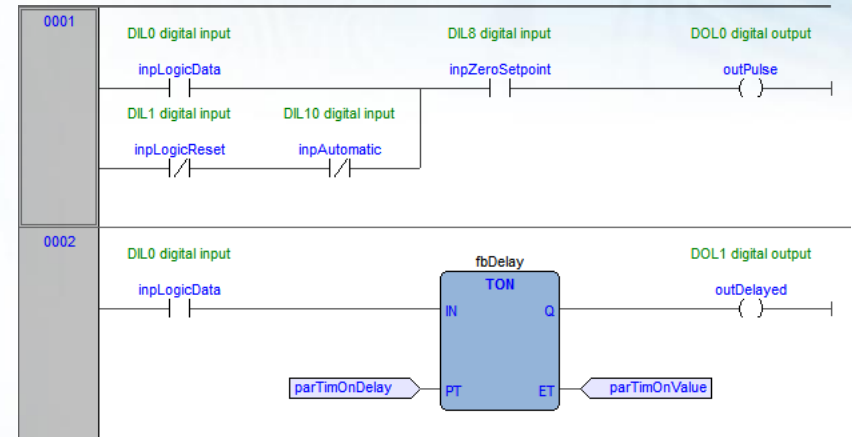
Cross reference key-points are:

- List of all effective usages of a symbol
- Dedicated view for browsing symbol references
- Access type (R/W, call, declaration) reported for each symbol
- Context sensitive cross reference symbol selection (the selection of an object in the workspace automatically selects the same object in the cross reference view)

# New style for graphic languages



Graphic editors have a new graphic appearance. The style is more simple, the elements are flat, the 3D borders have been dismissed. The result is an improved readability and a style that is aligned with the new software ergonomic.



Several new minor features have been added to LogicLab 4:

- Bit access di bit-word variables (BYTE, WORD, DWORD), like the following:  

```
controlWord.4 := statusWord.11 AND statusWord.3;
```
- SFC control flags allows to control the execution state of an SFC POU from other POU's thus allowing to stop or reset the SFC state machine independently from its internal state
- The SFC live debug has been improved in order to better display the active steps, now also the active step of an non-executing SCF POU is displayed
- The live debug of graphic code (LD, FBD, SFC) has been optimized in order to minimize the data request to the displayed part of the code
- String conversion operators have been added, TO\_STRING and TO\_STRING\_FORMAT are now available to convert numeric data in strings, the reverse conversion has been added to the numeric cast operators like TO\_INT, TO\_REAL etc.
- Now a backup of the PLC application is generated at save time. The backup copies of the PLC project are kept into the backup history folder. The depth of the backup history can be set into LogicLab options (default is 10)