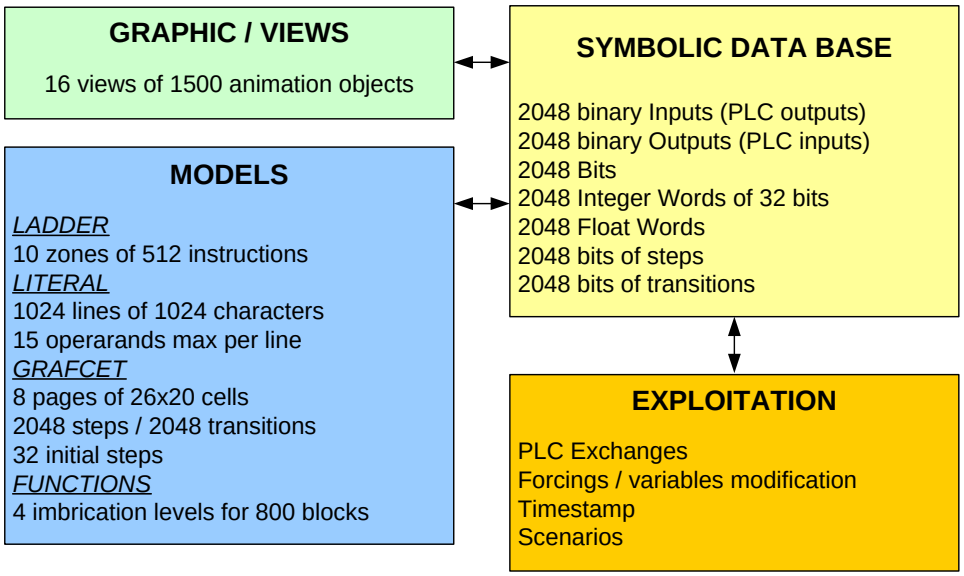
CONTENTS OF PROSIMUL 32 BITS

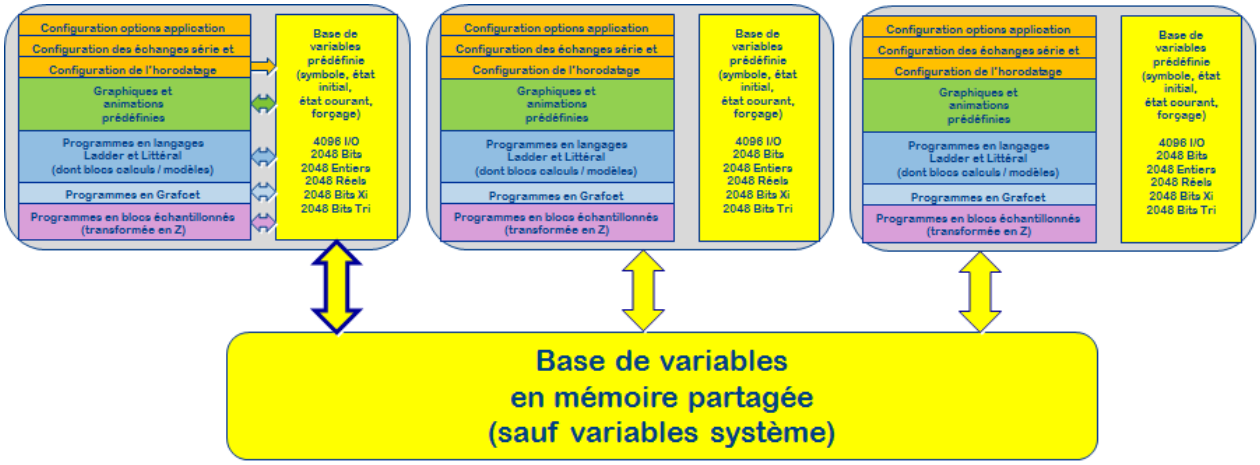
1. GENERAL INFORMATION
2. INSTALLATION
3. PROCEDURES
4. VARIABLES
5. GRAPHIC EDITOR
6. ANIMATIONS
7. LADDER EDITOR
8. STRUCTURED TEXT EDITOR
9. CHART EDITOR
10. FUNCTION EDITOR
11. EXCHANGES CONFIGURATION
12. EXPLOITATION
13. MONITORING
14. SCENARIO
15. IMPORT / EXPORT
16. CONVERTER
17. REPORT
18. EVOLUTIONS

GENERAL INFORMATION

Project Structure (.SIM)



Multi applications ou applications Maître-Esclave



Import / Export

An unique format (.BIM) for graphics and models.
A format text (.TXT) for exchanges of inputs/outputs symbols.

INSTALLATION

It's strongly recommended not to install PROSIMUL under virtual system because of less performance and process time-slices sensibility.

1 - Launch SetupPro32.
2 – Select USB or parallel key driver. Under Windows 7/8/10/11 (32-64 bits), administrator rights are required. This operation needs a reboot of the computer.
3 - Start Prosimul with or without file (*.SIM) in first parameter.
The Prosimul files (.SIM) association can be deleted via Windows configuration.
(Configuration ->Files->Association definition).
Other parameters could be given:
- RUNFORM : start application in training mode
- RUNWIN : start application in smallest window it could be.
- M1…M5 : start application on monitor 1 to 5 if existing.
ex : "PRO32fr.EXE TEST.SIM RUNFORM M2" start TEST application in training mode on monitor 2.

Protection Keys

- **USB** key : **"pro"** key for development (all options of the navigator), **"basic"** key for exploitation (application launching, modifications are possible in Configuration and Tools Menu).

Driver Installation : automatically installed with installer (automatic detection of OS). For **Windows 7/8/10**, the administrator rights are necessary. A reboot is necessary.

Software Launching / Cleaning PC

For runtime mode, discharge the PC from any program which can slow down simulation unnecessarily.
- If there is no Internet connexion uninstall antivirus (take care of USB keys).
- Uninstall drivers and services using Ethernet port (ROCKWELL/SIEMENS/SCHNEIDER/…).
- Deactivate services « Windows Search », « Windows Defender », « Windows Update »
- Deactivate services « Client DNS » et « Client DHCP » if PC have fix IP and no Internet access.
- Deactivate services « Remote Desktop Connection » et «Remote Access Connection Manager ».
- If SQL databases are installed but not used, deactivate its services.
- Deactivate services « Offline files » et « Computers explorer »
- Deactivate services « Google Update » (or other), « Adobe Acrobat Update Service » and all Updater unused.
- Verify task scheduler (Computer management) and deactivate all unused task.

Process priorities could be adjusted thru "Application" "Configuration".

PROCEDURE

Initial Menu (no project is opened)

New : create an empty project (seize a name)
Open : open an existing .SIM project (choose a name). When opening a project APPLI.SIM, if APPLI.BAT (batch file) exist on same directory, it would be executed.
Convert : converter of DOS application (select a DOS [.ACT] file)
Help -> Prosimul : access to the PDF help file
Help -> About : version information, available memory and maximum size of the project on the disc.

Project Explorer

Configuration : access to information of application configuration and exchanges with the PLC (default : no exchange).

Tools : access to tools of report management, monitoring information and scenarios files.

Graphics : access to graphic pages (0 to 15). **Names of page** can be provided (simple click on the name of the selected page). Only used pages are displayed in Project Explorer. **A right click on the root allows to "Add" blank pages**.

Variables : access to symbols, initial states, current values and forcing states of variables base. In standard version, 2048 binary inputs, 2048 binary outputs, 2048 integer words (32 bits), 2048 real (32 bits), 2048 bits of steps and 2048 bits of transition.

Ladder / **Literal** : access to the 16 zones of programs in LADDER or LITERAL , Main, sub-routines SR1 to SR15 and associated zone with Grafcet (execution after Grafcet cycle). **Names of zone** can be provided (simple click on the name of the selected zone). The selection of Ladder or Literal is executed when the first selection is done or by **right clicking** if program zone is empty. A right click on the root allows selecting "Add" menu. **Program named "Grafcet" can be executed after Chart cycle**.

Grafcet : access to the 8 Grafcet pages. Maximum : 2048 steps including 32 active steps (simultaneous). **Names of page** can be provided (simple click on the name of the selected page). A right click on the root allows selecting "Add" menu.

Function : access to the entry page of the function blocks. Sub-pages blocks allow structuring the application in 4 levels. All in all 800 blocks are usable.

Messages : access to the recording messages file named PROSIMUL.LOG.

Preferences : access to the commons preferences for all project (stored in PROSIMUL.INI). Those preferences are loaded at the start of PROSIMUL or when modified thru navigator. Parameters are : preferences for Literal editor, SMTP address and port, local IP address for each PROSIMUL application and "User" palette colours.

Project Menu (project is opened)

Export : save symbols (.TXT) or images+models (.BIM) to use in another project (seize a name). This function requires beforehand selections (see Import/Export).
I**mport** : read symbols (TXT) or images+models (.BIM) beforehand saved (see Import/Export).

Dictionary : this function work with Unicode text tabuled files (.TXT). The sub-menus are the following :
- Export : for saving in Unicode tabuled text file, the actual dictionary plus the new objects labels. The file must be choose and two options had to be slected :
    - to export (or not) the unused labels present in dictionary.
    - to export only labels issued from text objects.

If many languages exist in the dictionary, the Creation language must be fixed thru Application Configuration box. The file is opened with the Unicode Editor configured in Application Configuration.
- Import : to import a new Unicode tabuled text dictionary. The old dictionary is cleared before import. The Creation an Runtime languages can be configured (Application Configuration) after import operation. Imported words are displayed in PROSIMUL.LOG file. Not found words are labelled "-----Not Found-----".
- [XX] (language choice) : allow to switch between languages present in the dictionary. The source language is modified according to the choice executed by this menu. If a language is selected, modified labels of texts are modifying the dictionary.
- Initialize : for clearing all the dictionary of application (normaly only if case of problem).

Save : Save immediately the entire project on hard disk (.SIM file). A copy of the old file is done (.BAK file).
Save as : Save immediately the entire project on hard disk (.SIM) with choice of the saving name/directory (.SIM).

Run : switch to exploitation mode of the application according to selected PLC exchanges modes and configuration application. Project is automatically saved.

Raz_Errors : deletion of PROSIMUL.LOG file

Window : allow to switch easily between opened PROSIMUL applications (like Master/Slaves applications). **Screen select** move main window to next screen if available.

Messages Zones

Three messages zones exist in the status bar (bottom of screen).

- left : name of the active window,
- center : last information message (writed in the " **PROSIMUL.LOG**" file),
- right : position of the cursor for graphic editor.

VARIABLES (DIALOG BOX)

In the standard version, 2048 variables of each type are available (binary Inputs, binary Outputs, Bits, Integer Words [32 bits], Float [32 bits], Bits of steps and transitions).
This dialog box allows the seizure of the symbols/values and forced variables. It will also be used for the selection of variables for animations and programs.

Numbering (V12.02)

By default, variables are numbered from 0 to 2047. By adjusting « **DataBase.IO_HexaMode » parameter to 1** in preferences, numbering could be show in the "X.Y" format (X=0..127 / Y=0..15).

System Variables

Several systems variables are created automatically.
It is **imperative to not modify** the emplacement of these variables.

BIT 0 : SYS_TOP_DIXIEME → periodic (100ms – if process period is less) square signal**.**
BIT 1 : SYS_TOP_CENTIEME → periodic (10ms – if process period is less) square signal**.**
BIT 2 : SYS_INIT_GRAFCET → **Set** to 1 causes Grafcet initialization.
BIT 3 : SYS_INIT_APPLI → **Set** to 1 causes application reinitialization.
BIT 4 : SYS_RAZ_FORCAGE → **Set** to 1 causes cancellation of all variable forced state.

WORD 0 : SYS_HORLOGE_DIXIEME → word increment every 100ms.
WORD 1 : SYS_CYCLE_PROSIMUL → word contains the effective cycle time in ms (-1 when passing in STOP mode).
WORD 2 : SYS_PAGE_PROSIMUL → in Run mode, it allows to display graphic pages (value 0 to 15), Grafcet pages (value 20 to 27), Function Blocks (value 30) and Monitoring (value 31).
WORD 3 : SYS_COEFF_TEMPS → word allows the modulation of the real time running out - value expressed in % (initially at 100% → coefficient with 0% suspend the application).
WORD 4 : SYS_COMPTEUR_ECHANGE → word contains number of all exchanges with the PLC in serial connection (-1 in case of comm. error).
WORD 5 to 20 : SYS_COLOR_BLACK…WHITE → words containing value of color 0 to 15.

Symbols

- Maximum of 28 characters
- no space at the beginning and the end of string (a question mark replaces them if necessary).
- unavailable characters : « **(** » « **)** » « **{** » « **}** » « **,** » « **.** » « **;** » « **:** » « < » « > » « = » « + » « - » « ***** » « / » « **%** » « **#** », keywords of Literal program (If, While, Set,…) and calc functions (Pow, Abs,…)
- a control of double existence of symbol is made in every new seizure.
- a string may be remembered ([CTRL][INS]) in the course of seizure for a later use ([SHIFT][INS]).
- **a right click** on a seizure zone showing a floating menu which proposed the 10 last seizures.

Specific strings to use functions keys with symbol

- Associating to binary variables, strings **{FnnK}** include in the symbol allow the modification of the variable state in push/push mode (application in RUN). The variables can be modified by program.
- **nn** varies from **01** to **12** (12 function keys).
- **K** takes values **N** (single key), **S** (with Shift inserted), **C** (with Control) and **A** (with Alt).
- Example : " BPM **{F03S}** ". The state of the variable must be changed when [SHIFT] and [F3] key pressed during application exploitation.

Initial State

An initial state can be seized **in the selected base**. This state will be transferred at the launching of the application if the option initialization is retained in the screen of application configuration. This state can be modified in RUN mode with "WRITEINI" instruction.

Current Value

This zone allows to modify the variables in dynamic mode (forcing or not). It also allows to test animations during development phase. This value must be seized in the selected base.

Bases

For the integer words, display base and values seizure can be selected at the left-bottom of the page among 3 following one :

binary :	2#
decimal :	default
hexadecimal :	16#

Select variable(s) beforehand (see below).

Forcing

A check box is reserved to force directly the variables. In RUN mode, the program cannot write current value of forced variables : checking or unchecking box cause to force or unforce variable.

Selection of Variables(s)

- numbered buttons on the left of the symbols allow the selection of a variable (simple click) or several variables (click with SHIFT pressed on the last variable). Selected variables pass in blue visualization. Selected variables are obligatorily consecutive.

- Different tools operate on the selected variables :

. basic changes (for real words),
. copy/paste/cut/delete/insert/import/export functions.

A double click *on a numbered button causes* ***the exit of the dialog box*** *and memorizes the selection (like exit with single selection). Use it for the selections from the animation box and programs.*

Copy tool/ [Ctrl][C] / [Ctrl][Insert]

In STOP mode, selected variables are stored in buffer memory (symbols, initial and current values).
In STOP mode you can also use “Export” function (main menu) that export data in formatted TXT file (SYMBOL tab VALUE).

In RUN mode, copy store only text in memory (for search tool) and to clipboard (to be pasted).

Cut tool/ [Ctrl][X] / [Shift][Suppr]

Selected variables are stored in buffer memory and removed from the base. This function can be used **to move** variables in the base (see pasting).

Paste tool/ [Ctrl][V] / [Shift][Insert]

Variables in buffer memory are stored on the selected variables in destination (normally free).
A control of **double existence** of symbol is carried out.
Bases are adapted if necessary.
Modify the initial and current values if necessary.
A replacement of the cut variables is carried out in animations and Ladder, Grafcet and Function models. This can allow changing placement variables in the base.
Important : All operations (like Suppress or Insert) done between “Cut” and “Paste” **must** operate on the entire cute zone. Pay attention at the time of cut/paste between different types of variables because animations and programs (Ladder, Grafcet and Function) using them await sometimes types of fixed variables (default).

Increment tool

The selected variable (single) is duplicated on the following variable with increment of an index located at the end of the string (example : BOX_01 is duplicated in BOX_02).

Insert tool

This function allows to insert N free lines in the base. The selected variables (blue visualization) are shifted downwards.

Delete tool

This function allows to release N free lines in the base. The selected variables (blue visualization) are shifted upwards. It is interesting to release free lines **to limit the cycle time** in exploitation.

Search / Next tool

This function allows the search for a string or a forcing in the variables base.

. For research text, seize the desired string.
. For research **forcing,** seize the string **F#** .

The first found variable is selected. The " Next " function allows the continuation of the same research.

Print tool

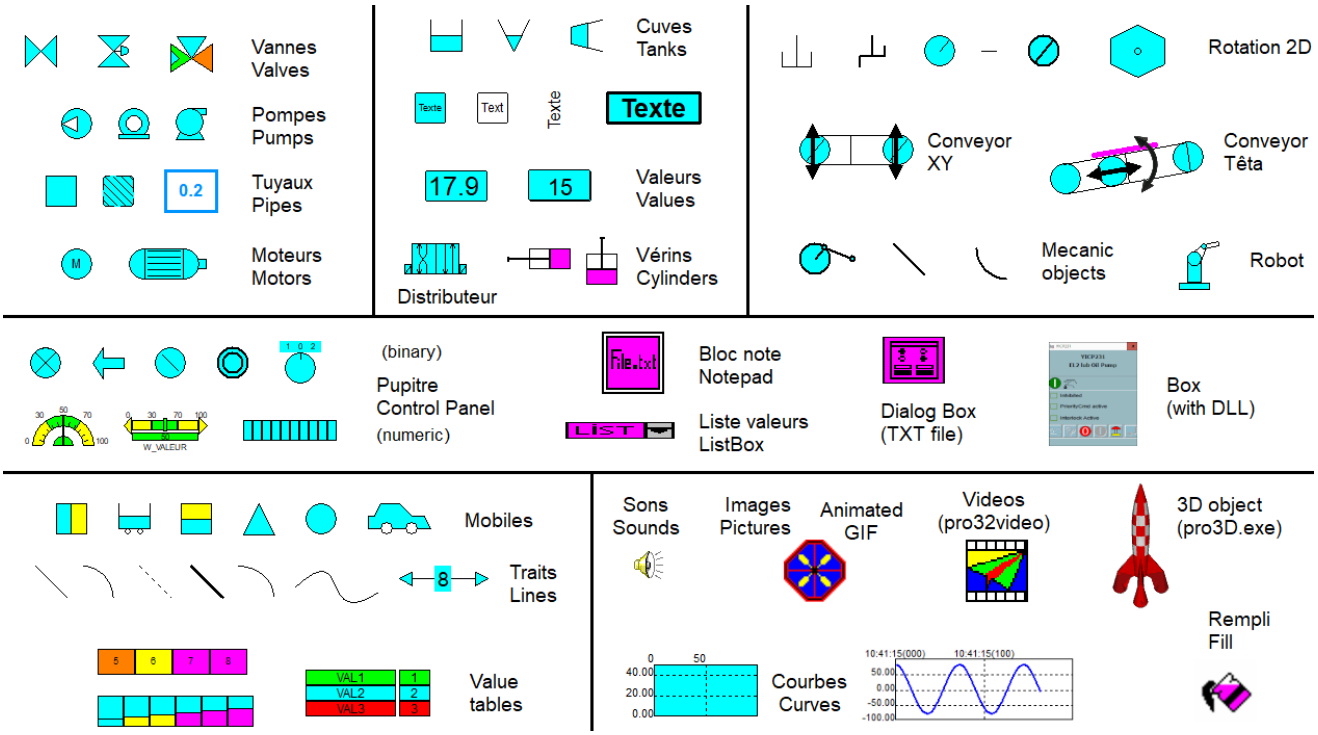
The selected variables are printed on the selected printer.

Shorcuts for using variable in editors

- **[SHIFT]** key associated with [/], [P], [N], [F] enable to use “Not, “Rising Edge”, “Falling Edge” and “Test force state” to the used variable.
- **[SHIFT]** key associated with [S], [R], [U] enable to use “Set, “Reset” and “Unforce” function to the used variable.
- All those shorcuts are available thru mouse right-click.

GRAPHIC EDITOR

This editor allows to exploit an objects library which have predefined animations (color, positions...), activables or not. It is necessary to know them before using an object.



Graph Menu

Graph → Create : access available to the object pallet. ***Double-click*** on ***an empty zone*** of the Graphic page causes the same action. Type the Unicode label of the object.
Graph → Modify Type : access to the object pallet to modify type of selected object (allows also a new selection of file name for the Sound and Image objects).
Graph → Properties : access to the configuration screen of the objects properties (position, size, animations...). Double-click on selected objects do the same action.
Graph → Resize : allow to resize all the objects with a given ratio. If magnetism is active, objects are positioned on grid, near objects should conserve their contact points.
Graph → Screenshot : make a copy of picture in clipboard.
Graph → Clean : to clean graphic object if application have display troubleshooting. Animation variable type is also controlled.
Select the page thru the Navigator → move to the command "Graph / Clean" <u>without entering in the graphics zone</u>. Execute the command "Clean".

Zoom[+]/[-] :in STOP mode, allow to show picture with display ratio (50% to 200%) by step of ±10%. **Mouse wheel** associated with [Ctrl] key do the same function.

Edit -> Undo : cancel the last operation (5 levels).

*NOTE : many functions are accessible via **the right click** with or without selected objects.*

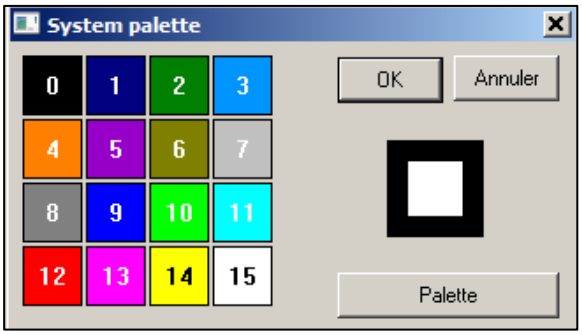
<u>Without selection</u>
Edit → Paste : paste the memorized objects to the origin position (shifted by 5 pixels on X and Y except if [SHIFT] key is pressed) .
Edit -> Paste object only : paste the memorized objects without animation.

<u>With selection</u>
Edit → Copy : store the selected objects in buffer memory. The copied objects could be exported .
Edit → Cut : copy and delete the selected objects.
Edit → Duplicate : make a copy / paste of selected objects.
Edit → Duplicate+Increment : duplicate selected objects with increment of animation variables having index number at end of string (example : FDCS_VERIN_01 becomes FDCS_VERIN_02 if this symbol exists in the base just after the first one).
Edit → Duplicate+Next : duplicate selected objects with next animation variables in the database.

Edit → Select all : select all objects of the current page.
Edit → Select Filled / Texts / Values Objects : select all filled, text or value objects of the current page.

Order → Group : create an object which groups the selection. Operations of resizing, displacement with magnetism can be done on the group. Each object in group keep its own characteristics of animation. A group can be gathered with other objects. The unit is transformed into only one and single group (only one level of grouping).
Two animations are available : "Hide" and "Move x, y" position of grouped objects.
Order → Dissociate : dissociate the grouped objects.

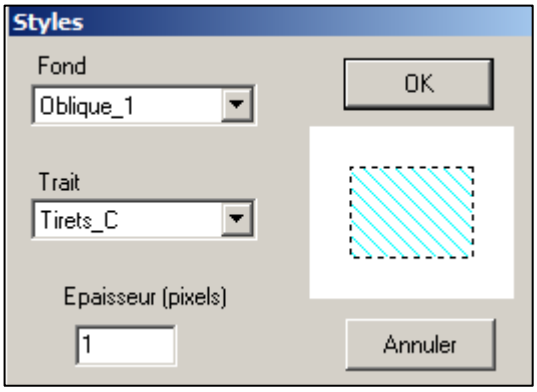
Order → Foreground : pass the selected objects on the 1st plan.
Order → Background : pass the selected objects on the background plan.



Color -> Objects : access to the colors box (number 0 to 15 for System palette and 16 to 31 for User palette).
- Selection of **foreground** : **left click** on color
- Selection of **background** : **right click** on color
- Double left click : validate (OK).

Color -> Background : selection of the background page.

- "**Palette**" button allow switching between "System" and "User" palette. "User" palette can be adjust thru "Preferences" and is stored in PROSIMUL.INI file (version < 11.06) or in .SIM file (version >= 11.06).



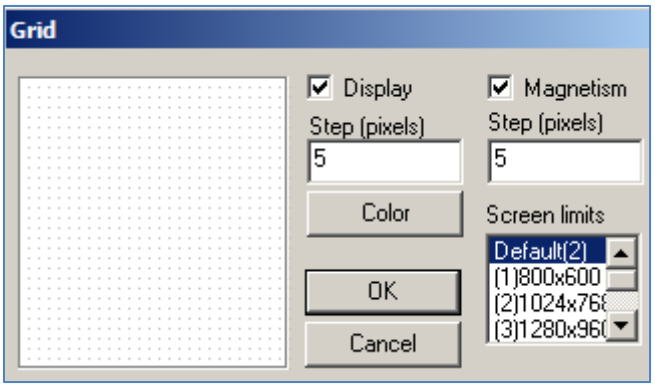
Style : access to the thickness box of the line, styles of border and background. A thickness above 1 produce a **Bold** Font selection for TEXT or VALUE objects.

Font : access to the font box of character and the framing of the texts.

NOTE : If no object is selected, the choices of color, style and font became the default options. In the other case, only the selected objects are affected by these choices.

Grid : access to the box of the grid and magnetism. For grid and magnetism steps, colour, showing and magnetism selections.

*MAGNETISM : if it is active the operations of movement/sizing (**keyboard or mouse**) move **point 1** on grid depending of it step. In case of movement of **grouped objects**, objects keep the same distance between them but move with grid step. **[M] key** toggle magnetism function.*



A screen limit can be selected. Combined with parameter "Animations" / "Page" selected (Configuration/Application), it allows an auto-sizing of synoptic in Run mode depending from the used screen dimensions.

Search : search for a string in the label or symbols of the different objects. Type the string. The first object containing the string is selected (research in all graphics pages).

Next : repeat the same search.

Objects library / Objects Creation

The menu **Graphic->Create** or the double-click on an empty zone causes the appearance of the objects library.

- Select the type of object to be created in the navigator.
- Seize so necessary label of the object.
- Particular cases of label :
  . Text : Maximum of *29 characters* in Unicode format (reminder : font and framing are adjusted independently - Graphic - > Font). **The character " \ "** allows to create texts on several lines.
  . Value : Indicate the desired format displaying **XXX.YY** (format valid for the integer and real values). The values can be displayed into binary (syntax **2#**XXXX...) or hexadecimal (format **16#**XXXX...). Except animation, a default value is displayed (50).
  . Inter_3P : The text is display above the object (ex : MANU…0…AUTO).
  . Bitmap/Image/GIF/Video/Notepad : Select the file name on ***repertory of application*** .SIM obligatorily. The bitmap is displayed in its origin size.
  . Sound : Select the file name (.WAV) (on ***repertory of application*** .WAV obligatorily). An icon is displayed. Only the animation of the object will make the audible sound.

- The object is created in top-left of the graphic page with default properties (colors, styles and font). For the **Filled** object, the color of outline indicates the border color of filling.
- The object can be moved and resized (see below).
- Adjust its animations properties.

*OBJECTS : the positioned objects on the pages have two points of fixing (**1** & **2**) defining their position and size. See below Select/Move/Size operations. Access to the properties of the object (double-click on objects or menu Graphic->Properties) gives the access to the manual adjustments of the co-ordinates of the 2 fixing points.*

Objects Selection

A selected object is represented with two square " handles".

- When the mouse is moved above an object, " arrow " cursor is changed into " cross " cursor. The object is selected by a simple click.

- To select several objects, repeat the same operation by maintaining **[SHIFT]** key (multiple selections).

- To select objects on background order, repeat selection with **[ALT]** key.

- An object or a group of objects can be selected by **a rectangular zone**. This can be useful in particular for overlapping objects :
  . Simple click on a place free of object or with simultaneous **[CTRL]** key.
  . Maintain the mouse click and increase the zone to full include objects (for **the valve** the zone must cover the exit stem).
  . Relax the mouse click. The selected objects appear with " handles ".
- Menu **Edit->Select all** allows to select the group of the objects of the same page.

Objects Movement

- selected objects can be moved with direction keys of keyboard.
- the maintenance of the left button of the mouse on selected objects causes the appearance of a " 4 directions " cursor. Mouse movements affect its position.
- In these 2 cases, if magnetism is active, the step of displacement will be the step of the grid.

Objects Size

- the passage of the cursor on one of the 2 " handles " modifies its aspect (" cross_1 " or " cross_2 "). A maintained click on these points involves a displacement of the " handles " according to movements' of the mouse.

- With the **keyboard** :
  . **[SHIFT]** key associated with direction keys modifies Point_1.
  . **[CTRL]** key associated with direction keys modifies Point_2.

- With the **mouse** : :
  . **[SHIFT]** key associated with mouse movements do resize the object that is proportional to ratio Y/X of screen.
  . when resizing a **group object**, all the objects of the group are resized by homothetic relations.

Window moving

The graphic window can be move :

- With the **mouse wheel** :
  . by default left / right movement.
  . **[SHIFT]** key allow up / down movement.

- With the **keyboard**, if no object is selected :
  . **[SHIFT]** key accelerate direction keys.
  . **[HOME]** key set the origin to (0,0).
  . **[END]** key move the window to the last object.

Mouse Right Click

With selected objects a right-click on the mouse open a context menu with Edit functions and allow to adjust:
. **[Pos]**: X or Y value (Z for X and Y) of Point 1 of selected objects.
. **[Size]**: X or Y size of selected objects.
. **[Format Value]**: format of numeric or real value of selected objects.
. **[Rotate Group]** : rotation of objects group according to center of the group
. **[Dev.Only/Training]** : modification (not) of « Dev.Only » parameter.

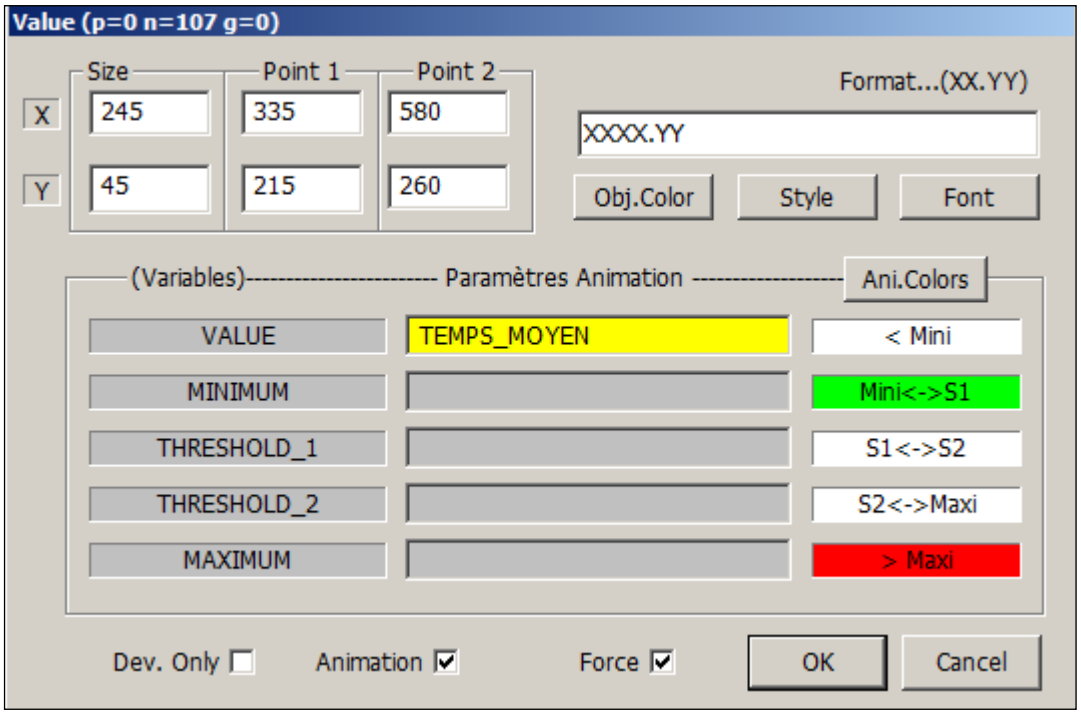
Keyboard shortcuts

. **[HOME]** : move display screen at origin.
. **[END]** : move display screen on farthest object.
. **[ESC]** : cancel all objects selection.
. **[CTRL][F]** : first string search in label and symbol associated with graphic objects.
. **[F3]** : next text search.
. **[CTRL][INS]/[CTRL][C]** : copy to clipboard function.
. **[SHIFT][INS]/[CTRL][V]** : paste from clipboard function.
. [**CTRL][W]** : paste objects without animation.
. **[DELETE]/[CTRL][X]** : copy and suppress selected objects.
. **[CTRL][D]** : duplicate selected objects.
. **[CTRL][I]** : duplicate selected objects with variable increment if possible.
. **[CTRL][Z]** : « Undo » function (5 levels).
. **[<-][->][^][v]** : without selection, move display screen. With selection(s), move/size selected objects depending from magnetism function and key [SHIFT] (point 1) or [CTRL] (point 2).
. **[M]** : toggle magnetism function.
. **[C]** : call « Colour objects » dialogbox.
. **[S]** : call « Styles » dialogbox.
. **[P] or [F]** : call « Fonts » dialogbox.
. **[SPACE] or [A]** : modify selected objects text alignment.
. **[O]** : put selected objects on foreground.
. **[B]** : put selected objects on background.
. **[X]/[Y]/[Z]** : modification of X or Y or with Z both sizes of selected objects.
. **[V]** : modification of display format of selected values.
. **[R]** : objects group rotation according to central point.
. **[0, 1..5]** : enable to modify animation colours 1 to 5 (0 for all).

ANIMATIONS

General

Each object has own characteristics of animation. Those are reached by the double-click on object, or via the right click->Properties or the menu **Graphic->Properties** (after selection).



The dialog box " Objects Properties " gives an access to :

. label of the object.
. its 2 fixing positions or/and its sizes X, Y.
. its attributes of color [C], styles [S] and fonts [F].
. its colors and associated animation variables according to rpreset rules (see table below).
. selection of display mode (dev. Only – see Application Configuration).
. validation of animation check box.
. validation of the force check box of the 1st animation variable. Forced state will be obtained by clicking on the object in RUN mode.

When an object is declared animated (check box) and settings correctly done, it will be drawn on the screen according to its animation rules. In this case it becomes " dynamic " object in opposite to " static " objects.

Colors selection

A simple-click on the assigned rectangles to the color, causes the appearance of the colors dialog box.
A double-click on one of them select the color.
A simple click on “Ani.Colors” button permit to change all the animation colors.
In the graphic editor, keys [1] to [5] allow to select 1 to 5 colours for selected objects. The [0] key allow to modify all colors.

Seizure of the animation variables

- Seizure zones of variables can :
  . be empty (not selected animation - **grey bottom**),
  . contain immediate values (0,1, or numeric/real values - **white bottom**),
  . receive symbols resulting from the base (**yellow bottom**). The symbol can **be seized directly or** reached in base by a simple click on its information text (on the left). **A right click** on the edition zone allows a choice among the 10 last seizures (floating menu).

- Types of indicated variables below must be respected.

- For **binary variables,** additional operations (rising or downward front) can be used :

. syntax complement :	/#VARIABLE
. syntax rising front :	**P#**VARIABLE
. syntax downward front :	**N#**VARIABLE
. syntax for testing state 1 of force bit:	**F#**VARIABLE

Shorcuts for using variable in editors

- **[SHIFT]** key associated with [/], [P], [N], [F] enable to use “Not, “Rising Edge”, “Falling Edge” and “Test force state” to the used variable.
- **[SHIFT]** key associated with [S], [R], [U] enable to use “Set, “Reset” and “Unforce” function to the used variable.
- All those shorcuts are available thru mouse right-click.

*NOTE : **rising and falling edges** are true during only 1 cycle of program.*

- In exploitation these variables will be animated.

Animations Tables

[CF] = change background color,
[CC] = change outline color,
[PR3] = proportion according to rule of 3,
[POSXY] = movement in X and Y (X+ for right movement, Y+ for up movement),
[LGXY] = variation length in X and Y,
[R2D] = effect of 2d-rotation,
[VAL] = displaying of values (alpha or curve),
[BMP][WAV] = specific to the images and sounds.
[SYM] = displaying Symbol of Data
[DIS] = hide or display objects
[TXT] = text seizure
[DIA] = Dialog Box

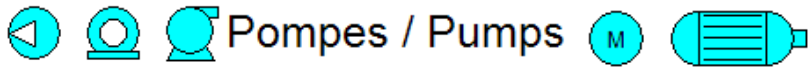
Tank – Pump-Motor - Valve – Cylinder - Conveyor - Mobile - Rotation 2D - Robot - Distributor
Pipe-Panel Binary – Value-Panel Numeric - Mechanic– Measurement
Text – Notepad – ListBox – Dialog Box – Line - Bezier– Curve
Fill – Sound - Picture – Animated IF – Video - Objet 3D - Table - Group

Vannes / Valves

2 ways : 5 binary variables + 4 colors – **3 ways** : 4 integer variables (like Cylinder)

[CF]	FDCF(NC)	FDCO(NC)	CTRF	CTRO	INDICATOR	State
	0	1	-	-	-	Closed
	1	0	-	-	-	Opened
	1	1	1	0	0 1	Operate (closing)
	1	1	0	1	0 1	Operate (opening)
	1	1	0	0	-	Operate
	Other cases...					Default

For **3 ways valve** animation colours are indicated in percent of opening position.

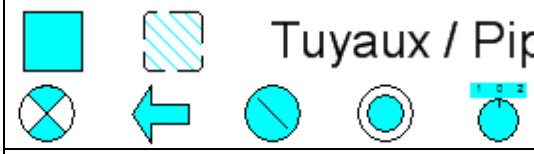


Pompes / Pumps Moteurs / Motors

2 binary variables + 2 integer variables + 3 colors

	START	INCIDENT (NO)	State
[CF]	0	0	Stop Color
[POSXY]	1	0	Start Color
	-	1	Default Color

HORIZONTAL : object is moved by X horizontal pixels (Up=+)
VERTICAL : object is moved by Y vertical pixels (Right=+)



Pupitre / Control Panel (Binary)

3 binary variables + 4 colors

	VISU 1 / BACKGROUND 1	VISU 2 / BACKGROUND 2	State
[CF]	0	0	State 0 Color
[PILOT]	1	0	State 1Color
[POSXY]	0	1	State 2 Color
	1	1	State 3 Color

SECONDARY color is used only by Lamp and PushButton objects.

HORIZONTAL : déplacement en X de l'objet
VERTICAL : déplacement en Y de l'objet

CMD_MODE :
0 → first variable of object is forced by push/push
1 → first variable of object is forced by continuous action
21 → first variable of object is is always forced to 1.
20 → first variable of object is is always forced to 0.
>= 100 → first variable of object is forced by pulse >= 100ms

Note :
- SWITCH_2P can be used as switch 2 positions (with oblique feature depending on the state of VISU 1).
- SWITCH_3P can be used as 3 positions switch. The line is draw on left direction if VISU1=1, in right position if VISU2=1 or center position in the others cases.

Surface (pipe)

Ratio Pixel (float) : X and/or Y dimensions are multiplied themselves and by this ratio to produce one surface unit displayed according to the title format (ex : 0.0).
Font Size (integer) **:** contains the font size of the displayed value for surface.

Robot

4 integer variables

[POSX]
[R2D]

The 2d rotation of the axes is displayed according to angular values' **AXE_1** and **AXE_2** expressed in degrees (+ in anti-clockwise direction).
HORIZONTAL : object is moved by X horizontal pixels (Up=+)
VERTICAL : object is moved by Y vertical pixels (Right=+)

Texte	Texte	Texte	Texte	Texte

	2 binary variables + 2 integer variables + 4 colors
[CF][CC] **[POSXY]**	T_FONT_BK : display text and background colours according to variable **VISU** (0/1) and according to the **BACKGROUND** (0/1) T_FONT: display text colour according to variables **VISU1 and VISU2** T_BK : display background colour according to variables **BACK1** and **BACK2** **CMD_MODE** : **0** → first variable of object is forced by push/push **1** → first variable of object is forced by continuous action **10** → **text is modified directly by clicking on object**. **21** → first variable of object is is always forced to 1. **20** → first variable of object is is always forced to 0. **30** → **hour** display (see title format below – ex : **HH:mm:ss**→12 :27 :45) **31** → **date** display (see title format below – ex : **dd-MMM-yy**→06-Oct.17) **>= 100** → first variable of object is forced by pulse >= 100ms **HORIZONTAL** : object is moved by X horizontal pixels (Up=+) **VERTICAL** : object is moved by Y vertical pixels (Right=+)

FORMAT TIME

h	Hours with no leading zero for single-digit hours; 12-hour clock
hh	Hours with leading zero for single-digit hours; 12-hour clock
H	Hours with no leading zero for single-digit hours; 24-hour clock
HH	Hours with leading zero for single-digit hours; 24-hour clock
m	Minutes with no leading zero for single-digit minutes
mm	Minutes with leading zero for single-digit minutes
s	Seconds with no leading zero for single-digit seconds
ss	Seconds with leading zero for single-digit seconds
t	One character time marker string, such as A or P
tt	Multicharacter time marker string, such as AM or PM

FORMAT DATE

d	Day of month as digits with no leading zero for single-digit days.
dd	Day of month as digits with leading zero for single-digit days.
ddd	Day of week as a three-letter abbreviation.
dddd	Day of week as its full name.
M	Month as digits with no leading zero for single-digit months.
MM	Month as digits with leading zero for single-digit months.
MMM	Month as a three-letter abbreviation.
MMMM	Month as its full name.
y	Year as last two digits, but with no leading zero for years less than 10.
yy	Year as last two digits, but with leading zero for years less than 10.
yyyy	Year represented by full four digits.
gg	Period/era string.

Notepad

	2 binary variables + 3 integer variables
[DIS][CF]	**VISIBLE** : display text or not (end of text is character ¤).
[POSXY]	**BACKGROUND COLOR** : color (0..31) depending from PROSIMUL palette
	HORIZONTAL/VERTICAL : X/Y pixels displacement on screen

ListBox

	LINES NUMBER (Int) :
[CF]	0 = no text displayed but background color, no selection available
[POSXY]	1 = 1 line displayed with background color, no selection available
[TXT]	N(>=2) = 1 line displayed, selection is available with N items visible if « Forc » option is active
	DISPLAYED LINE (Bool / Int) : n° of displayed line coming from .TXT file (<0 : no display) configured in title.
	BACKGROUND or BORDER/TEXT(Bool / Int) : selection of background or text color 0 to 4
	HORIZONTAL/VERTICAL : X/Y pixels displacement on screen

17.9 Valeurs / Values

Pupitre / Control Panel (Numeric)

	5 variables (all standards) + 5 colors
[CF][VAL]	The **VALUE** is displayed according to the selected format (ex : 0.00).
[SYM]	Special formats : binary : 2#xxxxxxx, hexa :16#xxxx, Time : T#format
	For T# format value of time variable must be expressed in milliseconds starting at 0 hour.
	The display of background is a function of **MIN, MAX** and **THRESHOLDS**.
	Value of « **VALUE_T2 »** object can be seized directly by clicking on object and MIN / MAX limits are controlled.
	CURSOR object can be mouse controlled by clicking on left or right arrows or by moving cursor or clicking on value.
	For “AnalogTxt” and “Cursor” objects, symbol can be hidden by putting “-“ character at the end of format (ex: 0.00-).

Dialog Box

[DIA]	5 variables (Bool / Int / Float) + 5 colors Dynamic dialog box created thru text file .TXT indicated in title. See texte file format below (Excel tool available on demand)

FORMAT of text file (.TXT) for CUSTOM Dialog Box
One line for box descriptor.
Code Box : 33150 = separator **'**,**'** 32150 = separator Tabulation
33150/32150 : controls coordinates are indicated in window
3315**1**/3215**1** : controls coordinates are indicated in frame of window, horizontal size automatic (full width) if <= 0.
PosX / PosY: position of box on the screen. -1=automatic position near the click point

Ligne 1	Caractéristiques générales de la boîte								
	Code Box	PosX (écran)	PosY (écran)	Largeur (écran)	Hauteur (écran)	Titre			
Ligne N	Définition des objets (3 lignes par objet)								
	Type contrôle	PosX (box)	PosY (box)	Largeur (box)	Hauteur (box)	Text0	Text1	Text2	Text3
Ligne N+1	Attributs d'affichage					Couleurs de fond			
	Font Size	**Fore** Color (0..31)	**Border** (0 / 1)	**Align** |0_1_2|	**Bold** / Gras (0 / 1)	**C0** (0..4)	**C1** (0..4)	**C2** (0..4)	**C3** (0..4)
Ligne N+2	**Visible**		**Enable**		**State**		**Valeurs**		
	VarV (-1,0..4)	Default (0 / 1)	**VarE** (-1,0..4)	Default (0 / 1)	**Var1** (-1,0..4)	**Var2** (-1,0..4)	**Val1** Min	**Val2** Max	

Control type : 50=Text, 51=Value without seizure, 60/62=Button Text with/without color, 61/63=Button Value with/without color, 64=button with icon, 70=Radio Button, 72=Check box, 81=Input Value, 90=ListBox, 101=Bargraph, 103=Scale with cursor.

Text0…3 : texts or .bmp file name (code 64) displayed depending from State value.
Var V/E/S : animation variable n° (0 to 4).
Visible : if VarV is true (1), control is visible. If -1 value of **Default** is taken.
Enable : if VarE is true (1), control is editable. If -1 value of **Default** is taken.
State : if used, state is calculated by (Var 1 + Var 2) for numeric variables or (Var 1 + 2* Var 2) for boolean variable. If -1 **Val1/Min** is taken
For Text controls (codes 50, 60, 90) **Var1/2** drive or/and receive n° of displayed text.
For Value controls (codes 51, 61, 81, 101, 103) state is displayed or draw or transmitted (81).

Values :
- binary command (codes 70, 72) : **Val1** and **Val2** value are store in **Var1** et **Var2** if used
- numeric value (code 81) : low and high limits of State
- bargraph (101) ans scale (103) : low and high limits of display

Cn : background colors (0..4) selected among configured colors (animation box).
For numeric controls (51, 61, 81, 101, 103) :
- State < Min → **C0**, Min <= State <= Max → **C1**, State > Max → **C2**, Enable=false → **C3**.

For text controls, selected color Ci follow current **state** of Var1 and Var2 (0..3 - modulo). If **Enable** is false color 4« Window » is display.

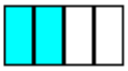
Cuves
Tanks

[CF][PR3]	<u>5 variables (all standards) + 5 colors</u> The **HEIGHT** is displayed in proportion of the **MAXIMUM** indicated. The display color is a function of the **THRESHOLDS** (selected or not). **If first character of comment is | tank is closed**.

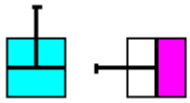
Cuves
Tanks (rotative)

[CF][PR3] **[R2D]**	<u>2 variables (all types) + 3 variables (integer) + 5 colors</u> **VOLUME** is displayed in proportion of **VOLUME MAX**. Displayed color depends from **%** of volume (0-20-40-60-80-100%). Tank form angle is given by **FORM ANGLE EXT** (Int). Tank rotation angle is given by **FIX ANGLE** (Int). Rotation point is calculated as below (integer value) : PR.X = MilieuObjetX + OFFSET_XY/10000 – 1000 PR.Y = MilieuObjetY + OFFSET_XY%10000 – 1000 Ex: OFFSET_XY = 10001000 ➔ rotation point is centred Ex: OFFSET_XY = 09001000 ➔ rotation point is left shifted by 100 pixels

Pupitre / Control Panel (Numeric)



[CF][PR3]	<u>4 integer or real variables</u> Quarters (numbered by **"N-PART"**) are displayed according to **VALUE** and proportionally to the **MINIMUM** / **MAXIMUM**. Bargraphe VALUE can be <u>controlled</u> by mouse clicking on one of quarter (to reset, click on last superior active quarter).



Vérins / Cylinders

[CF][PR3] **[POSXY]**	<u>4 integer variables + 2 colors</u> The display of stem is a function of the value **TEMPO** and proportionally to the **MAXIMUM**. Entry or exit color is displayed according to direction of operation. **HORIZONTAL / VERTICAL** : object is moved by X / Y pixels (Right+ ; Up=+)

Mobiles

[CF][PR3] **[POSXY]**	<u>5 integer variables + 3 colors</u> The **HEIGHT** is displayed in proportion of the indicated **MAXIMUM**. The display color is a function of the **THRESHOLD** (selected or not). For RX et RY mobiles, SECONDARY color is used to fill from current HEIGHT to MAXIMUM **HORIZONTAL / VERTICAL** : object is moved by X / Y pixels (Right+ ; Up=+)

Distributeur

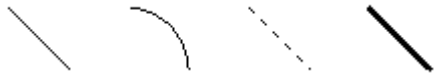
[POSX]

2 binary variables + 2 integer variables

According to **LEFT** (all standards) and **RIGHT** (flip-flops) commands, the distributor shifts on the right (all standards) or on the left.

HORIZONTAL : object is moved by X horizontal pixels (Up=+)

VERTICAL : object is moved by Y vertical pixels (Right=+)



Traits / Lines

[CF][CC]
[POSXY]
[LGXY]

5 integer variables

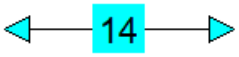
HORIZONTAL and **VERTICAL** have a number of pixels +=right, +=up) to change the position of the line according to *the type of retained animation*.

DISPLAY COLOR contains the displaying color of the line (0... 15). -1 for non-display.

CLEAN COLOR contains the cleaning color to the old position (0... 15 - to allow movements effects for example). -1 for non-display.

ANIMATION TYPE	Point 1		Point 2	
	HORIZONTAL	**VERTICAL**	**HORIZONTAL**	**VERTICAL**
0	+	+	+	+
1 (rotation)	Fixed	Fixed	+	+
2	-	-	+	+
3 (crane)	+	Fixed	+	+
4	Fixed	+	+	+
5	+	+	-	+
6	+	+	+	-

Measure (line)



Ratio Pixel (float) : X and/or Y dimensions are multiplied themselves and by this ratio to produce one surface unit displayed according to the title format (ex : 0.0).

Font Size (integer) **:** contains the font size of the displayed value for surface.

Delta X1 / **Delta Y1** (integer) : contains offset of point 1.

Delta X2 / **Delta Y2** (integer) : contains offset of point 2.

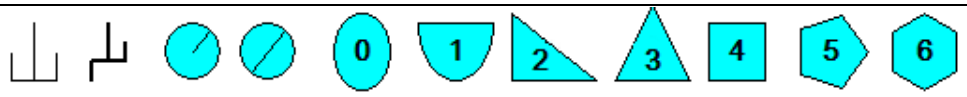
Note : Measure are displayed with arrow at the ends of line. To avoid arrow you can introduce "-" character at the beginning and/or end of value format in title (ex: -0.0-).

Bezier (line)

Display a « Bezier » (2 start-end points + 2 control points).

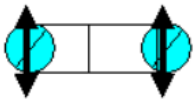
To adjust control points, select object and press space bar to make (dis)appear control points. Then move points with arrow keys combined with [SHIFT](p1) or [CTRL](p2).

Offset AX1 / AY1 / BX1 / BY1 (integer) : offsets of control points 1 et 2.



Rotation 2D

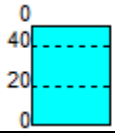
	<u>4 integer variables</u>
[R2D]	ROTATION ANGLE contains the angular value in degree for the effect of rotation.
[POSXY]	**HORIZONTAL** : object is moved by X horizontal pixels (Up=+) **VERTICAL** : object is moved by Y vertical pixels (Right=+) **BORDER MODE**: 0=border of circle is display, 1=border isn't display Or **ANIM_TYPE** : define geometric form of object (see numbers above).



ConveyorXY

ConveyorTêta

Conveyor	<u>5 integer variables</u>
[R2D]	**ROTATION ANGLE** contains the angular value in degree for the effect of rotation.
[POSXY]	**For XY style**, points 1 and 2 can be moved by **DELTA_X1/2** and **DELTA_Y1/2**. For Têta Style **FIX POINT(%)** : conveyor rotation point expressed in percent of length **FIX ANGLE(Deg)** : conveyor rotation angle applied at rotation point **PRODUCT START/END(%)** : conveyor product expressed in percent of length **Product Color** + **Layer2..5** : if different from "page color", conveyor product thickness is 1 to 5 pixels multiplied by thickness of object



Courbes / Curves

	<u>5 variables (all standards) + 3 colors</u>
[VAL]	**ABSCISSA** contains the display position in pixels of the points compared to the origin (modulo the length of the axis to the screen). When the X limit is reach, curve is shifted and new graduations of ABSCISSA are displayed. For "CURVE_AUTO" model, this value in converted in hour depending from the base time **BT (ms)**. **VALUE 1/2/3** contain the co-ordinates of the displayed points according to the indicated **MAXIMUM** (in proportion of). Each value has its own displaying color. When a graphic page is activated, the last available 1000 values are displayed. For "CURVE_AUTO" model, Maxi and Mini values are stored in the label formatted "**Maxi/Mini**" (ex: 100.0/-100.0) For « CURVE_PC » values are given in 0..100% scale and Y-values aren't displayed. The label (ex : 2/1) give the number of Y parts (2) and if X-values are displayed (1) or not(0) at the bottom of the curve.

Sons / Sounds

	1 binary variable
[WAV]	**LAUNCH** : on rising edge, the WAV file is played in background.

Images / Pictures

	5 integer/real variables
[BMP] **[POSXY]** **[LGXY]** **[R2D]**	**HORIZONTAL** and **VERTICAL** contain image movements in pixels (+ towards the right and the top). **RATIO_X.Y(%)** : contain increase/reduce factor of the image (4 digits per axis – high bytes for X axis). **Seize value 0100 0100** by default for a 100% ratio X and Y. **ROTATION POINT**: from 0 (or NULL) to 8 (see schema below). **ROTATION ANGLE**: contain the rotation angle in degrees. It is apply according to the rotation point. **Note** : All BMP/GIF/JPG/PNG/TIF/EMF files are accepted.

GIF animés

	2 binary variables + 1 real variable + 2 integer variables
[BMP]	**START** (binary) : used as command without effect on GIF display. **STATE** (binary) : if True, GIF pictures are displayed according to INTERVALLE time. **INTERVALLE** (real) : is expressed in second for GIF pictures display rythm. **IMAGE NUM** (integer) : if STATE is FALSE, this variable can drive the picture number that is displayed. Equal to -1 means that GIF is erased. If STATE is TRUE, it receives the displayed picture number (optional). **CMD MODE** (integer) : 0=object forced by one push, 1=object forced by continuous action.

GROUP

	1 binary variable + 2 integer variables
[DIS] **[POSXY]**	**HIDE** : allow to hide from screen or display on screen grouped object. **HORIZONTAL** : objects are moved by X horizontal pixels (Up=+) **VERTICAL** : objects are moved by Y vertical pixels (Right=+)

VIDEO

[VID]

5 integer variables + video.AVI or .WMV filename

ID : n° 0 to 9 of VideoPlayer. One player can be used for few videos (see function VideoLoad).

STATE : variable for VideoPlayer control and information

b0 : initialize VideoPlayer « Pro32Video.EXE » on rising edge or change filename
b1 : command Play (1) or Stop (0)
b2 : 1= automatic rewind command to the beginning at the end of video
b3 : 1=manual rewind command to the beginning
b4 : 1=modify timer position of VideoPlayer (with TIME parameter)
b5 : 1=modify display position & ratio (with SCREEN and RATIO parameters)
b6 : 1= close and free the VideoPlayer
b7 : scheduling bit (optional). Used to receive time and position of VideoPlayer
b8 : 1=VideoPlayer is launched
b9 : Play (1) / Stop (0) state
b12 : 1=new time position available
b13 : 1=new screen position available. Use GetSX, GetSY, …functions

SCREEN*10-POS : screen number * 10 + window position

0 : position of PROSIMUL object
1 : top left corner 2 : top right corner
3 : bottom left corner 4 : bottom right corner
5 : centred 6 : full screen
+100: video is displayed with borders +200: borders and progress bar
Ex : W_VIDEOPOS=125 → video with borders on screen 2 and centred

RATIO : contain display factor in **percent**

TIME : control or state of time position expressed in **1/10ème second**

Example: one VideoPlayer used for 2 videos

```
If Fe B_VIDEO2 Then : W_EXEC = VideoLoad (("PageVideos"; "Wildlife" ) : End_If
If Re B_VIDEO2 Then : W_EXEC = VideoLoad (("PageVideos"; "Test" ) : End_If
W_VIDEO_STATE . 0 = Re B_VIDEO_PLAY
W_VIDEO_STATE . 1 = B_VIDEO_PLAY
W_VIDEO_STATE . 2 = 1
W_VIDEO_STATE . 3 = Re B_VIDEO_REWIND
If Fst W_VIDEO_TIME_1\10S Then : W_VIDEO_STATE . 4 = 1 : UnF W_VIDEO_TIME_1\10S : End_If
If Fst W_VIDEO_RATIOPC Then : W_VIDEO_STATE . 5 = 1 : UnF W_VIDEO_RATIOPC : End_If
W_VIDEO_STATE . 6 = B_VIDEO_EXIT
W_VIDEO_STATE . 7 = SYS_TOP_DIXIEME And B_VIDEO_PLAY
If W_VIDEO_STATE . 13 Then : W_VIDEO_WIDTH = GetSX ("PageVideos" ; "Wildlife" ) : End_If
```

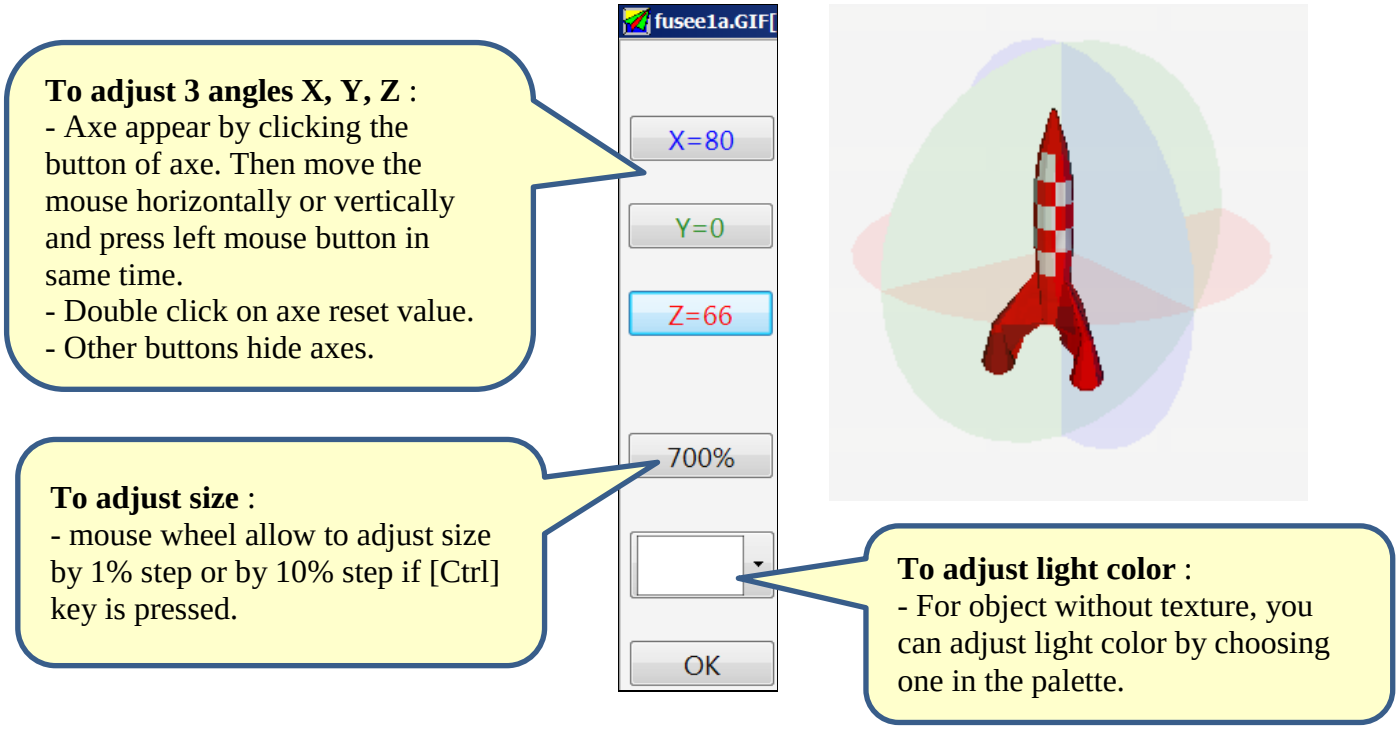
NOTA: Pro32Video.EXE player can be started alone with following parameters :
1st parameter (must be present) : video filename (.AVI or .WMV).
ID0…ID9 : ID of VideoPlayer if used by PROSIMUL
M1 M5 : screen number P0 P6 : window position (as defined in SCREEN-POS)
%vvv, Xvvv, Yvvv : display ratio and window X, Y position on screen

❑ 3D OBJECT

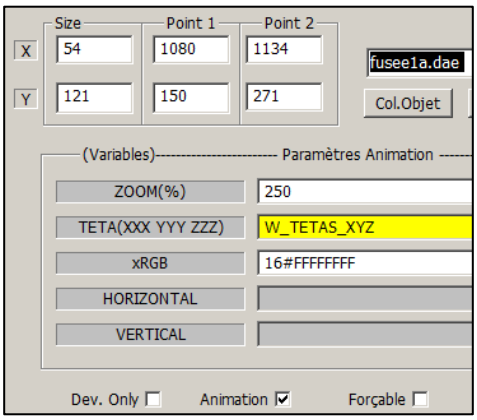
Select the 3D file (must be on .SIM application directory).

- ⇨ Recognized file : .ASE, .DAE (Collada) et .OBJ.
- ⇨ Max pixel dimensions : 1024 pixels X 768 pixels
- ⇨ Colours : 32 bits.

Using 3D object cause start of viewer named Pro3D.exe (that must be on same directory as Pro32.exe). In STOP mode, initial size and angles (X,Y,Z) are adjusted thru viewer and exchanged with PROSIMUL. Only simple object can be used to avoid too slow display.

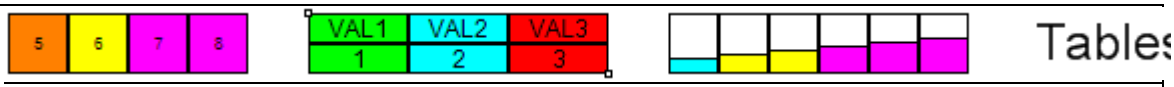


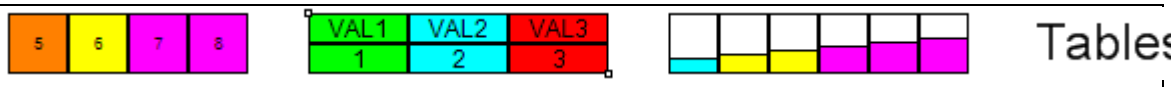
In RUN mode, animation is adjusted thru 5 variables (integer) :
- **ZOOM(%)** : store or give the growth in % of the object
- **TETA (XXX YYY ZZZ)** : contains X,Y,Z angle values (3 digits by angle – Ex : 90 000 000 → X=90° - Y and Z = 0°)
- **xRGB** : contains light color in 4 bytes format [Transparency][Red][Green][Blue] (ex : 16#FFFFFFFF = white)
- **HORIZONTAL** et **VERTICAL** allow object displacement on the screen (in pixels).



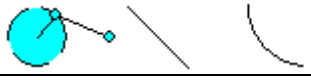

```
If SYS_INIT_APPLI Then
    W_ANGLE_X = W_TETAS / 1000000 :
    W_ANGLE_Y = ( W_TETAS % 1000000 ) / 1000
    W_ANGLE_Z = W_TETAS % 1000
End_If
If Re SYS_TOP_DIXIEME Then :  W_TETAS = W_ANGLE_X * 1000000 + W_ANGLE_Y * 1000 + W_ANGLE_Z : End_If
```

NOTE: Pro3D.exe « viewer » can be use in standalone mode by giving name of 3D file as first parameter (.ASE, .DAE ou .OBJ file). It can be used to control correct display of object.

FILL VALUE	
[CF][VAL]	<u>1 bool/int variable + 1 int/float value + 3 int values</u> **REFRESH** : when changing state allow the table refresh. **TABLE** (LEN=First Element) : the first integer or float variable of the table. This first element must contain the number of consecutive elements (integer or float). Elements of table are displayed (values for "Table Value") with colours depending from **MINIMUM** / **THRESHOLD** / **MAXIMUM**.

SYMB BAR TETA	
[CF][VAL]	<u>1 bool/int variable + 1 bool/int/float value + 3 int values</u> **REFRESH** : when changing state allow the table refresh. **TABLE START** : the first variable of the table. **TABLE LENGTH** : the quantity of elements in the table. Data are displayed with colours depending from **THRESHOLD1** / **THRESHOLD2** for **SYMB** or **BAR** tables or with selected colours by 20% steps for **TETA** table. For **BARH** / **BARV** / **TETA** tables % of display depends from Max and Min value seized in title (ex1: 100.0/0.0 ex2: 0.5/-0.5 ex3: 100/-100) **TETA** table can use a rotation point **OFFSET XY** and a rotation angle **ANGLE(°)**. Rotation point is calculated like below (with integer values) : RP.X = MiddleObjectX + OFFSET_XY/10000 – 1000 RP.Y = MiddleObjectY + OFFSET_XY%10000 – 1000 Ex: OFFSET_XY = 10001000 ➔ rotation point is at center of table Ex: OFFSET_XY = 09001000 ➔ rotation point is left shifted of 100 pixels

	Rempli / Fill
Filled [CF]	<u>5 integer variables</u> **BACKGROUND COLOR** contains the display color value (0 to 15) of the filling in the selected style. The border color remains the stop color of the filling. **BORDER** : Empty or **0** : The filling operation operate on every closed surface. **1** : A rectangular border is draw and fill. **2** : A circular border is draw and fill. **3** : A star border is draw and fill. **10**: The fill colour follow the colour present at the center point of fill object. **OFFSET COLOR** : (a) if not empty the value is subtract from **BACKGROUND COLOR**. **MODULO COULEUR** : modulo operation on BACKGROUND COLOR after operation (a). **MINI COLOR** : value finally added to BACKGROUND COLOR. → Final color = MINI + Mod(BACKGROUND – OFFSET , MODULO)

ROD CRANK	 Mécanique
[R2D] **[POSXY]**	<u>1 integer variable + 2 output integer variables for positions returns</u> **ROTATION ANGLE** : contain the angular value of the driving shaft. **X1**(Output) / **X2**(Output) / **Y1**(Output) : those 2 returns variables contains the value of circular movement (X1,Y1) and translation movement (X2). The total course is the sum of X1+X2.

LINK	Mécanique
[R2D] **[POSXY]**	<u>4 integer variables + 1 output integer variable</u> **Delta X1** / **Delta Y1** : contains offset of point 1 (X+ = right, Y+ = up). **Delta X2** / **Delta Y2** : contains offset of point 2. **LEN** (Output) : this return variable contain the value of length of LINK element.

HINGE	Mécanique
[R2D] **[POSXY]**	<u>4 integer variables + 1 input/output integer variable</u> **Delta X1** / **Delta Y1** : contains offset of point 1 (X+ = right, Y+ = up). **Delta X2** / **Delta Y2** : contains offset of point 2. **LEN** (Input/Output) : this variable contain the value of length of HINGE element. It can be writed by Prosimul if new calculated len is out of original size. The Len sign determine the Hinge angle when moving.

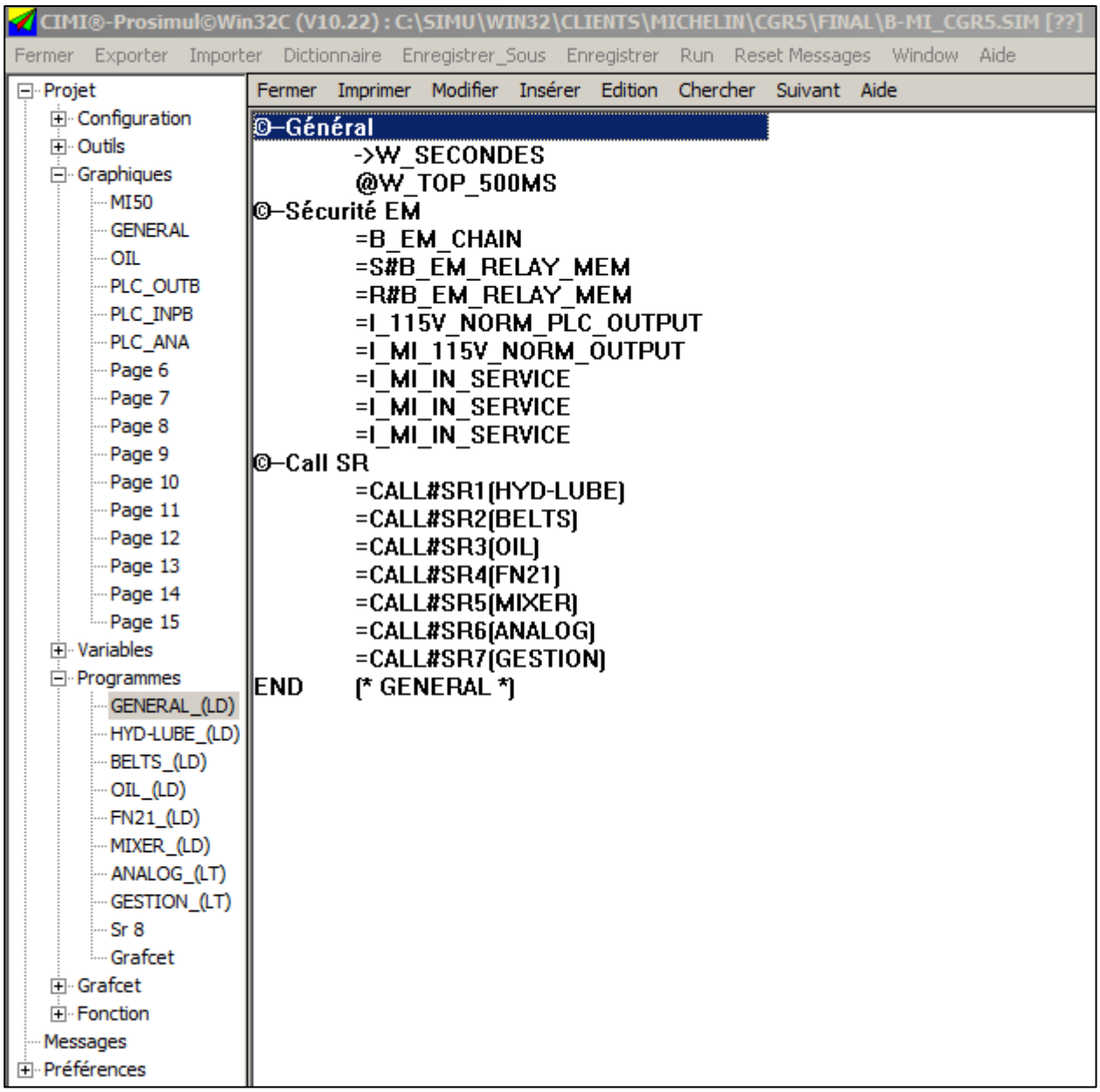
LADDER EDITOR

This editor has a simple instruction set (timer, counter, logical equation, block calculation) which can be seized through 1 Main program, 15 sub-routines and 1 specific program (treatment of the transitions/actions related to Chart if named “Grafcet”).

The programs are visualized in list mode (default) or in detailed instructions.

Each program can receive a label through navigator application.

Maximum of **512 instructions** per program.



LADDER : LIST MODE

Ladder Menu

Print : print, in a graphic form, all **selected** instructions.

Modify : entry in the modification screen of the selected instruction in the list (detailed viewing mode). **Double-click** on an instruction causes the same effect.

Insert : insertion of an empty instruction in front of the selected instruction.

Edit -> Cut : copy and clear selected instructions.
Edit -> Copy : store in buffer memory the selected instructions. The copied instructions could be exported .
Edit -> Paste : recopy of the instructions memorized in front of the 1st selected position.
Edit -> Paste with Increment : recopy of the instructions memorized in front of the 1st selected position with increment (if possible) of variables (example : FDCS_VERIN_01 becomes FDCS_VERIN_02 if the symbol exists consecutively in the base).
Edit -> Paste with Next : recopy of selected instructions memorized with following variables in database.

The Copy/Paste/Cut functions are accessible via the right click (finely contextual).

Search : search for a string in the comments or symbols of the different instructions. Seize the string. The first instruction containing the string is selected (research in all Main programs and SR).

Next : continuation of the same search.

Displaying Format of the instructions

©-- Comment
LABEL **[T]** VARIABLE/RESULT_ACTION

[T] represents

@	temporization
±	counter
=	logical equation
->	block calculation
©	Comment

Last instruction END (*End of program*) exists by default.

Colours of instructions an font size can be selected thru “Preferences” in project explorer.

Instructions Selection

- a simple click on an instruction causes its selection. Direction keys and wheel mouse allows to move in the list. [CTRL] key associated with wheel has same action than right and left keys.

- a simple click with **[SHIFT]** on an instruction causes an extend selection until the selected instruction.

- a simple click with **[CTRL]** allows a multiple selection of non-consecutive instructions.

LADDER : INSTRUCTIONS

Comment

Each instruction can receive a comment of 28 maximum characters. This comment is optional.

Label

Each instruction can receive 4 maximum characters used for the **JUMP**. This label is optional.

Variables Seizure

- Zones of variables seizure can :
  . be empty (state 0 - **grey bottom**),
  . contain immediate values (0,1, or numeric/real values - **white zone**),
  . receive symbols resulting from the base (**yellow bottom**). The symbol can **be seized directly or** reached in the base by a simple click on its information text (above). **A right click** on the edition zone allows a choice among the 10 last seizures (floating menu),
  . receive operators, jump actions or program calling (**blue bottom**).

- Types of indicated variables for the different instructions must be respected.

- For **read binary variables,** additional operations (rising or downward front) can be used :

. syntax of complement :	**/#**VARIABLE
. syntax of rising front :	**P#**VARIABLE
. syntax of downward front :	**N#**VARIABLE
. syntax for testing state 1 of force bit:	**F#**VARIABLE

*NOTE : **rising and falling edges** are true during only 1 cycle of program.*

- For **written binary variables (results),** Set/Reset operations of variables, reset of force bit, jump and call of program can be used :

. syntax of Set operation :	**S#**VARIABLE
. syntax of Reset operation :	**R#**VARIABLE
. syntax for unforce bit :	**U#**VARIABLE
. syntax for program jump (*for equation and block calculation*) :	**JUMP#LABEL**.
. syntax for subroutine calling (*for equation and block calculation*) :	**CALL#SR1..15** **CALL# MYPROG**

Shorcuts for using variable in editors

- **[SHIFT]** key associated with [/], [P], [N], [F] enable to use “Not, “Rising Edge”, “Falling Edge” and “Test force state” to the used variable.
- **[SHIFT]** key associated with [S], [R], [U] enable to use “Set, “Reset” and “Unforce” function to the used variable.
- All those shorcuts are available thru mouse right-click.

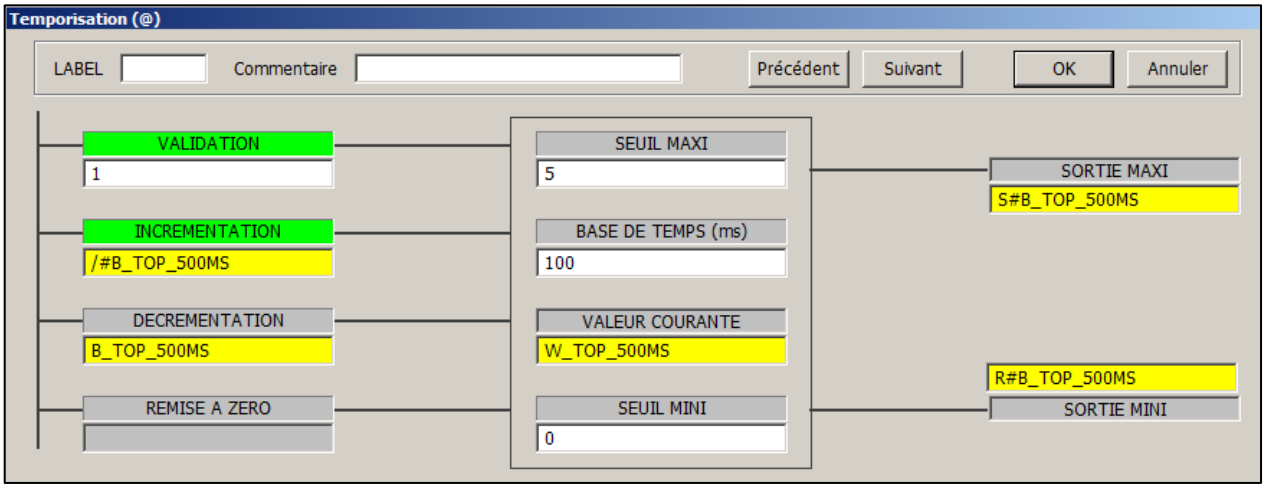
- In exploitation these variables will be animated.

Validation

- [OK], [Next] and [Previous] buttons validate the instruction in progress if all the fields of variables are correctly indicated.

- **[Next]** and **[Previous]** buttons cause moreover the passage to the previous or next instructions of the lists mode.

TEMPORIZATION



CURRENT VALUE : written integer variable containing the temporization value.

TIME BASIS : value of time basis in millisecond.

MAX THRESHOLD : value of the maximum threshold of temporization (blocking with beyond).

MIN THRESHOLD : value of the minimum threshold of temporization (blocking in on this side).

VALIDATION : binary condition allowing the functioning of the temporization and the activation of the maximum and minimum outputs.

INCREMENT : binary condition allowing the positive flow of the timer.

DECREMENT : binary condition allowing the negative flow of the timer.

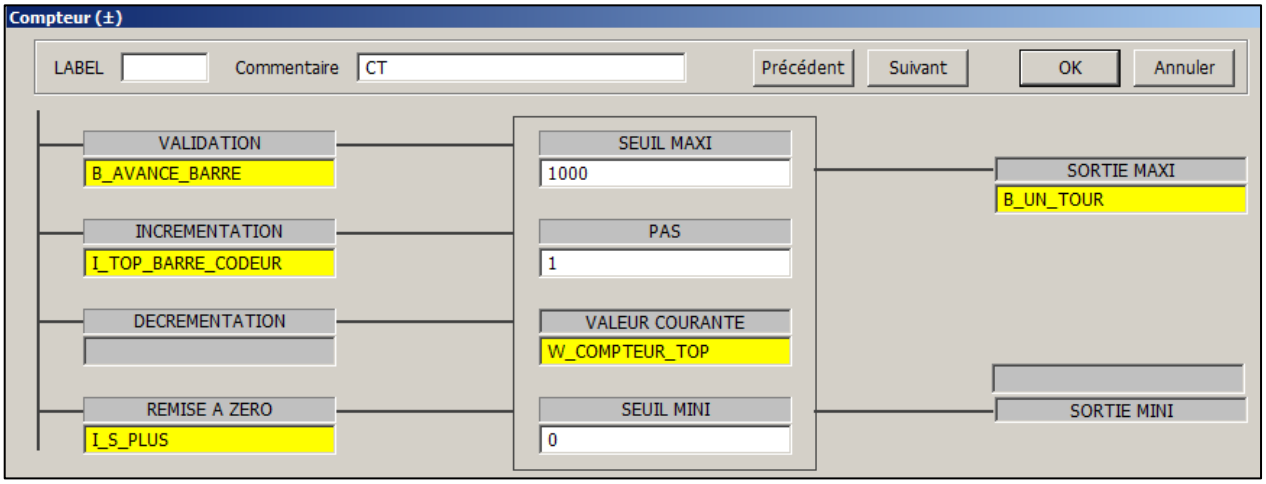
RESET : binary condition causing a reset with the minimum threshold of the current value (priority).

MAX OUTPUT: executed action or binary variable written in 1 if *current value > = maximum threshold* and if *validation = 1*.

MINI OUTPUT: executed action or binary variable written in 1 if *current value < = minimum threshold* and if *validation = 1*.

NOTE : a temporization can count several base of time between 2 Prosimul cycles according to cycle time's in RUN (see system variables *)*

COUNTER



CURRENT VALUE : written integer variable containing the value of the counter.

STEP : value of increase/decrease of the counter.

MAX THRESHOLD : value of the maximum threshold of the counter (blocking with beyond).

MINI THRESHOLD : value of the minimum threshold of the counter (blocking in on this side).

VALIDATION : binary condition allowing the functioning of the counter and the activation of the maximum and minimum outputs.

INCREMENT : binary condition allowing the increment of the counter (counting is executed *on* ***rising front*** of the condition).

DECREMENT : binary condition allowing the decrease of the counter (the countdown is executed *on* ***rising front*** of the condition).

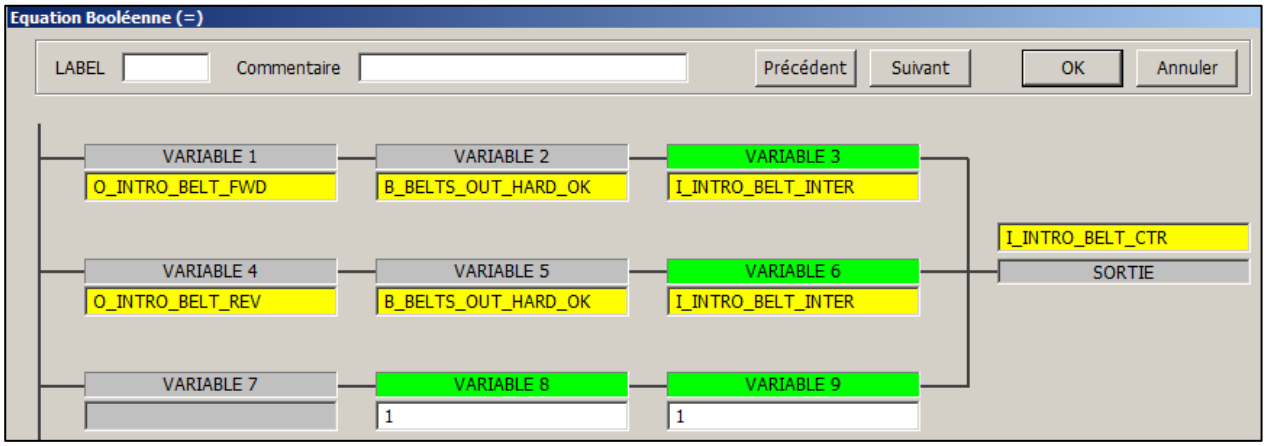
RESET : binary condition causing a reset with the minimum threshold of the current value (priority).

MAX OUTPUT: executed action or written binary variable in 1 if *current value > = maximum threshold* and if *validation = 1*.

MINI OUTPUT: executed action or written binary variable in 1 if *current value < = minimum threshold* and if *validation = 1*.

LOGICAL EQUATION

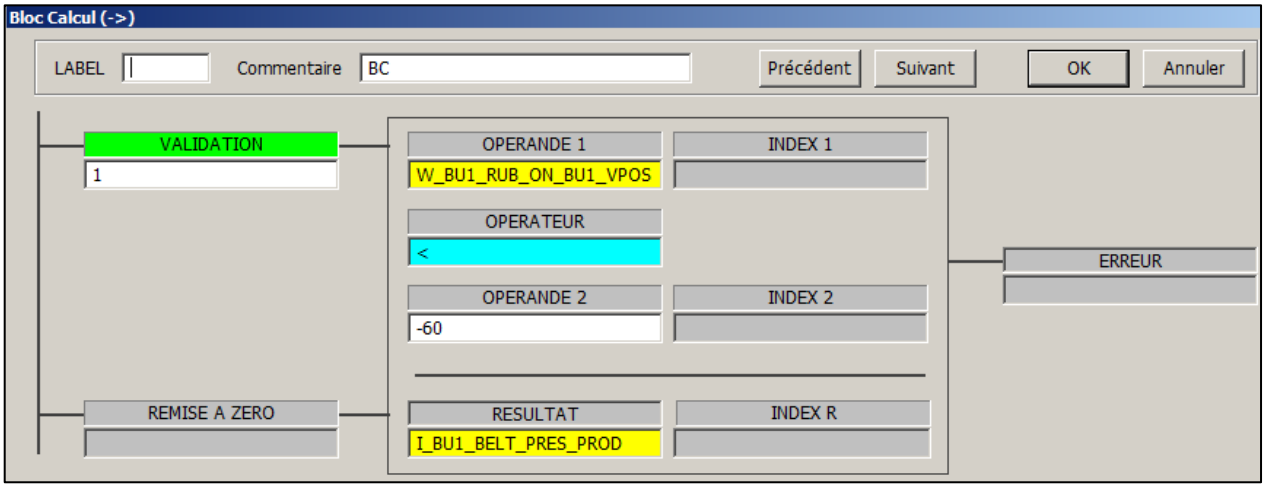
Boolean equation of fixed structure : 6 functions " AND " and 3 functions " OR ".



VARIABLE 1 TO VARIABLE 9 : binary conditions. For **always true conditions,** seize value 1.

OUTPUT : executed action or written variable in 1 if the logical equation is true.

BLOCKS CALCULATION



VALIDATION : binary condition allowing operation of the block calculation.

RESET : binary condition causing the passage to zero of the variable result.

ERROR : executed action or written variable in 1 in case of treatment error.

INDEX 1, INDEX 2, INDEX R : if used, these variables will contain the respective indices of Operand 1, Operand 2 and Result. These indices are *obligatorily* integer type.

NOTE : the current value of ***an index*** *acts as an address movement (+N position in the base - normally positive) compared to its associated variable. A overtaking control of the number of accessible variables (2048) is carried out. In event of use (except particular cases indicated) the positions in the base of operands 1 or 2 or the result will be shifted of index values.*

OPERANDE 1, OPERANDE 2, RESULT : variables of different types according to the operator used (Arithmetic , Logical , Mathematic , Flow-Moist Air , Comparison , Table , Conversion , Bit<->Word).

Operators and types of variables

[B] : binary value
[M] : integer or real value (word)
[Mi] : integer value (32 bits) or real value converted into integer implicitly.
[Mr] : real value or integer value converted into real implicitly.
[BM] : binary variables or words (agreement of the types must be respected normally)

ARITHMETIC OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Addition	[M][M]→[M]	+	OP1 + OP2
Subtraction	[M][M]→[M]	-	OP1 - OP2
Multiplication	[M][M]→[M]	*	OP1 * OP2
Division	[M][M]→[M]	/	OP1 / OP2
Absolute value	[M][M]→[M]	ABS	ABS(OP1)*OP2
Maximum	[M][M]→[M]	MAX	Maximum of OP1 and OP2.
Minimum	[M][M]→[M]	MIN	Minimum of OP1 and OP2.
Modulo	[M][M]→[M]	MOD or %	Remain of OP1 divided by OP2.
Random value	[Mi][Mi]→[M]	RND	OP1 <= Result <= OP1+ OP2.
Test Zero	[M][M]→[M]	DELTAZERO	OP2 if OP1<OP2 or –OP2 if OP1>OP2
Mean value	[M][M]→[M]	MEAN	(OP1+OP2) / 2

LOGICAL OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
AND	[Mi][Mi]→[M]	AND	OP1 AND OP2
OR	[Mi][Mi]→[M]	OR	OP1 OR OP2
Exclusive OR	[Mi][Mi]→[M]	XOR	OP1 XOR OP2
Left Shift	[Mi][Mi]→[M]	SHL	OP1 shifted on the left of OP2 bits (0 in)
Right Shift	[Mi][Mi]→[M]	SHR	OP1 shifted on the right of OP2 bits (0 in)
Left Circular Shift	[Mi][Mi]→[M]	ROL	OP1 shifted on the left of OP2 bits and reintroduced on the right
Right Circular Shift	[Mi][Mi]→[M]	ROR	OP1 shifted on the right of OP2 bits and reintroduced on the left.

MATHEMATIC OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Cosine	[Mr][Mr]→[M]	COS	cosine(OP1)*OP2 *(OP1 in degree)*
Exponential (base E)	[Mr][Mr]→[M]	EXPN	OP2 * e puissance(OP1/OP2)
Exponential (base 10)	[Mr][Mr]→[M]	EXP10	OP2 * 10 puissance(OP1/OP2)
Napierian logarithm	[Mr][Mr]→[M]	LOGN	OP2 * ln(OP1/OP2)
Logarithm base 10	[Mr][Mr]→[M]	LOG10	OP2 * log10(OP1/OP2)
Power	[Mr][Mr]→[M]	POW	OP1 with power OP2
Sine	[Mr][Mr]→[M]	SIN	sine(OP1)*OP2 *(OP1 in degree)*
Square root	[Mr][Mr]→[M]	SQRT	OP2 * sqrt(OP1)
Hypotenuse	[Mr][Mr]→[M]	HYPO	Sqrt (Pow(OP1;2) + Pow(OP2;2))
Tangent	[Mr][Mr]→[M]	TAN	tan(OP1)*OP2 *(OP1 in degree)*
Arc Tangent PI/2	[Mr][Mr]→[M]	ATAN	atan(OP1)*OP2 $-90° \leq RES \leq +90°$
Arc Tangent PI	[Mr][Mr]→[M]	ATGPI	atan(OP1)*OP2 $-180° \leq RES \leq +180°$

FLOW - MOIST AIR OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Vapour Pressure (Pa)	[Mr][Mr]->[M]	PVAPEUR_PATM_RS PVapPatmRs	Vapour pressure for atmospheric pressure OP1 (Pa) with OP2 (kg/kgas) absolute humidty
Saturation vapour pressure (Pa)	[Mr][Mr]→[M]	PSVAPEUR_TAH PSVapeurTah	Saturation vapour pressure at OP1 (°C) moist air temperature. Formula OP2 : 0: Pow(10 ;2.7877+(7.625*OP1)/(241.6+OP1) 1: 6,11*Pow(10;7.5*OP1/(237.3+OP1)) 2: Exp(46.784-6435.0/(273.15+OP1) - 3.868*Ln(OP1+273.15))/750*1000
Vapour pressure (Pa)	[Mr][Mr]→[M]	PVAPEUR_TAH_HR PVapeurTahHr	Vapour pressure at OP1 (°C) moist air temperature and for OP2 (%) relative moisture. (PSat with formula 2)
Air density	[Mr][Mr]→[M]	ROAIR_PAS_TABS RoAirPasTabs	Air density at OP1 (Pa) pressure and OP2 (°C) temperature
Vapour density	[Mr][Mr]→[M]	ROVAPEUR_PV_TABS RoVapeurPvTabs	Vapour density at OP1 (Pa) pressure and OP2 (°C) temperature
Absolute moisture (kg / kg as)	[Mr][Mr]→[M]	RS_PV_PAS RsPvPas	Absolute moisture for OP1 (Pa) vapour pressure and at OP2 (Pa) pressure
Moist air enthalpy (kJ/kg)	[Mr][Mr]→[M]	HS_TAH_RS HsTahRs	Moist air enthalpy at OP1 (°C) temperature and OP2 (kg/kgas) absolute moisture
Enthalpy variation (kJ/kg)	[Mr][Mr]→[M]	HK_DT_RS HkDtRs	Enthalpy variation (kJ/kg) for OP1 (°C) delta T and OP2 (kg/kgas) absolute moisture
Air temperature (°C)	[Mr][Mr]→[M]	TAH_HS_RS TahHsRs	Air temperature for OP1 (kJ/kg) enthalpy and OP2 (kg/kgas) absolute moisture
Condensation temperature (°C)	[Mr][Mr]→[M]	TR_PV TrPv	Condensation temperature for OP1 (Pa) vapour pressure
Delta Pressure(K,Q)	[Mr][Mr]→[M]	DELTAP	=OP1 * Pow(OP2;2) * sign(OP2)
Flow(K,DP)	[Mr][Mr]→[M]	FLOW	=OP1 * Sqrt(abs(OP2)) * sign(OP2)
Celsius → Fahrenheit	[Mr][Mr]→[Mr]	Fahrenheit(OP1;OP2)	OP1*1.8 + 32.0 + OP2 (offset)
Fahrenheit → Celsius	[Mr][Mr]→[Mr]	Celsius (OP1;OP2)	(OP1 – 32.0) / 1.8 + OP2 (offset)

COMPARISON OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Superior	[M][M]->[B]	>	True binary variable if OP1 > OP2
Equality	[M][M]->[B]	=	True binary variable if OP1 = OP2
Inferior	[M][M]->[B]	<	True binary variable if OP1 < OP2
Superior or Equal	[M][M]->[B]	>=	True binary variable if OP1 >= OP2
Inferior or Equal	[M][M]->[B]	<=	True binary variable if OP1 <= OP2
Different from	[M][M]->[B]	<>	True binary variable if OP1 <> OP2
Interval 1..2..5..10 (i)	[M][M]->[B]	?1 ?2 ?5 ?10	True binary variable if : OP2-i <= OP1 <= OP2+i
Near from	[M][M]->[B]	><	True binary variable if Abs(OP1) <= OP2

ARRAY OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Copy a value into N variable	[BM][Mi]->[BM]	NCOP (not in ST)	OP1 -> RESULT[OP2]
Copy table in another table	[BM][Mi]->[BM]	TBCOP (not in ST)	OP1[OP2] -> RESULT[OP2]
Sum of table elements	[M][Mi]->[M]	TBSUM	Sum(OP1[OP2]) -> RESULT
Search a value in 1 table (equality) INDEX 1 must indicate the length of the table starting from OP1	[M][M]->[M] Note INDEX 1 and INDEX R of the type [Mi]	TBEQ	Search in table OP1[INDEX 1] of the 1st value equal OP2. If the value is found RESULT receives it and INDEX R receives the offset based on OP1 (-1 if not).
Search a value in 1 table (lower than) INDEX 1 must indicate the length of the table starting from OP1	[M][M]->[M] Note INDEX 1 and INDEX R of the type [Mi]	TBLT	Search in table OP1[INDEX 1] of the 1st value lower than OP2. If the value is found RESULT receives it and INDEX R receives the offset based on OP1 (-1 if not).
Search a value in 1 table (superior to) INDEX 1 must indicate the length of the table starting from OP1	[M][M]->[M] Note INDEX 1 and INDEX R of the type [Mi]	TBGT	Search in table OP1[INDEX 1] of the 1st value higher than OP2. If the value is found RESULT receives it and INDEX R receives the offset based on OP1 (-1 if not).

CONVERSION OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Binary to Gray	[Mi][Mi]->[Mi]	BTG	Value in Gray code of OP1 coded on OP2 bits.
Gray to Binary	[Mi][Mi]->[Mi]	GTB	Value in binary of OP1 coded on OP2 bits.
Binary to BCD	[Mi][Mi]->[Mi]	BCD	BCD value (16 base) OP1. A complement is carried out if OP2 is negative.
BCD to Binary	[Mi][Mi]->[Mi]	BIN	Binary value (bases 10) of OP1. A complement is carried out if OP2 is negative.
Integer To Float	[Mi][Mi]->[Mr]	FLOAT	Float value of OP1 / OP2.
Float To Integer	[Mr][Mr]->[Mi]	INT	Integer value of OP1 / OP2.

BIT ←→ WORD OPERATORS	OP1, OP2, R	Operator LADDER	Result
Word → Bits array	[Mi][Mi]->[B]	BIT (not in ST)	OP2 binary variables coming from OP1 (low bits in 1st) are stored at RESULT and next
Bits array → Word	[B][Mi]->[Mi]	MOT (not in ST)	OP2 binary variables from OP1 are gathered in RESULT (high bits set to 0 if necessary).
Writing a bit of word	[B][Mi]->[Mi]	EBM (not in ST)	Writing n° OP2 bit of variable RESULT with OP1 binary variable.
Reading a bit of word	[Mi][Mi]->[B]	LBM (not in ST)	Writing RESULT binary variable with the bit n° OP2 extracted from OP1 integer variable.

VARIABLE ACCESS OPERATORS	OP1, OP2, R	Operator LADDER	Result
Initial value	[M][M]->[M]	WriteIni	The current and the initial values of OP1 receive the current value of OP2 RESULT receive the old initial state of OP1

FLOW / TEMP/ PRESSURE	OP1, OP2, R	Operator LADDER/ST	Result
Delta Pressure(K,Q)	[Mr][Mr]→[M]	DELTAP	=OP1 * Pow(OP2;2) * sign(OP2)
Flow(K,DP)	[Mr][Mr]→[M]	FLOW	=OP1 * Sqrt(abs(OP2)) * sign(OP2)
Celsius → Fahrenheit	[Mr][Mr]→[Mr]	Fahrenheit(OP1;OP2)	OP1*1.8 + 32.0 + OP2 (offset)
Fahrenheit → Celsius	[Mr][Mr]→[Mr]	Celsius (OP1;OP2)	(OP1 – 32.0) / 1.8 + OP2 (offset)

INFORMATION OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Object size	[Mi][Mi]→[Mi]	GetSX GetSY	Dynamic X or Y sizes for an object draw on page OP1 with number OP2.
Object position	[Mi][Mi]→[Mi]	GetLeftX GetRightX GetTopY GetBotY	Dynamic positions cadre for an object draw on page OP1 with number OP2.
Object coordinates	[Mi][Mi]→[Mi]	GetX1 GetY1 GetX2 GetY2	Dynamic coordinates of points 1 and 2 for an object draw on page OP1 with number OP2.

Note : take care of graphics modification (ex : object order) that modify number objects.
To avoid this problem, OP2 can be given by the label of the object with quote delimiters (ex: GetSX("Synoptic";"Belt1"). For picture and video objects, name of file is given without it extension.

FRIGO (*) OPERATORS	OP1, OP2, R	Operator LADDER/ST	Result
Gas density	[Mr][Mr]→[M]	FRIGO_ROGAZ_T FrigoRoGazT	Gas OP1 density at OP2 (°C) temperature
Liquid density	[Mr][Mr]→[M]	FRIGO_ROLIQ_T FrigoRoLiqT	Liquid OP1 density at OP2 (°C) temperature
Saturation pressure (bar)	[Mr][Mr]→[M]	FRIGO_PSAT_T FrigoPsatT	Saturation pressure at OP1 fluid at OP2 (°C) temperature
Saturation temperature (°C)	[Mr][Mr]→[M]	FRIGO_PSAT_P FrigoPsatP	Saturation temperature at OP1 fluid for OP2 (bar) pressure
Gas enthalpy (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HTGAZ_T FrigoHtGazT	Saturation enthalpy of OP1 fluid in gas state at OP2 (°C) temperature
Liquid enthalpy (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HTLIQ_T FrigoHtLiqT	Saturation enthalpy of OP1 fluid in liquid state at OP2 (°C) temperature
Latency enthalpy (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HLAT_T FrogoHlatT	Latency enthalpy of OP1 fluid at OP2 (°C) temperature

(*) Available gas : R134a : OP1= 134.0. For other gases please contact us.

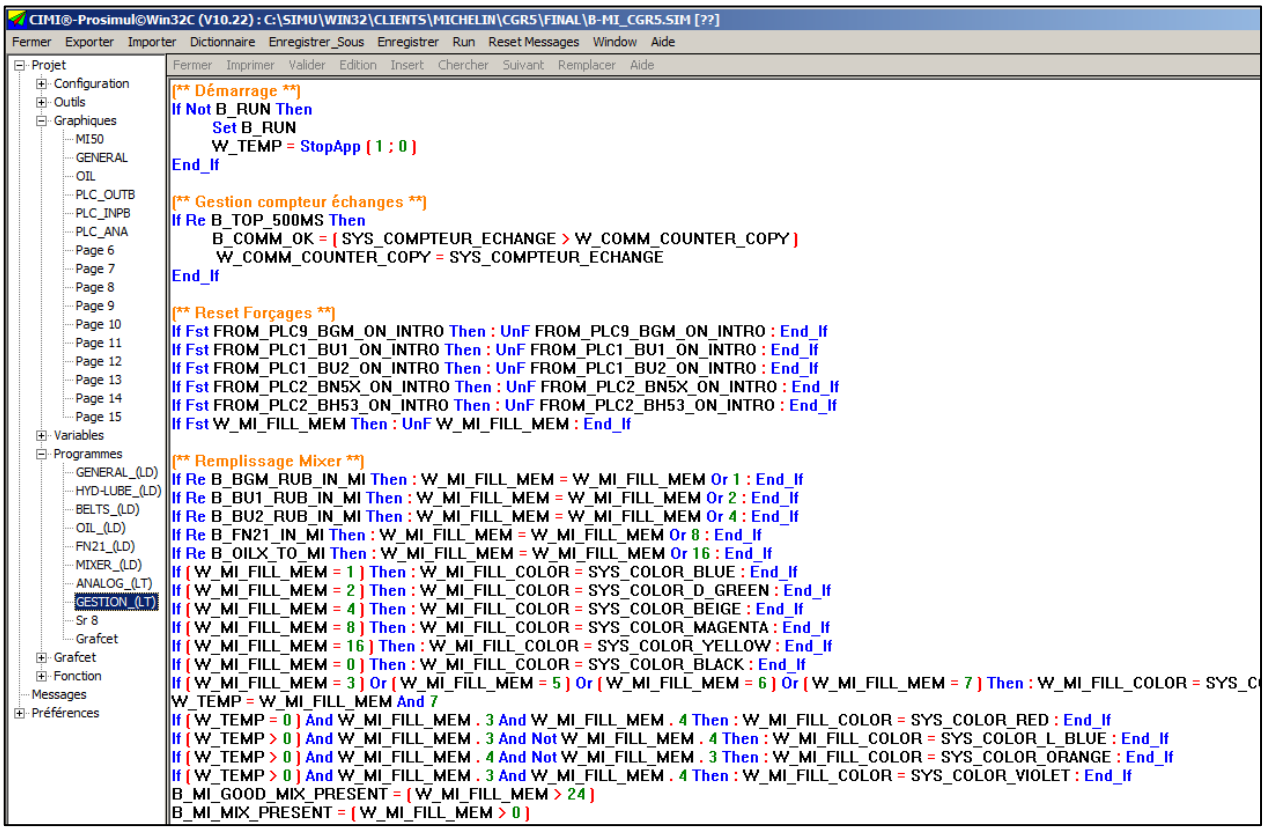
STRUCTURED TEXT EDITOR

This editor provide some instructions (struct IF , struct WHILE , struct FOR , Boolean calc , numeric calc and bit instructions) that can be used in the 15 sub-routines (the Main is always in Ladder) and in the specific program affected of the Chart treatments (actions and transitions if program is named “Grafcet”).

Both Boolean or numeric instructions can be used.

Otherwise, Literal language can be used indexed addressing , bits extracted from words and bits or variables arrays.

Maximum of **1024 lines of 1024 characters containing 15 operators** per program.
Total number of characters per program is 55.000 for a total number of 8192 words.



Structured Text Menu

Validate : launch a compilation of edited program. In failure case, a message is posted in the status bar and is recorded in the Errors.TXT file.
Insert → Keywords : automatically insert structure keywords IF…THEN…ELSE…END_IF , WHILE…DO…END_WHILE , FOR…TO…STEP…NEXT.
Insert → Functions: library of function
Insert → Commentary : insert structure (*…*).

Edit → Copy : store in buffer memory the selected text.
Edit → Paste : recopy of the text memorized to the selected position.
Edit → Cut : copy and clear selected text.
Edit → Cancel : cancel the last operation.

Search : search for a string in the text from the current position of the cursor.
Replace : function Search with replacement of string in the text.
Replace all : same than function Replace but replacement operate for whole text in all ST programs.
Next : repeat the same search or replacement.

Comment

Comments begin at the line containing **(*** and finish after the line including ***)**.
F1 and F2 keys are shortcuts to introduce or suppress begin and end of comment.
Also, a **double dash at end of line** allow to write comment (-- *Comment*).

Instruction separator

Instructions are normally separated by line feeds or by the character "**:**"

Indexed addressing

Variables can be used with addressing indexed by the syntax: :
VARIABLE , INDEX_WORD
Word index is necessarily an integer variable.
Example : Set MOTOR_01 , M10 (If M10 equal 3, puts to1 the 3rd variable following MOTOR_01).

Bit extracts from word

The bits of integer numerical variables can be individually used by the syntax :
VARIABLE_INT_NUM . RANK
Example : BIT08 = TRANS . 8 : transfer the bit of rank 8 of TRANS word on BIT08 variable.

Array

Variables can be used in a array form via the syntax :
VARIABLE [LENGTH]
Variables array can be used only in simple transfers instructions.
Example 1 : TRANS = CASE_01 [8] : transfer 8 bits from CASE_01 and next in the TRANS word by beginning with low bits.
Example 2 : ARRAY_1[10] = ARRAY _2[10] (Equivalent to TBCOP in LADDER)
Example 3 : ARRAY _1[10] = VAR (Equivalent to NCOP in LADDER)
Indexed addressing and ·bit extracts from word can be simultaneously used with array.
Example : WORD . 4 [4] = 5 (puts to1 the bits 4 and 6 of WORD variable and puts to 0 the bits 5 and 7)

Structure : IF <CONDITION> THEN <OPE1> ELSE <OPE2> END_IF

The structure is a condition structure.
If the Condition is verified, operations composing OPE1 are executed but not those of OPE2.
If the Condition is not verified, operations composing OPE2 are executed but not those of OPE1.
The structure may be used simply : IF <CONDITION> THEN <OPE1> END_IF.

Structure : WHILE <CONDITION> DO <OPE> END_WHILE

The structure is a loop structure with condition. The loop executes OPE operations as long as the CONDITION is true.

Structure : FOR <VAR=VALBEG> TO <VALEND> STEP <N> <OPE> NEXT

The structure is an increase or decrease structure according to a fixed step. The loop executes OPE operations from initial variable state VALBEG to includes final value VALEND.

Example – Sum of a word on 2 in an array of 20 :

```
SUM = 0
FOR INDEX = 0 TO 20 STEP 2
        SUM = SUM + VALWORD , INDEX
NEXT
```

Call function : CALL

Call function can operate with Sr number or program name between simple quotes.
Ex : Call Sr5 or Call ‘MyProg’

Simple transfers

Booleans and numeric transfers are made via the operator **=** .
The simple transfers do not allow operators' custom. They can be notably used for the array transfers and affectations of different types variables.
VAR_INT = VAR_FLOAT

Booleans Operators

Every operator must be surrounded with a space. Certain spaces are automatically introduced.
The useful boolean operators for the tests are by **priority order** :
Not : bit complement (unable on logical expression)
Re : Rising edge syntax
Fe : Falling edge syntax
Fst : test of the forcing state of the variable
< , > , = , >= , <= , <> : Value comparisons
?1 , ?2 , ?5 , ?10 : Test of specific interval. Example : VAR ?5 100 will be true for VAR between 95 and 105 includes.
><: Test of every interval. Example: B_INTERVAL = (W_VALUE – 100) >< 7 : is true if W_VALUE contain value between 93 to 107.
And : And function
Or : Or function
Xor : Exclusive Or function

Numeric operators **+, -, /, *, %** (modulo) can be used inside logical equation

Parentheses may be used to increase priority order.

Bits Operations

The useful boolean operators for writings are :
Set : Puts to1 the bit (exclusive for booleans variables)
Reset : Puts to 0 the bit (exclusive for booleans variables)
Unf : reset the forcing bit of the variable.

Example : reset a bit and its forcing

```
IF BIT_10 THEN
        Unf BIT_10 : Reset BIT10
END_IF
```

Numeric Operators

Every operator must be surrounded with a space. Certain spaces are automatically introduced.
The useful numeric operators are by **priority order** :

Priority 1: **Functions** (see above)
Priority 2: *** , /** executed from left to right
Priority 3: **+** , **-** executed from left to right
Priority 4: **AND** , **OR** , **XOR**.

The **parentheses** may be used to increase priority order.

Functions

The useful functions are identical to those of LADDER editor and are executed in the following order :
ABS, MAX, MIN, MOD, RND,
SHL, SHR, ROL, ROR,
COS, SIN, TAN, ACOS, ASIN, ATAN,ATGPI
EXPN, EXP10, LOGN, LOG10, POW, SQRT,
TBSUM,
BTG, GTB, BIN, BCD, INT, FLOAT
WRITEINI, GETSX, GETSY, GetLeftX, GetRightX, GetTopY,GetBotY, GetX1, GetY1, GetX2, GetY2

To use these functions, you must use the following syntax :
FUNCTION (OPERATOR1 ;OPERATOR2)
Example : RES = SHL(WORD 1 ; 8) . RES receive WORD1 moved of 8 bits to the right (0 are introduces on low bits).

Operators and function can be stacked (nested) on OP1 and OP2.

Following functions does exist only in LADDER editor :
- EBM, LBM : in Literal use bits extracted from words.
- BIT, MOT : in Literal use array of bits.
- TBGT, TBLT, TBEQ : in Literal use a index search onto array of variables.

Examples

Example 1 : index search

```
INDEX = 0 : FIND = -1
WHILE (INDEX < 10) And (FIND = -1) DO
        IF TAB , INDEX = SEARCHVALUE THEN
                FIND = INDEX
        END_IF
END_WHILE
(* Treatment of find value *)
IF (FIND > -1) THEN ….
```

Example 2 : creation of cadence bit

```
TEMPO_TOP = TEMPO_TOP  + SYS_CYCLE_PROSIMUL
TOP_CADENCE = (TEMPO_TOP >= TIME_MS)
IF TIME_MS THEN : TEMPO_TOP = Mod ( TEMPO_TOP ; TIME_MS ) : END_IF
(* Using of Top *)
If TOP_CADENCE And ….
```

Example 3 : creation of realtime base for « auto » curve

```
W_ELAPSED_MS = GetDateTime ( 10 ; W_NB_MS )
```

W_NB_MS : contain the period of time base in millisecond.
W_ELAPSED_MS: receive the number of elapsed period since the beginning of the day.

Function ConnectToTcp(LocalIP, CommTable[3])

This function allows to open TCP-IP connection.
Parameter LocalIP : depends on configured IP address (exchange configuration) : default (0) or preferences file (1..3).
Parameter CommTable : first word (INT) of the 3 words of table exchange.
1st word : IP target address in hexa form : 16#DDCCBBAA
ex : 192.168.1.180 = 16#B401A8C0)
2nd word : port number (ex : Modbus=502)
3rd word : connection timeout (ms)

Return value : 0..7=OK : socket number to use (max 8 connections)
-1=No Winsock / -2=NoSocket available /-3=Bad local IP / -4=Socket Error / -5=Bind Error / -6=Connect Error

Function Disconnect(NumSocket, AllSocket)

This function allows to close one or all TCP-IP connections.
Parameter NumSocket : socket number given by « ConnectToTcp ».
Paramètre AllSocket : if non 0, all connections are closed.

Return value : -1=OK / -2=bad connection number

Fnction ModbusReadRegister(NumSocket, CommTable[4])

This function allows to read registers from a Modbus connection.
Parameter NumSocket : socket number given by « ConnectToTcp ».
Parameter CommTable : 1st word of the 4 words of table exchange.
1st word : timeout (ms) + slave number x 1000
ex : 1010_D = slave 1 / timeout 10 ms
2nd word : first Modbus register address.
3rd word : number of 16 bits register + request code (1=bit / 3=Holding / 4=Input) x 1000
ex : 3002_D = request 3 / 2 registers16 bits
4th word : local address to store register + data type (Bit=0000_H / Int16=1000_H / Int32=2000_H / Float32=4000_H)
ex : 16#4014 = position 20 in Real values

Return value : 0=OK / -1= socket number error / -2=Write Error / -3=No Data / -4=Negative Answer

Function GetTcpPortState (Ip ; Port)

Parameter IP : 0..3 : **0** default IP address / **1..3** : one of the 3 addresses configured in preferences.
Parameter Port : TCP port number which must be tested
Return value : **5** = port is active / -1 : port is closed or inexistent

Function SendMail (n°X ; Server)

This function use *PRO32_Email_X.TXT* files located on the same directory of the .SIM application and formatted like below :

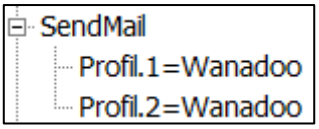
From: sender@cimi.fr
To: recept@cimi.fr
*Subject: Title of the email (**Current Date and Time** will automatically be added)*

Free body text (one blank line must exist like above) with dynamic variables..
HEURE=%MW28% ← one dynamic variable per line (rank is indicated between % characters)
MINUTE=%MW29%
DEBIT B2=%MF10% ← %MFXX% for real value, %MWXX% for integer value
The body must be terminated by a single point on the last line (see below)
Bye, bye.

.

Current Date and Time will automatically be added in the subject field.
Parameter X : is the end number of file *PRO32_Email_X.TXT*

V12 : two server profiles can be configured thru SendMail.exe application (*) and **profile names** must be provide in PROSIMUL **preferences** (PROSIMUL.INI file).
Parameter Server : is the used server profile number (**1 or 2**) given into preferences file (Profil.1 / Profil.2). 0 is same as 1 (for V11 compatibility).
(*) Server profile configuration: start SendMail.exe application, "Options" → "Configuration" → "Add/Modify" profile.



V11: Parameter Server : 0=SMTP protocol / 1=ESMTP protocol

Return value : 1 if OK, 0 otherwise (see LOG file)

Functions DialogOpen / DialogClose ("Page";"Dialog")

Those functions allow to start / stop by program dialog boxes configured on graphic pages. They replace mouse click on graphic object.
In case of success, the number of object is return otherwise value -1 is return.
Example :

```
If B_APPEL Then : W_APPEL = DialogOpen ( "Synoptique" ; "DIALOG_PID" ) : End_If
If B_CLOSE Then : W_APPEL = DialogClose ( "Synoptique" ; "DIALOG_PID" ) : End_If
```

Extern BOX functions BOX

It's possible to call (12 simultaneous max) exe files with data exchange (R/W) thru a share zone accessible with a DLL Pro32BoXDLL.dll which include 4 functions.

❑ Share memory structure

struct **pro32_box** {
boxzone BoxData[10]; };
 	
union **var32** {
__int32 i;
float f; };
	struct **boxzone** {
	HWND boxH; *// Box Handle stocké par l'exe*
	__int32 boxID; *// Box ID stocké par PROSIMUL*
	__int32 state; *// 0 initialement - Gestion libre*
	__int32 monitor; *// 0 initialement - Gestion libre*
	__int32 left; *// Gestion libre*
	__int32 top; *// Gestion libre*
	var32 var[10]; }; *// Gestion libre en int ou float*

❑ PROSIMUL : instruction usable in structured text program

W_ID = **BoxInit**(X ; BoxID) -- Call of **Pro32BoxX.exe** with user id **BoxID** with « Hidden » attribute.
Return value in W_ID:

- ✅ If call is OK : **value n** from 0 to 11 of activated box.
- ⚠ If BoxID is already use : Value of box + 20 (20 to 31)
- ⚠ Exe can't start : -2

If OK, Exe file start and initialize **BoxData[n].boxId** =**BoxID** and **BoxData[n].state** = **0**.

W_SEL = **BoxSelect**(W_ID ; -1) – Select Box Data zone with W_ID (0..11)
W_SEL = **BoxSelect**(-1 ; BoxID) – Select Box Data zone with BoxID (> 0).
W_SEL receive value 0..11 from selected Data zone or -1 if selection is impossible.
If OK **Select** = W_SEL (**Select** is internal to each application).

W_VAR = **BoxSelRead**(NUM ; 0) -- Reading variable NUM (depending from **Select** Data zone).
For 10 to 19, if NUM is integer, return will be interger, if NUM is float return will be float.

NUM=0	BoxData[Select].boxId
NUM=1	BoxData[Select].state
NUM=2	BoxData[Select].monitor
NUM=3	BoxData[Select].left
NUM=4	BoxData[Select].top
NUM=10…19 or 10.0 …19.0	BoxData[Select].var[NUM-10]

W_VAR = **BoxSelWrite**(NUM ; VAL) -- Writing variable NUM (depending from **Select** Data zone). For 10 to 19, if NUM is integer, return will be integer, if NUM is float return will be float.

W_SEL = **BoxClose**(W_ID ; -1) – Stop the Box with W_ID (0..9)
W_SEL = **BoxClose**(-1 ; BID) – Stop the Box with BoxID (> 0).
W_SEL receive -1 if stop is impossible or W_ID or BID if Box is unfindable.
If OK **BoxData[n].boxId** =**-1** et **BoxData[n].state** = **0** and a message WM_CLOSE is sended to the Box Handle. Then **BoxData[n].boxH** is forced to NULL.

User can develop its own closing or visibility state program using variable **BoxData[n].state**.

❑ PROSIMUL code

```
(*** Starting BOX ***)
If Re B_BOX_VISIBLE Then
    W_BOX99 = BoxInit ( 99 ; 100 ) : W_TEMP = BoxSelect ( W_BOX99 ; -1 )
    W_TEMP = BoxSelWrite ( 10 ; Int ( B_START_STOP ; 1 ) ) -- écriture du sélecteur binaire dans la 1ère variable User
    -- La position de l'objet appelant peut être retrouvé avec les fonctions GetLeftX et GetTopY ( avec leurs noms )
    -- La Box peut alors être positionnée à l'endroit voulu
    W_TEMP = BoxSelWrite ( BOX_MONITOR ; 1 )
End_If

(* Gestion courante *)
If Re SYS_TOP_DIXIEME And ( W_BOX99 > -1 ) Then
     W_TEMP = BoxSelRead ( BOX_STATUS ; 0 )
    If ( W_TEMP = WANT_CLOSE ) Then
        UnF B_BOX_VISIBLE : B_BOX_VISIBLE = 0
         W_BOX99 = BoxClose ( W_BOX99 ; -1 )
    Else
        W_TEMP = BoxSelect ( W_BOX99 ; -1 )
         W_TEMP = BoxSelRead ( 10 ; 0 ) -- lecture sélecteur StartStop
        B_START_STOP = W_TEMP . 0
        If B_START_STOP Then : W_TEMP = BoxSelWrite ( 11 ; SYS_HORLOGE_DIXIEME ) : End_If-- écriture de
l'horloge dans la 2ème variable User
         R_INPUT = BoxSelRead ( 12.0 ; 0.0 ) -- lecture SetPoint en réel
    End_If
End_If

(* Fermeture BOX *)
If ( SYS_CYCLE_PROSIMUL = -1 ) Or Fe B_BOX_VISIBLE Then
     If ( W_BOX99 <> -1 ) Then : W_BOX99 = BoxClose ( W_BOX99 ; -1 ) : End_If
End_If
```

❑ Client Application

Application receive the number of Data zone used by the Box (ex. **Pro32Box0.exe ID5**).
Library Pro32Box (.LIB + .DLL) must be linked to the project.
4 functions manage access to share data zone.

```
extern "C" __declspec(dllimport) bool _Pro32BoxInit(int Num, HWND hbox, int State);  //true if OK
extern "C" __declspec(dllimport) var32 _Pro32BoxGetData(int var);    //-9999 if error
extern "C" __declspec(dllimport) int _Pro32BoxSetData(int var, var32 val); //-9999 if error
extern "C" __declspec(dllimport) bool _Pro32BoxClose(); //true if OK
```

❑ Example of Box builded with C++ (C++ Builder)

```
//---------------------------------------------------------------------------

#include <vcl.h>
#pragma hdrstop

#include "main.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
#pragma comment(lib, "Pro32BoxDLL");

#define MAXBOX    12
#define BOXID                    0
#define STATE                    1
#define MONITOR   2
#define LEFT                     3
```

```
#define TOP                     4
#define VAL10I          10
#define VAL11I    11
#define VAL12R    12.0
#define TOCLOSE   0
#define UNVISIBLE 1
#define VISIBLE       2
#define WANTCLOSE       3

union var32 {__int32 i; float       f; };
struct infoecran {__int32 width; __int32 height; __int32 left; __int32 top; };
infoecran tabmon[5];
TForm1 *Form1;
HMONITOR hmonitor[5];
int Ecran=0,Px=0,Py=0,NbEcrans,boxId=-1,State=0,ValI=0,Id=-1;
float ValR=0.0;
bool IsPro32=false;
var32 ret;

extern "C" __declspec(dllimport) bool _Pro32BoxInit(int Num,HWND hbox,int State);  //true is OK
extern "C" __declspec(dllimport) var32 _Pro32BoxGetData(int Id,int var);    //-9999 if error
extern "C" __declspec(dllimport) int _Pro32BoxSetData(int Id,int var,var32 val); //-9999 if error
extern "C" __declspec(dllimport) bool _Pro32BoxClose(int Id); //true if OK

//---------------------------------------------------------------------------
__fastcall TForm1::TForm1(TComponent* Owner) : TForm(Owner)
{
Int NbParam=ParamCount();
NbEcrans=InfoEcrans();
Caption="Pro32Box99";
State=UNVISIBLE;
if (NbParam > 0) {
      int sp=1;
      AnsiString pn;
      while (sp <= NbParam) {
            pn=ParamStr(sp);
            if (LeftStr(pn,2) == "ID") Id=StrToInt(RightStr(pn,pn.Length()-2));
            sp++;
            }
      if (Id >=0 && Id < MAXBOX) {
            IsPro32=_Pro32BoxInit(Id,Form1->Handle,State);
            if (IsPro32) {
                  boxId=_Pro32BoxGetData(Id,BOXID).i; //read only - for use is necessary is the BoxN is is used several times with different BoxID
                  }
            else Caption=Caption + " Mapping_error";
            }
      else Caption=Caption + " Id error";
      }
Else {
      Caption=Caption + " No_param";
      Show();
      }
}
//---------------------------------------------------------------------------//-----------------------------------------------------------------------------
void __fastcall TForm1::CalcPos(void)
{
if (IsPro32) {
      ret.i=Px; _Pro32BoxSetData(Id,LEFT,ret);
      ret.i=Py; _Pro32BoxSetData(Id,TOP,ret);
      }
Ecran=Min(Ecran,NbEcrans);
Px=Px+tabmon[Ecran-1].left; Form1->Left=Px;
Py=Py+tabmon[Ecran-1].top; Form1->Top=Py;
}
//---------------------------------------------------------------------------//-----------------------------------------------------------------------------
void __fastcall TForm1::FormClose(TObject *Sender, TCloseAction &Action)
{
if (IsPro32) {
      State=_Pro32BoxGetData(Id,STATE).i; //from Pro32
      if (State > TOCLOSE) {
            State=WANTCLOSE; //dmde Close user
            ret.i=State; _Pro32BoxSetData(Id,STATE,ret);
            }
      }
else State=TOCLOSE;
if (State == TOCLOSE) {
      if (IsPro32) _Pro32BoxClose(Id);
```

```
	Action=TCloseAction::caFree;
	}
else Action=TCloseAction::caNone;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::Timer1Timer(TObject *Sender)
{
if (IsPro32) {
	if (State == UNVISIBLE) {
		Ecran=_Pro32BoxGetData(Id,MONITOR).i;
		if (Ecran > 0) {
			ValI=_Pro32BoxGetData(Id,VAL10I).i;
			if (ValI) B_Stop->Enabled=true;  else B_Start->Enabled=true;
			Px=_Pro32BoxGetData(Id,LEFT).i; Py=_Pro32BoxGetData(Id,TOP).i;
			CalcPos();
			State=VISIBLE;
			ret.i=State; _Pro32BoxSetData(Id,STATE,ret);
			Show();
			}
		}
	Else {
		ret=_Pro32BoxGetData(Id,VAL11I);  // récupération de l'horloge
		Horloge->Caption=IntToStr(ret.i);
		}
	}
}
//---------------------------------------------------------------------------
BOOL CALLBACK MonitorEnumProc(HMONITOR hmonitor,HDC,LPRECT r,LPARAM)
{
static int i=0;
MONITORINFO mi;
mi.cbSize=sizeof(MONITORINFO);
GetMonitorInfo(hmonitor,&mi);
if (mi.dwFlags == 1) i=0;
else i++;
tabmon[i].left=mi.rcWork.left;  tabmon[i].top=mi.rcWork.top;
tabmon[i].width=mi.rcWork.right-mi.rcWork.left; tabmon[i].height=mi.rcWork.bottom-mi.rcWork.top;
return(true);
}
//---------------------------------------------------------------------------
int __fastcall TForm1::InfoEcrans()
{
int i,nb;
nb=GetSystemMetrics(80);
for (i=0;i < nb;i++) {
	hmonitor[i]=NULL;
	tabmon[i].left=tabmon[i].top=0;
	tabmon[i].width=1280; tabmon[i].height=1024;
	}
EnumDisplayMonitors(NULL,NULL,MonitorEnumProc,NULL);
return(nb);
};
//---------------------------------------------------------------------------
void __fastcall TForm1::B_StartClick(TObject *Sender)
{
ret.i=1; _Pro32BoxSetData(Id,VAL10I,ret);
B_Start->Enabled=false; B_Stop->Enabled=true;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::B_StopClick(TObject *Sender)
{
ret.i=0; _Pro32BoxSetData(Id,VAL10I,ret);
B_Start->Enabled=true; B_Stop->Enabled=false;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::E_spKeyDown(TObject *Sender, WORD &Key, TShiftState Shift)
{
if (Key == VK_RETURN) {
	ret.f=_wtof(E_sp->Text.c_str()); _Pro32BoxSetData(Id,VAL12R,ret); // écriture 3ème paramètre en réel
	}
}
```

Function WinExec (n°X ; Display)

This function use *PRO32_WinExec_X.TXT* files located on the same directory of the .SIM application and containing one command line like below :
<DIRECTORY\APPLI.EXE> <DIRECTORY\PARAMETRE>
PROSIMUL directory could be writed <REPPRO> (without \) and SIM application directory (if same) could be writed <REPAPPLI>.
Complete path of PROSIMUL executable file is shorted by <APPPRO>
Sample 1 : C:\Program Files\Internet Explorer\iexplore.exe http://www.cimi.fr
Sample 2 : C:\SIMU\PRO32FR.EXE C:\SIMU\SUPERVISEUR.SIM
Sample 3 : <REPPRO>PRO32FR.EXE <REPAPPLI>SUPERVISEUR.SIM
Sample 4 : <APPPRO> <REPAPPLI>SUPERVISEUR.SIM

Parameter X : is number of file named *PRO32_WinExec_X.TXT*
Parameter Display : is the display mode of application
0 SW_HIDE
1 SW_SHOWNORMAL
2 SW_SHOWMINIMIZED
3 SW_SHOWMAXIMIZED
4 SW_SHOWNOACTIVATE
5 SW_SHOW
6 SW_MINIMIZE
7 SW_SHOWMINNOACTIVE
8 SW_SHOWNA
9 SW_RESTORE
10 SW_SHOWDEFAULT
11 SW_FORCEMINIMIZE
Return value : > 30 if OK, otherwise see LOG file.

Function MouseCtrl (param1 ; param2)

Param1 : type of executed operation
0 X coordinate is return
1 Y coordinate is return
2 max X coordinate is limited according to **param2** and new coordinate is return
3 max Y coordinate is limited according to **param2** and new coordinate is return
4 min X coordinate is limited according to **param2** and new coordinate is return
5 min Y coordinate is limited according to **param2** and new coordinate is return
10 return screen number on which mouse cursor is located

Function GetDateTime (param1 ; param2)

Param1 : indicate type of return value
0 Year
1 Month
2 Day of Month
3 Day of Week
4 Hours
5 Minutes
6 Seconds
7 Milliseconds (current)
8 Seconds elapsed in the week (starting at Monday 00:00)
10 Milliseconds elapsed in the day (since 00:00)
Param2 : if > 0 the value is subtracted from return value (for value 0 to 7 of Param1) or divide the return value (if Param1 = 10).

Function MsgLog (param1 ; param2)

Function used to write a user message in PROSIMUL.LOG file.
If OK, return Param1 value.
Param1 : line number (0..n) of text in MSGLOG.TXT file (located on .SIM directory)
Param2 : colour of the message in status bar.
One variable could be inserted in the message like **%MWx%** (integer x) or **%MFx%** (real x).
Ex : Temperature=%MW125%

Function StopApp (param1 ; param2)

Param1 : =0 : function return « Lock » state
=1 : to (un)lock application (**Param2** : 0=unlock or 1=lock)
=2 : to stop and quit application (**Param2** = delay in ms)
=3 : to stop and restart application (**Param2** = delay in ms)
=4 : application **Param2** is maximized and foregrounded.
=5 : to stop other Prosimul application indicated by **Param2** (0..n=1 application, 99=all applications, -99= abort all stops)
=6 : application **Param2** is minimized.
=7 : application **Param2** is hided.

Function return « Lock » state (P1=0 to 3) or application number (-1 if failing).

Function InfoApp (param1 ; param2)

Function used to read application state.
Param1 : =0 : function return application number
=1 : function return the number of loaded applications
=2 : function return the number of running applications
=3 : function return state Run(1)/Stop(0) of application indicated by **Param2**
=4 : function return window state of application (see WinExec)
=10 / 11 : function return X / Y origin of synoptic
if **Param2** = 1 screen origin is added; if **Param2** = -1, screen origin is subtracted.

Function ScreenParam (param1 ; param2)

This function allow to receive information on window or screen placement and allow to change window position.

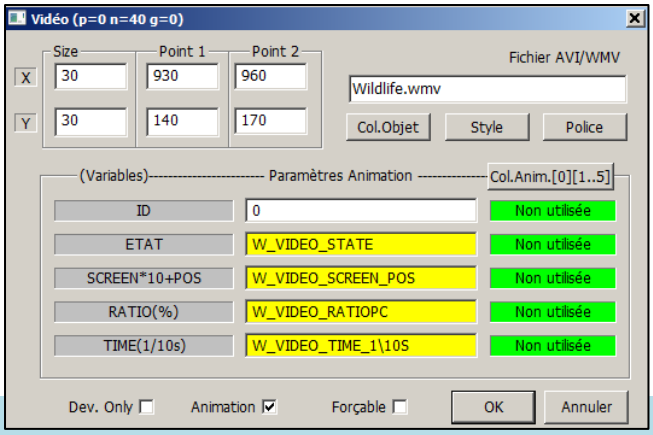
Param1	Param2	Retour	Action
0	0	Number of connected screens	-
0	1	N° of current screen (application)	-
1	0=appli	X left coordinate	-
2		Y top coordinate	-
3	1..5=	Width W	-
4	screen	Height H	-
5	0	Graphics display ratio (%)	
10	1..5 (screen number)	Screen obtained after operation	Application full screen
11			Application Top Left
12			Application Top Right
13			Application Bottom Left
14			Application Bottom Right
15			Application centred
21	0…n (X, Y, W, H)	X, Y, W, H value obtained after operation control	Move X left of application
22			Move Y top of application
23			Adjust Width W of application
24			Adjust Height H of application

Function VideoLoad (param1 ; param2)

Function used to change média FileName of video player.

Param1 : =n or “pagename” : page number or page name of graphic video object.
Param2 : = "filename" (between quotes and **without extension**) : file name of video object.
Return = ID of founded object (first parameter) or -1 if no object found or ID unconfigured.

Ex : W_VIDEO=**VideoLoad("Pvideo";"TheVideo")** : search on page “Pvideo” the object named « TheVideo.AVI » or « TheVideo.WMV » and prepare media file for VideoPlayer 0 to 9 as indicated in it first parameter. For VidePlayer control see « Vidéo ».
NOTA : if n Video objects use same VideoPlayer (for switch between videos), **only one must be animated** (with animation variables) and others mustn’t be animated but must have same ID in their first parameters (see below).



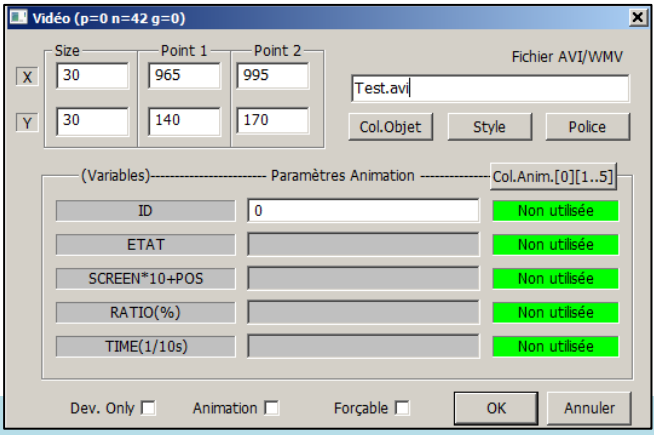
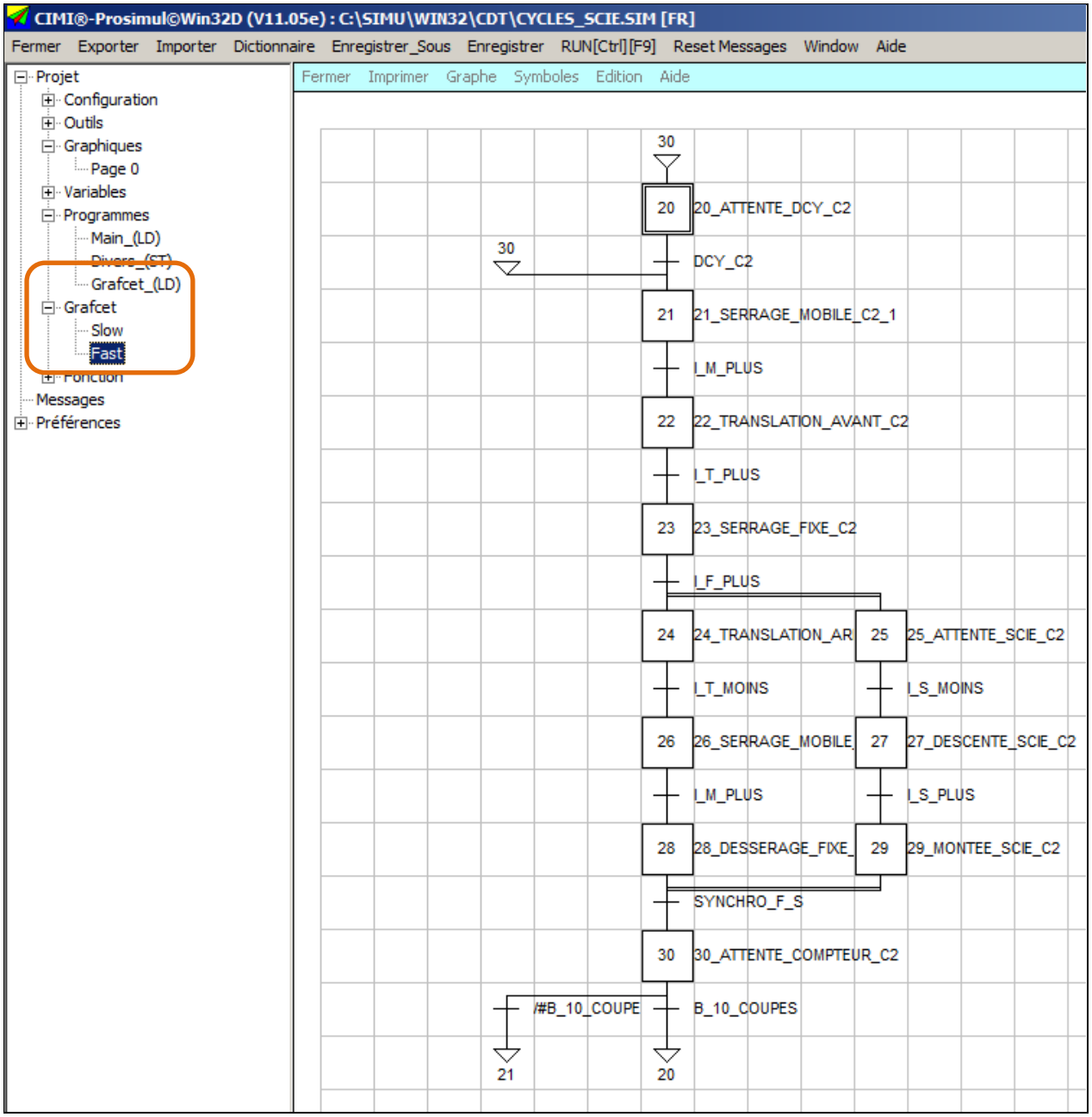


CHART EDITOR

This editor allows the construction of graphs on 8 pages of 26 X 20 cells receiving the symbols.

Maximum of 2048 steps including 32 active into simultaneous.

*NOTE : each step manages a single bit of step (no other action is possible) and each transition tests a single binary variable. A **zone of Ladder action named " Grafcet ", reserved for the treatment of the graphs, will have to be used to cause the actions from the step bits and to treat the transitions.***



Grafcet Menu

Graphic -> Check : consistency check of the graphs (structure, references...). See message zone.
Graphic -> Renumber : renumbering (top towards down and left towards right) selected steps and references.
Graphic -> Properties : information of the selected steps/transitions. *The double-click* on a cel causes the same effect.

Symbols -> Cancel : return to the cursor of selection. Identical action obtained with *the right click* on a cell of the graph.
Symbols -> Initial Step : select the cursor to place initial steps.
Symbols -> Step : select the cursor to place simple steps.
Symbols -> Transition : select the cursor to place transitions.
Symbols -> Link to Destination : select the cursor to place reference of destination.
Symbols -> Link from Origin : select the cursor to place reference of origin .
Symbols -> AND Function : select the cursor to make divergences and convergences in AND (simultaneous parallel actions).
Symbols -> OR Function : select the cursor to make divergences and convergences in OR (exclusive parallel actions).
Symbols -> Down links : select the cursor to place down links .

NOTE : steps, transitions and references must be informs .

Edit -> Copy : store in buffer memory the selected cells. The copied cells could be exported .
Edit -> Paste : recopy of the instructions memorized to the selected position (top left corner).
Edit -> Cut : copy and clear selected cells.

*NOTE : all these functions are accessible via **the right click** (contextual menu). Steps and transitions can be moved by mouse after selection*

Information of steps / transitions / references

- After symbol placement or Graphic->Properties or double-click on cell, information of associated variable is proposed for the steps, transitions and references.

- In the 3 cases you can :
  . Seize its associated symbol (modifiable in the base).
  . Select its symbol in the base via simple click on the text " symbol/number ".
  . A right click on the edition zone allows a choice among the 10 last seizures.

- Specifically **for the steps and references** :
  . The variables of " Steps " type are **obligatory**.
  . The numbers of steps must be single.
  . You can directly seize their **numbers** (0... 2047) instead of symbol.

Function AND and OR

- Select the corresponding cursor.
- **AND Divergence :** select the origin transition and after the final step for connection.
- **AND Convergence :** select the final step and after the transition for connection.
- **OR Divergence :** select the origin step and after the transition for connection.
- **OR Convergence :** select the origin transition and after the step for connection.

NOTE : in all cases, couples of connections step/transition will be distant of a line to the maximum. Foresee references if necessary.

References of destination and origin

- references are used for connections between steps/transitions from different pages or to prevent an absence of directed links. References function per pairs.
- references of **destination** are positioned behind a transition. Indicate the number of destination step.
- references of **origin** are positioned in front of a step. Indicate the number of origin step (located directly at the top of the starting transition).

A Group Symbols Selection

- Click on the top-left corner, maintain the click and move the mouse towards the down-right corner.
- After relax, all included cells in the zone are red. Cut/copy operations can operate above it.

Dynamics visualization

In exploitation the graphs are displayable and forceable in dynamics.

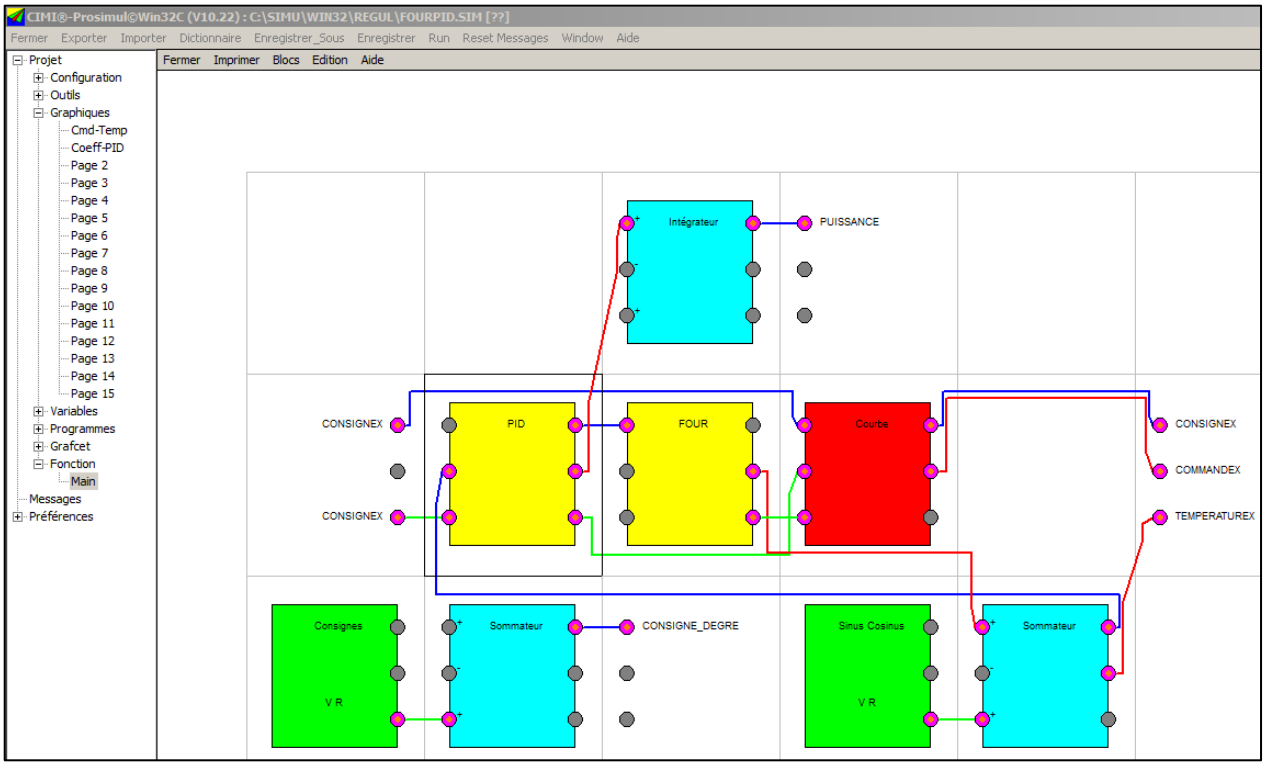
FUNCTION EDITOR

This editor allows creation of processes models ("continuous" type). This can imbricate functions blocks resulting from different families (sources , circuits , systems , logics , organization , outputs) subjected to one **period of execution** (discretized signals). Some parameters allow to adjust block function.

Two specific blocks " Wave Pulses " and " Bode " allow the layout of Bode diagram

Each page has 6 x 4 cells.

The blocks " sub-page " allow to structure the applications on 4 levels of depth. They have 3 entrance points and 3 exit points of values to connect levels between them.



*NOTE : **Blocks calculation** is executed from left to right and from top to bottom. In a recurring way, " sub-page " blocks ran in the same way as soon as they are met in the higher levels. Resulting values from the blocks are " pull " by the consuming blocks and not " put " by the producing. This involves a shift of a turn of " period " for the pulled values from right towards left.*

A maximum of 800 blocks function is limited.

Function Menu

Block -> Create/Inform : after a selection of a blank cell this menu gives access to blocks pallet.
Double-click on blank cell of the Graphic page causes the same action. If the cell is not empty the dialog box of parameter appears.
Block -> Zoom In : access to a " sub-page ".
Block -> Zoom Out : exit of a " sub-page ".
Block -> Period : change " period " parameter for the blocks of current page. The sub-pages are affected too.
Block -> Saturation : change " saturation/maximum "parameter for all the selected blocks.

Edit -> Copy : store in buffer memory the selected cells. The copied cells could be exported .
Edit -> Paste : recopy of the instructions memorized to the selected position (top left corner). The connections with the other blocks are reconstructed if necessary. If the polynomial identification tool of monitoring menu were used, the resulting polynomial is paste).
Edit -> Cut : copy and clear selected cells.

*NOTE : all these functions are accessible via **the right click** (contextual menu). Blocks can be moved by mouse after selection*

Group Blocks Selection

- Click on top-left corner, maintain the click and move the mouse towards the below-right corner.
- With the relaxation, all cells include in the zone become red. Cut/copy operations can be use on.

Connections between functions blocks

The connections between blocks are done exclusively between pairs points of entry and exit.

- **Create a connections** : when mouse is on a connection point, the cursor changes into " cross 1 ". Click on the point and **maintain the mouse** up to he second point. The cursor is transformed into " cross 2 " below the 2nd point. Relax mouse click, the connection is created.

- **Clear a connection** : click on one of the 2 points of connection. The removal of a block involves the obliteration of its connections.

Common Parameters of functions blocks

NOTE : all blocks parameters can receive immediate values or real variables from the base (converted in all cases into real).

Comment : text of 19 characters maximum (displayed in the block).

Duration of samples(sec) : execution period (required) expressed in **second**. In case of period overtaking in RUN a message indicates it. Adjust PROSIMUL cycle time .

Block Gain : multiplying gain applied at block exit.

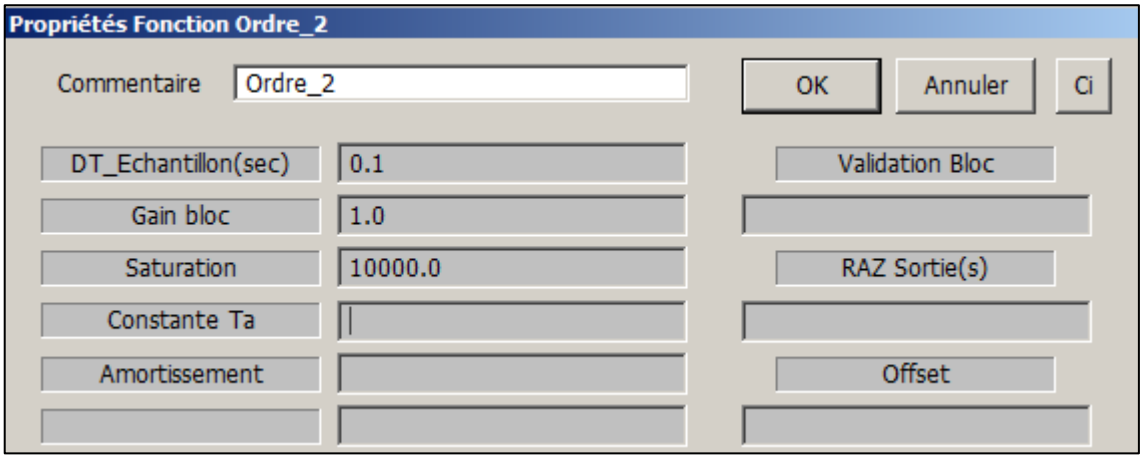
Saturation : maximum absolute value at block exit.

Validation Block : binary condition allowing or not the periodic execution of the block (*by default a block is permanently validated*).

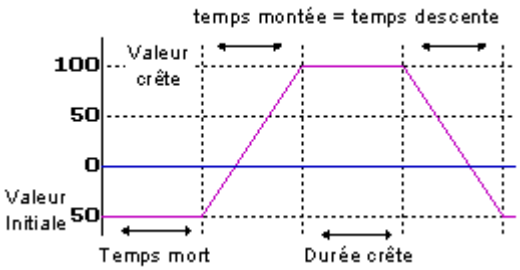
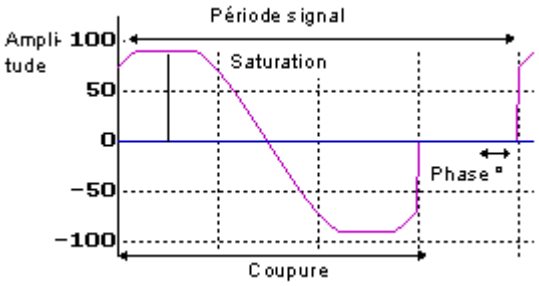
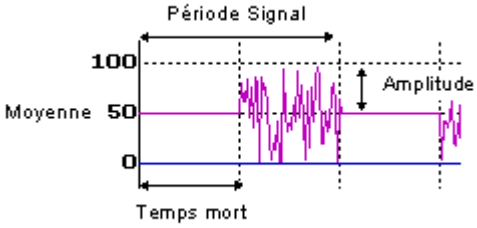
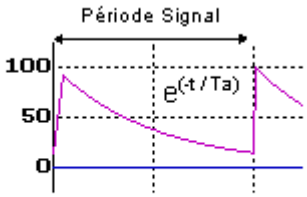
Ouput(s) reset : binary condition allowing a forced reset of block exits.

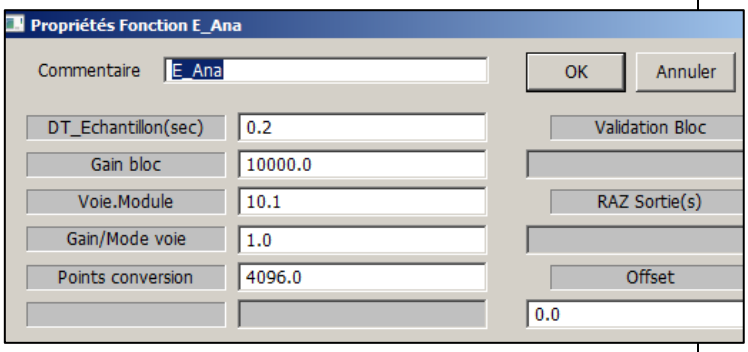
Offset (or Coefficient) : if present, this word variable works by default on offset of the block output.

Variable base 1/2/3 : for **the input /output blocks bases**, these variables are used to make internal exchanges of values to the blocks from/to the data base (normally of real type).



Source blocks parameters

Blocs	Signal Parameters	Inputs Outputs
Square-triangle		- No input - 3 identical outputs
Sine-cosine		- No input - 3 identical outputs
White noise		- No input - 3 identical outputs
In_Base	Variables base 1/2/3	- No input - Outputs = Variables
Exponential		- No input - 3 identical outputs

In_Ana	**According to configured cards** - Gain Block K_B : value full scale - Channel . Module : n° of channel and n° of card EASYPORT=0..3 •1..4 SEALEVEL=1..16•1..4 PIO-DAxU=0..1•0..n - Gain/Mode channel : gain of signal LABPC+ : 1, 1.25, 2, 5, 10, 20, 50, 100 ACL8112 : 1, 2, 4, 8, 16 EASYPORT : **1.0** (unused) SEALEVEL : **1.0** (unused) PIO-DAxU: **1.0** (unused) - Conversion Points (2048, 4096,..,32767) EASYPORT : 2048 (11 bits) SEALEVEL : 4096 = 12 bits unipolar –for bipolar positive values 0..2047 and negative 4095..2048) - Offset: added/subtract optionally to value.  	- No Input - 3 identical outputs =$EAna_{(GE)}$*K_B/Points
Instructions	Generation of instructions - Initial Value of the outputs - Period : starts again a global cycle - 12 couples of Time, Value instructions (access by button [Ci]) - *Seize **last time** with value **-1.0***	- No input - 3 identical outputs
Wave Pulse	Used to produce Bode diagram - a Number N of sample per pulsation (360°) - a Number of periods I (pulsations) before error - Omega Min (rd/s) - Steps by LOG10 decade - Omega Max (rd/s) - Amplitude (/2 at each crossed decade)	- No input - Output Pulsation Pi = Amplitude * sin(ω.N.Te)

Circuit blocks parameters

Blocks	Signal Parameters	Entrées Sorties
Integrator	- Integral of 0 with n.T_E of the sum of inputs - initial Value of the integrator - Saturation - Gain K - Initial value (can be non zero) - Minimum (don't decrease below)	- Inputs +, -, + - 3 identical outputs
Differentiator	- Derived over the period from the sum of inputs - Saturation - Gain K	- Inputs +, -, + - 3 identical outputs
Summator	- Sum of inputs (0 if no connection(s)) - Saturation - Gain K - Initial value (can be non zero) - Minimum (don't decrease below)	- Inputs +, -, + - 3 identical outputs
Multiplier	- Input1 / Input 2 * Input 3 (1 if no connection(s)) - Saturation - Gain K - Initial value (can be non zero) - Minimum (don't decrease below)	- Inputs *, /, * - 3 identical outputs
Delay	- Delay of the input signals of N periods of sampling (**1..28**) - Saturation - Gain K - The values can be writed in the table **Values**	- 1 input - Output = Lag
Threshold	- Solidifying with 0 of the sum of inputs between ± low Threshold (shifted outputs of ± low Threshold if not) - Saturation - Gain K	- Inputs +, -, + - 3 identical outputs

Power	- Calculation of power N of the sum of 3 inputs - Saturation - Gain K - Exponent N	- Inputs +, -, + - 3 identical outputs
Minimum	- Calculation of the minimum of 3 inputs (not connected = ∝) - Saturation - Gain K	- 3 identical outputs
Maximum	- Calculation of the maximum of the 3 values of entry (not connected = -∝) - Saturation - Gain K	- 3 identical outputs
Average	- Moving average of N (1..28) values - Saturation - Gain K - Initial state - Parameter « Start elements » contain number of initial values (0 ..N) to take when system start (0 by default).	- 1 output Average

NOTE : F(p) functions are calculated (z^{-1} operator) with order 0 blocker.

System blocks parameters

Blocks	Signal Parameters	Inputs Outputs
Order 1	- Function F(p) = **K / (1+Ta . p)** - Gain K - Constant Time Ta - Saturation - Initial value (can be non zero) - Minimum (don't decrease below)	- Inputs +, -, + - 3 identical outputs
Order 2	- Function F(p) **. K / ($Ta^2.p^2$ + 2.m.p + 1)** **if** ***0 < m < 1*** **. K / $(1 + Ta.p)^2$** **si** ***m = 1*** **. K / (1 + $(m+(m^2-1)^{0.5}).Ta.p)$*** **(1 + $(m-(m^2-1)^{0.5}).Ta.p)$** **if** ***m > 1*** - Gain K - Constant Time Ta - Damping m - Saturation	- Inputs +, -, + - 3 identical outputs
High Pass Filter	- Function F(p) = **K * Ta.p / (1 + Ta.p)** - Gain K - Constant time Ta - Saturation	- Inputs +, -, + - 3 identical outputs
Lead / Lag phase	- Function F(p) = **K * (1 + Tb.p) / (1 + Ta.p)** - Gain K - Constants time Ta, Tb - Saturation	- Inputs +, -, + - 3 identical outputs
PIDP Controller (parallel structure)	- Function F(p) = **K + 1/Ti.p + Td.p / (1 + Td/N.p)** - Gain K - Constants time 1/Ti, Td - Differentiator filter N - Saturation Min and Max : integrator stop working below or above those values	- Inputs +, -, + - 3 same outputs - P/I/D values are available for R/W**(*)** thru button Vi
PIDM Controller (mixed structure)	- Function F(p) = **K (1 + 1/Ti.p + Td.p / (1 + Td/N.p)** - Gain K - Constants time 1/Ti, Td - Differentiator filter N - Saturation Min and Max : integrator stop working below or above those values	- Inputs +, -, + - 3 same outputs - P/I/D values are available for R/W**(*)** thru button Vi

(*) PID values are written from external variables if they exist (value 0 by default in case of reset) only when variable « Manual Vi | Raz Out » is in "True" state.

Polynomial	- Polynomial $P(X) = \Sigma C_i * X^{Ni} + Initial_Value$ (12 max time) - Indicate **-1** for the last Ni (access by button [Ci])	- Inputs +, -, + - 3 identical outputs
Phase shift not min.	- Function F(p) = K * (1 - Tb.p) / (1 - Ta.p)2 - Gain K - Constants time Ta, Tb - Saturation	- Inputs +, -, + - 3 identical outputs
Conveyor	- Delay of the input signals of **N periods** of sampling (**1..24**) according to direction (MForward/MBackward) and with management of accumulation if null speed - **Saturation** clip output (loss of matter). - **Gain** is distributed on N elements (nième square of Gain for each). - **Coefficient** : the coefficient speed of the conveyor in integer format between –24 and +24 (0 = stop, < 0 MBack, > 0 MFor). 1 = one shift by period - The internal values can be writed in the table **Values** **NOTE**: **period** can also be variable to allow only one shift function (Coefficient=1) with various speed.	- Inputs : Q_iMFor, Unused, Q_iMBack - Outputs Q_oMFor, Total Charge, Q_oMBack
Screw	- Delay of input signal of **N period** (1 à 27). - **Saturation** limit output, but the excess E= (Nième value – Saturation) is distributed on N previous values (Vi) with decreasing algorithm (reflux of matter). Ei= E / ((N+1)*N/2) * i. - **Gain** is distributed on all N éléments (nième square of Gain). - The internal values can be writed in the table **Values** **NOTE**: **period** can also be variable to allow only one shift function (Coefficient=1) with various speed.	- Input : Qin - Outputs : Unused Total Charge Qout

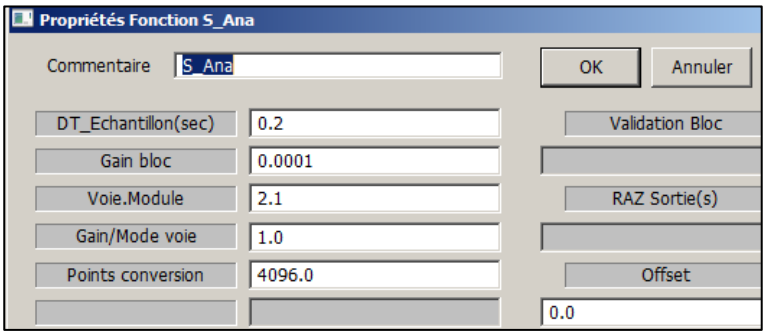
Logical blocks parameters

Logical	Signal Parameters	Inputs Outputs
OR	- OR Function between inputs (0 if no connection(s)) - Gain K - Saturation	- 3 identical outputs
AND	- AND Function between inputs (1 if no connection(s)) - Gain K - Saturation	- 3 identical outputs
NOT	- NOT Function for each input - Gain K - Saturation	- 1 output per input

Organization blocks parameters

Blocks	Signal Parameters	Inputs Outputs
Sub-Page	- Sub-page Function (accessing by Zoom In or simple click in RUN mode) - Gain K - Saturation	- inputs of “ sub-page ” - outputs of“ sub-page ”
Copy	- Copy of input to the 3 outputs - Gain K - Saturation	- 1 input - 3 identical outputs
Multiplexer	- Multiplexing of 2 values (input 2 or 3) according to a threshold compared with input 1 - Gain K - Saturation (Threshold)	- 3 identical outputs (input 2 if i1 >= threshold or input 3 if i1 < threshold)
Split-Range	- Gain K - Saturation - **D**irect/**R**everse : condition S1/S3 function : Fast/Slow (same direction) or Warm/Cold (opposite direction) - Low Threshold (LT) + High Threshold (HT) → S2 < SB Direct : S1=S2*Gain+[Offset] Reverse : S1=(LT – S2)*Gain+[Offset] → LT < S2 < HT Direct : S1=LT*Gain+[Offset], S3=0 Reverse : S1=S3=0 → S2 > HT S3=(S2 - HT)*Gain+[Offset] Reverse : S1=0 Direct : S1 = LT*Gain+[Offset]	- S1 : Cold or Slow action - **S2** = sum of inputs (+, -, +) - S3 : Warm or Fast action
Share	Allow to share a proportion of inputs (**Ratio Out (0..1)**) in 2 parts depending from **Coeff Out(%)**. Qe = Min((E1-E2+E3)*Gain ; Saturation) Qr= Qe*(1- RatioOut*CoeffOut / 100) * ScaleRemain Qo= (Qe – Qr) * ScaleOut	- Inputs Ei +, -, + - Qr - Qo - Qo

Output blocks parameters

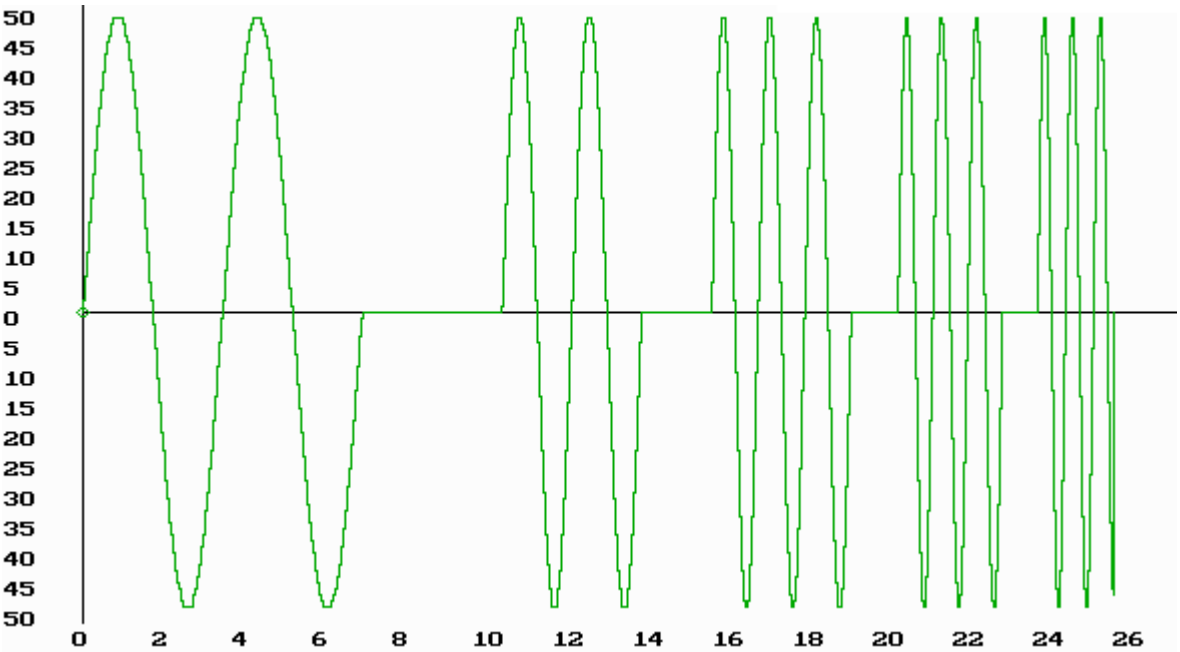
Blocks	Signal Parameters	Inputs Outpus
Curve	- inputs*K Viewing in exploitation - Gain K - Saturation (maximum ordinate)	- isolated inputs - outputs = inputs.K
LED	- 3 LEDS Viewing (1 if input*K >= Saturation) in exploitation - Gain K - Saturation	- isolated inputs - outputs = LED state (0 ou 1)
Out_Base	- Loading of variables base 1/2/3 with iputs values*K - Gain K - Saturation	- isolated inputs - outputs = inputs.K
Out_Ana	**According to configured cards** - Gain K_B : adjusted for Input * K_B =1.0 - Channel . Module : n° of channel and n° of card EASYPORT=0..1 •1..4 SEALEVEL=1..2•1..4 - Channel Gain/Mode : 1.0 (unused) - Conversion Points (2048, 4096,..,32767) EASYPORT : 2048 (11 bits) SEALEVEL : 4096 = 12 bits unipolaire - Offset: added/subtract optionally to value. 	- 1 input - 3 identical outputs = Input * K_B * Points
Bode	Used to produce Bode diagram - Coefficient N of the filter (> 1 – 100=wihtout) - acceptable Error of integrals variation - Scale of phase layout (in degree) and the gain (in dB) in the form of mini/maxi. The launching of the layout is executed by the " Zoom In " function if the block is correctly connected.	- Input 1 : reference signal (resulting from pulsation). - Input 2 : signal to be analysed. - Output, the block transmits the values of the filtered signal, its derivative and its integral.

BODE diagram – Method

In the case of simulated models, the transmission of sinusoidal signals in input of those does not pose any problem of physical behaviour. Moreover, as all functions used are discrete, the duration of test can be singularly shortened by the acceleration of the internal time on which the treatments are dependent. Thus a signal *sin(0.001 t)* could be sent in the form of samples $sin(\omega.n.T_E)$ to the same rate as a signal *sin(1000 t)*, provided that the discrete systems receiving it are regulated over the same period of sampling. Note that fictitious time generated will offer an incomparable precision for the treatment of the blocks function.

The adopted method consists, to enter in the system a continuation of pulsation $sin(\omega.N.Te)$, by a specific source box. Pulsations interval (ω), numbers of samples signal and gain could be adjusted according to needs. In order to ensure the continuity of the signal on several sendings, **the integer number of sample,** forming the pulsation.

A maximum number of sending of pulsation will be indicated, in order to stop application since the output signal could not be identified (too weak gain or too important delay).

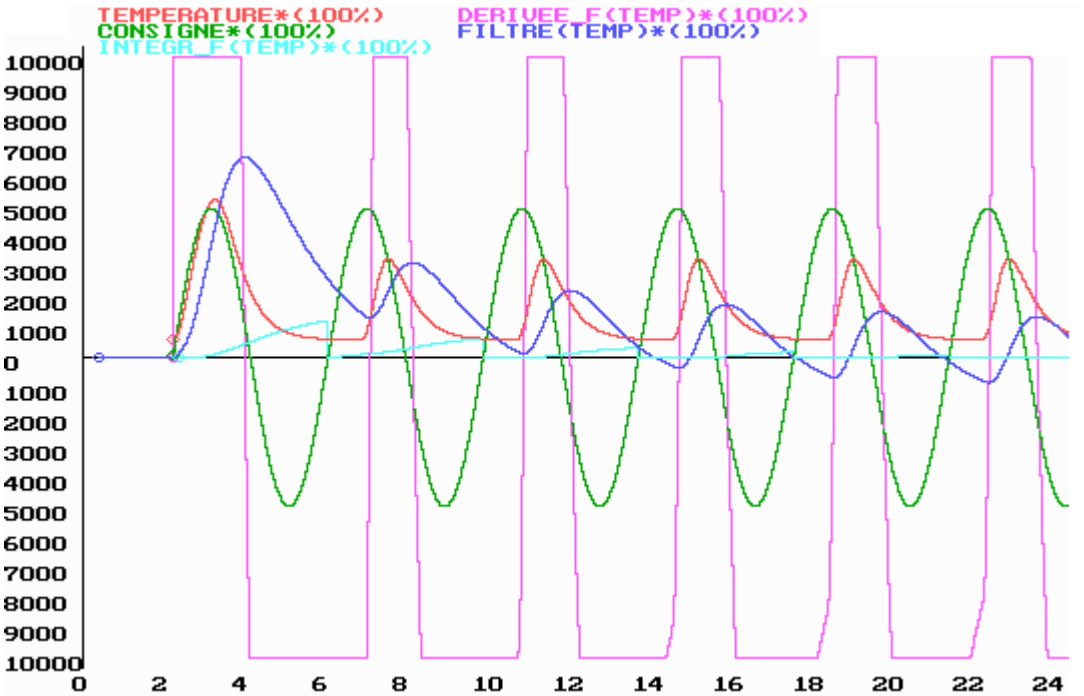


Form of Signal source (Block Wave Pulse)

The functions blocks treated in the application will be initialized before each new sending of pulsation, and the period, the increments of system time will be forced to the value Te (corresponding to the current omega). Note : for the delay block, an iterative shift of the values was introduced, as long as this transmitted fictitious time is not equal to the parameterized value of delay.

A box " Bode Diagram " collects the answer resulting from the process as well as the signal of origin, and tries to identify the gain and phase shift introduced by the sought system. Several operations are made on the signal in order to accelerate the identification :

- a band pass filtering $F(p) = \dfrac{1}{1+\dfrac{T_c}{N}\cdot p} * \dfrac{N.T_c.p}{1+N.T_c.p}$ (N > = 1) in order to eliminate the continuous components and the high spurious frequencies,
- a derivation first in order to detect the maxima of the signal (memorizing of the maximum values and sample numbers for phase shift),
- a derivation second to count oscillations in the case of lead phase on the first pulsation or not filtered continues component,
- an integral of the calculated filtered signal over each period to measure stabilization by comparison between 2 periods.



Example of signal processing
(source=instruction, output=temperature)

Coefficient N of the band pass filter is adjustable by user (an N small lets pass a weak frequency band) and the acceptable error on the integral can be adjusted according to needs (compromised between precision and speed of convergence). In the case of signals without harmonic, this filter can be inhibited by N=100.

When the allowed error on the integral is reached and maxima is detected, the gain and the phase are calculated with withdrawal of the action of the pass-band filter known.

$$Gain_System = 20 * \log_{10}\left[abs\left(\frac{\max_output - \min_output}{\max_source - \min_source} \right) * \frac{1}{K_{corrector}} \right]$$

$$Phase_System = \left(Num_sample(\max_output) - Num_sample(\max_source) \right) * \frac{360}{Max_sample} - \varphi_{corrector}$$

however for the specific pulsation, the filter becomes :

$$F(j\omega_c) = \frac{j.N.T_c.\frac{2.\pi}{T_c}}{\left(1 + j.\frac{T_c}{N}.\frac{2.\pi}{T_c}\right) * \left(1 + j.N.T_c.\frac{2.\pi}{T_c}\right)} = \frac{j.N.2\pi}{\left(1 + j.\frac{2\pi}{N}\right) * \left(1 + j.N.2\pi\right)}$$

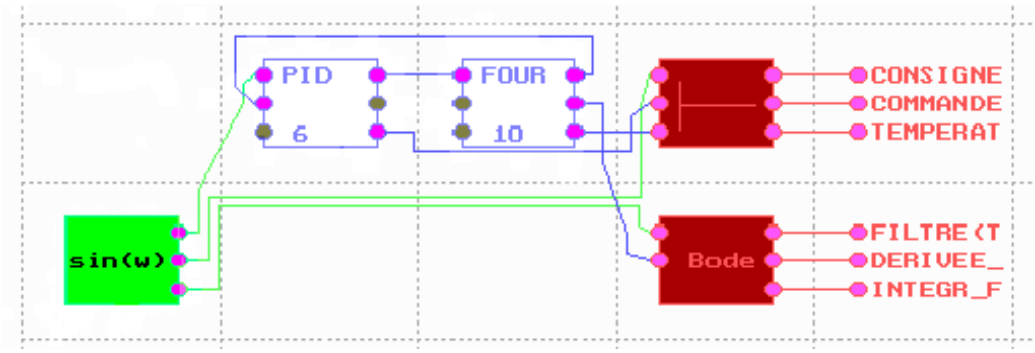
either $\varphi_{Corrector}(\omega_c) = 90 - \arctan\left(\frac{2\pi}{N}\right) - \arctan(2\pi.N)$ $(in\deg ree)$

$$K_{Corrector}(\omega_c) = \frac{2\pi.N}{\sqrt{1 + (2\pi / N)^2} * \sqrt{1 + (2\pi.N)^2}}$$

These obtained values can be traced in the Bode diagram, according to gain and phase scales selected (two different scales and colors).

BODE diagram – How to use

The user must connect a " Wave Pulse " box in entry of the system to be treated. A " Bode " box collects on the one hand this input and on the other hand the output of the process.



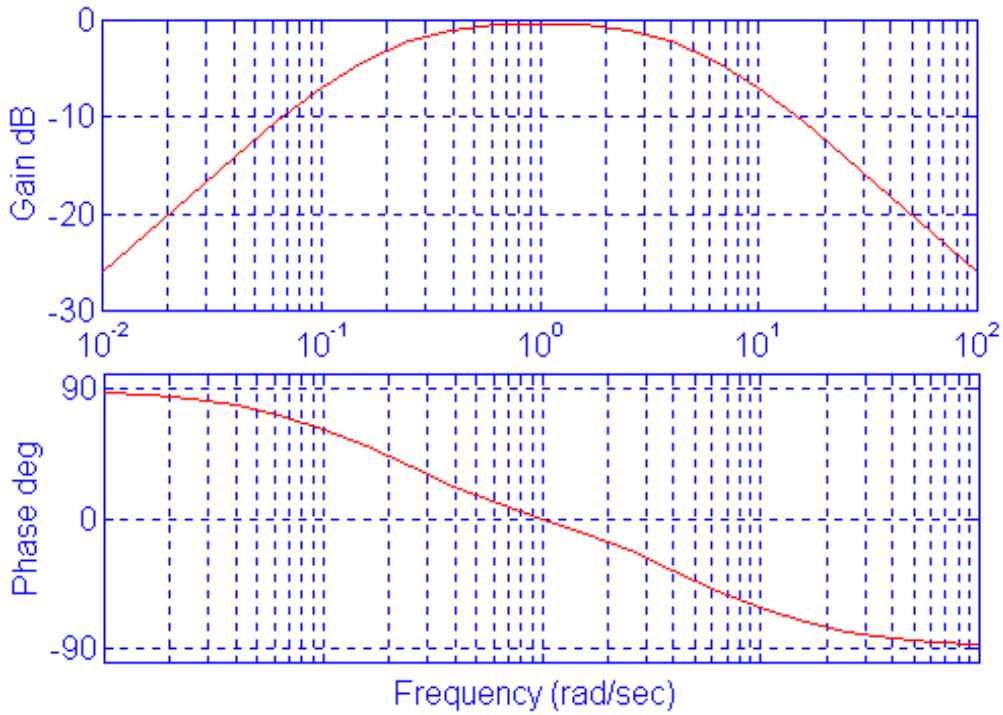
Example of connections of Wave Pulse and Bode boxes

Consequently, the application can be launched and the layout activated by the function " **Zoom In** " on the " Bode " box in **STOP**. *Saturations of the various blocks* can be maintained (real case) or be eliminated (theoretical case).

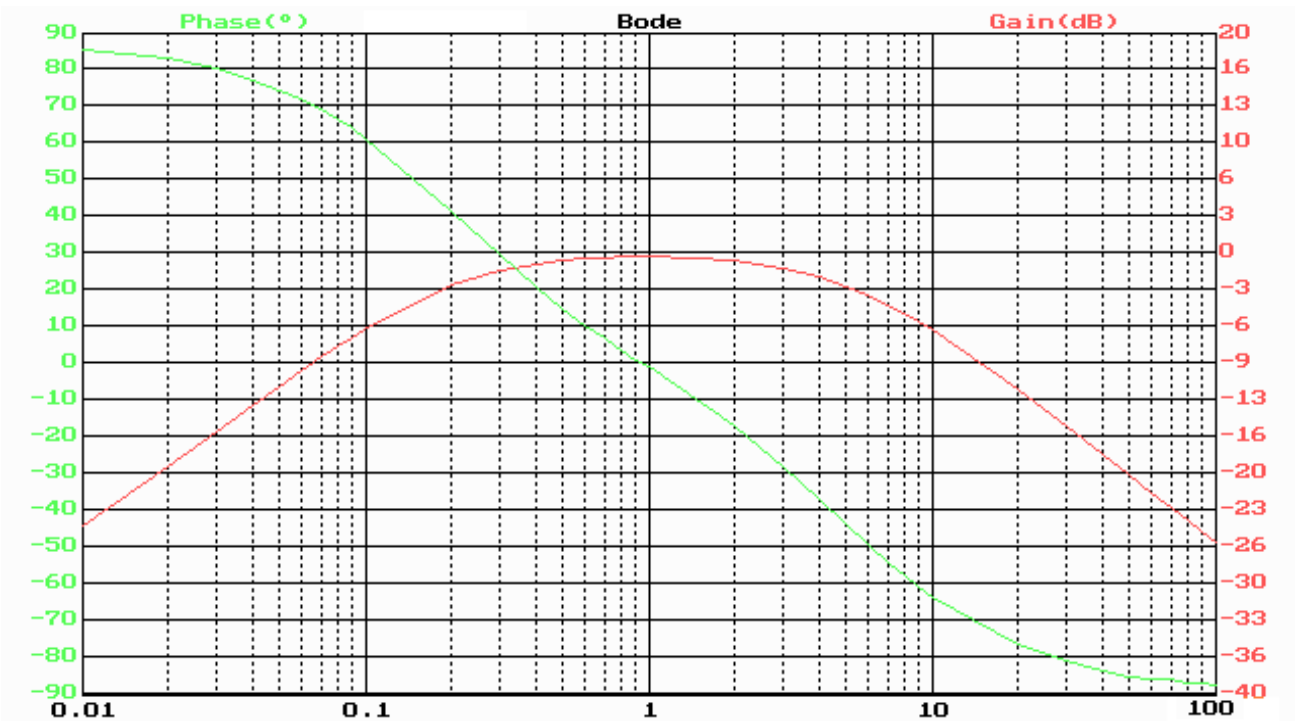
From a methodological point of view, the tests must begin without filtering (N=100), and the speed of convergence towards zero of the integral (indicated on the screen) or output signal forms it recorded will make it possible to adjust this coefficient and the acceptable error. A detailed attention must be paid to the constant sizes of the systems which, if they are preserved, will generate a time of important filtering and will mask the signals too weak source of amplitude. The unipolar processes will present, them also, a continuous component, and their gain will be attenuated of a coefficient 2. Saturations of blocks, although really present, can generate parasitic oscillations and decrease the intrinsic performances of the systems.

The number of samples forming the source, must at least be equal to the desired precision of layout of phase-shift. Thus for 1 degree of precision, 360 samples will be necessary.

For example, the figures below give the Bode diagram of $F(p) = \frac{5p}{(1+0.2p).(1+5p)}$ layout on the one hand with Matlab, and on the other hand with Prosimul (without blocks saturation).



Example of layout of Bode Diagram(Matlab)



Example of layout of Bode Diagram(Prosimul)

EXCHANGES CONFIGURATION

PROSIMUL allows simultaneous exchanges via Binary and analogic parallel connections and via serial connections (for large quantity of exchange of variables).

Cards on USB port are used as parallel cards.

According to the mode of selected exchange, times of exchange with PLC can vary. A system variable records the total number of exchange in serial connection (1 exchange = reading of PLC outputs and writing of PLC inputs). In case of Communication error, the SYS_COMPTEUR_ECHANGE variable is set to -1 and can be tested

Access to the exchanges configuration is made from application navigator “ Configuration->Exchange ”.

Exchanges Configuration – Binary Zone

- Select if necessary the card type (contact us for a specific integration).
    * Card EPC64100 and ACL7225 : with « PORT95NT » driver
    * Card PCI7432 : with « PCI-DASK » driver
    * Card PIO32IO[A] : with « DRIVERLINX » driver
    * Card PISO-P32A32 : with « UNIDAQ » driver
    * Card FESTO EASYPORT on USB port
    * Card SEALEVEL 8227 on USB port

- Indicate the card address into decimal if necessary (EPC64100 cards). For cards drived with DriverLinx, indicate the configuration of cards via " DriverLinx Control Panel ".

- Indicate the number ways (modulo 16) read and written via the cards (according to connections). Inputs cards will be stored in the 1st inputs Prosimul and outputs will result from the 1st outputs Prosimul.

Exchanges Configuration – Analog Zone

- Select if necessary the card type (contact us for a specific integration).

- Indicate the card address into decimal.

- Select its operation mode (unipolar, bipolar, isolated ways...) according to its type.

- Exploitation of the values is done thru **functions blocks** " Input_Ana " and " Output_Ana ".

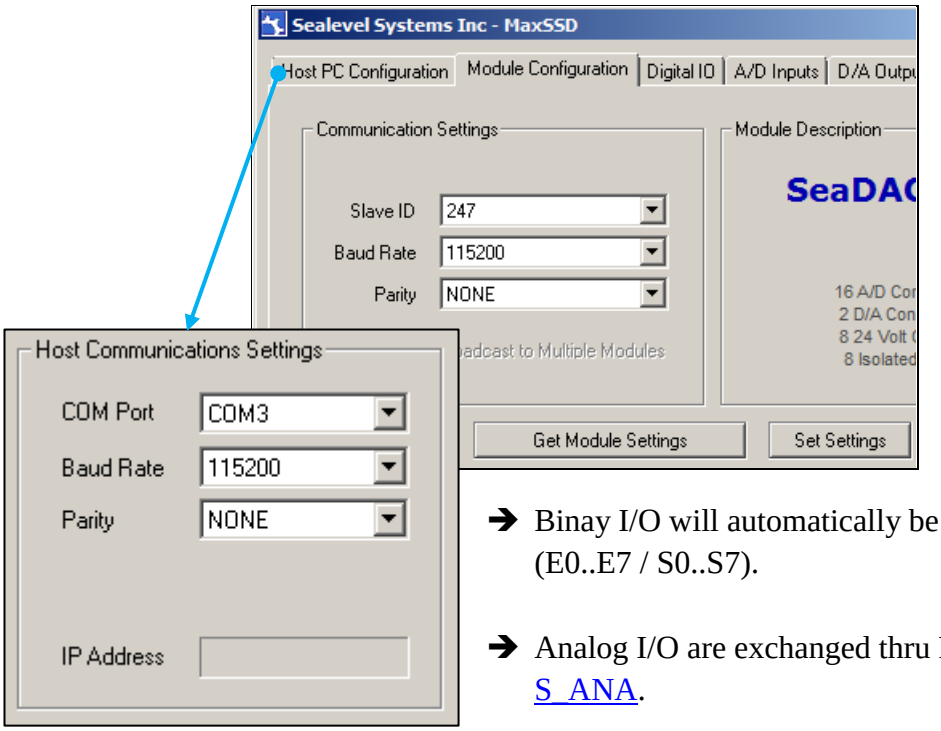
Case of I/O on USB port

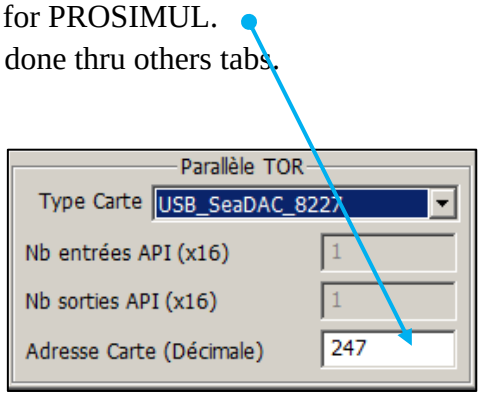
BEFORE TO CONNECT hardware, install available driver of card.

(*) Configure available options with switch (see technical doc) or with configuration software.

(*) CARD FESTO EASYPORT
By default works at 115.200 bauds.
Unipolar mode 0-10V or bipolar mode +-10V is adjusted by switch.
Gain is adjusted by switch.

(*) CARD SEALEVEL 8227 (see page 29 and next of SeaDAC User Manual)
- Move switches that allow to adjust :
    * E1/E2 : analog outputs 1 and 2 signal (0-5V or 0-10V by default)
    * E3 : analog inputs gain : 10 (0,5/1,0V signal) or 1 (5/10V signal – by default)
    * SW3/SW4 : analog inputs in current mode or tension mode (by default)
- Then start MAXSSD.EXE software (installed with driver).
    * Select COMn port that is mapped on USB port
    * For first use card speed is configured at 9.600 bauds. To increase performances, speed must be modified to 115.200 bauds (stay parity to NONE).
    * In tab « Module configuration » use command « Get Module Settings »
    * Modify speed to 115.200 baud et use command « Set Settings ». Afer execution, la vitesse du PC USB port speed must also be modified (tab Host PC Configuration)
    * Slave ID can be modified. It will be use as « Card Address » for PROSIMUL.
    * Read / Write tests of binary or analog inputs / outputs can be done thru others tabs.

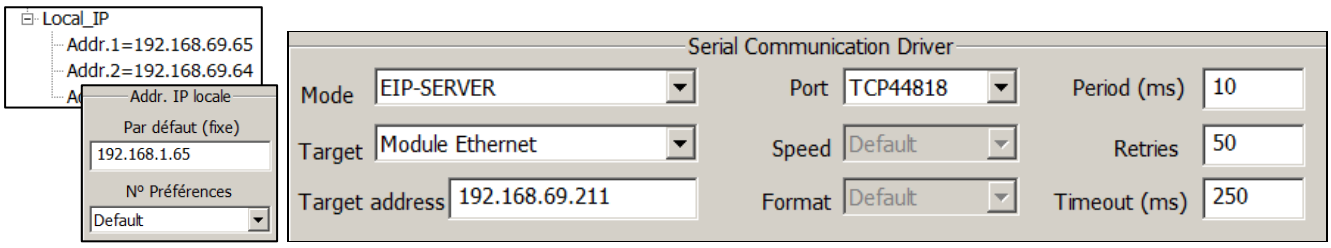




➔ Binay I/O will automatically be exchanged with PROSIMUL I/O (E0..E7 / S0..S7).

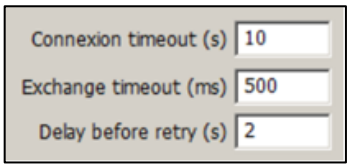
➔ Analog I/O are exchanged thru Function Blocks E_ANA and S_ANA.

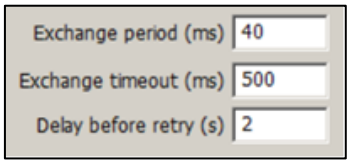
Serial Connection Parameters



- **Local IP n°**: « Défault » or « Addr.1/2/3 ». This parameter allow to select local first physical IP address (Default) or one of IP addresses configured and stored in « Preferences" file. If Default is choosen, a fix IP address stored in .SIM file can be specified.

- **Mode** : selection of serial or Ethernet or specific communication driver.
  . Without (no serial exchange)
  . Modbus : serial RTU or Ethernet TCP/IP client**(C)** or server**(S)**
  . Unitelway : serial
  . DF1-Direct : serial
  . P3964-RK512 : serial
  . Rockwell Driver : specific for RSLinx server
  . OPC Client : specific for OPC Server
  . Applicom Driver : specific for Applicom cards and protocols
  . S7-MPI : specific for S7Online driver on MPI
  . EthernetIP client : client for Rockwell PLC on Ethernet**(C)**
  Datas addressed by file Nx (CIP-PCCC) or by Tags (CIP-TAGS)
  . S7-ISO on TCP : client for Siemens PLC on Ethernet**(C)**
  . EthernetIP server : server for Rockwell "Ethernet Module" **(S)**





- **Target** : selection of distant equipment type depending from selected driver.

- **Target Address** : IP address or number or name of distant equipment.

- **Port** : TCP ports for Ethernet drivers(**C/S**) - COMx/USB with/without RTS/CTS for serial driver.

- **Speed** : for COM1 or COM2 - 110 to 115200 bauds.
- **Format** : for COM1 or COM2 - 8 bits, 1 stop, parity even/odd/without.

- **Exchange period (ms)** : for serial driver or Ethernet in « **client** »(**C**) mode : adjust exchange periodicity (R/W cycle). It is recommended to adjust « Process » period at half of « Exchange » period.

- **Connexion timeout (s)** : for Ethernet in « **serveur** »(**S**) mode : timeout (second) of client connection.

- **Exchange timeout (ms)** : timeout for detection of connection fault.

- **Delay before retry (s)** : for Ethernet drivers : delay to restart driver after connection or exchange timeout.

- **Retries** : for serial drivers : number of exchange trial before driver exit.

- **Switch Bytes** : allow to switch LSB/MSB of binary or numeric values.

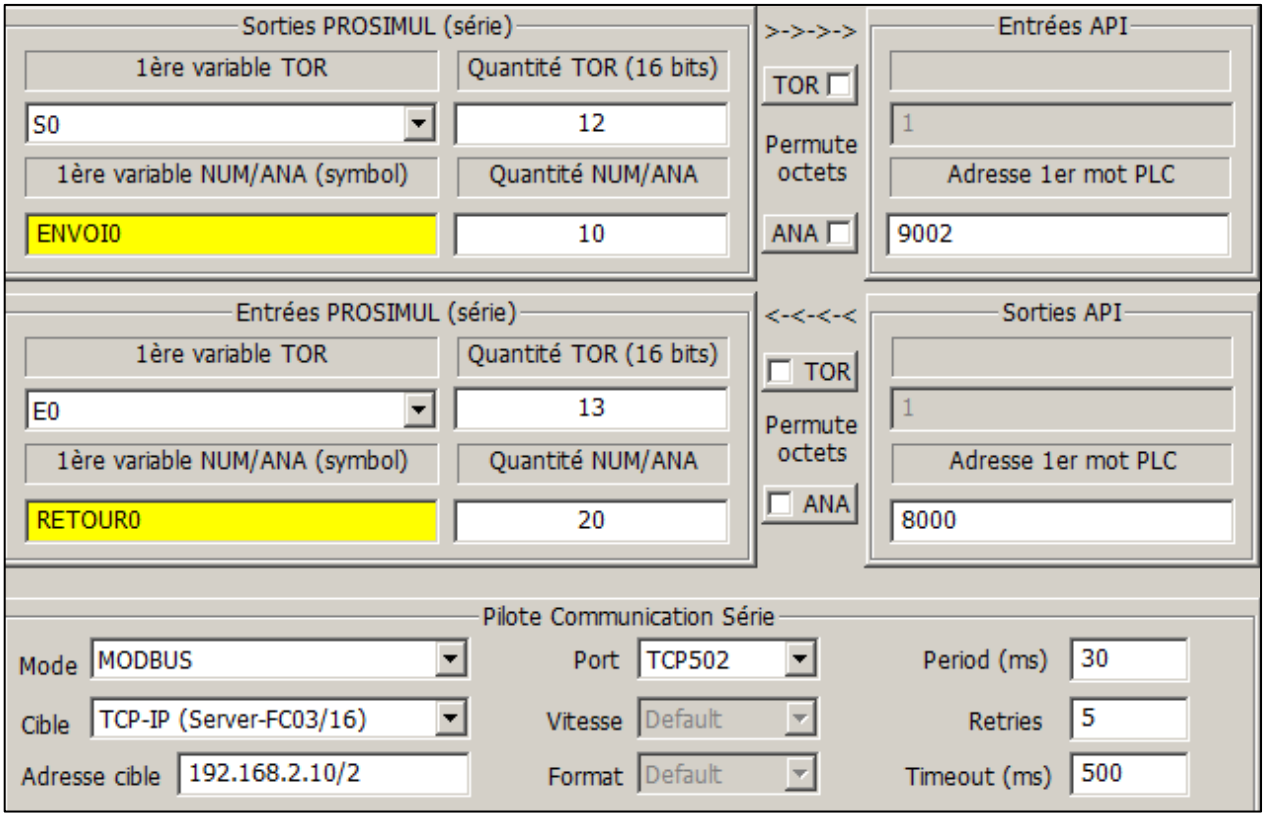
Modbus Protocol

- Serial – RTU (target = slave)
Prosimul is **Master** of the exchanges. Indicate the number of reached slave **S**. Configure exchanges tables with the Modbus addresses of PLC.

Note : for the automat **APRIL series 1000,**: word %IW0 is located at a negative index of -4353 compared to %MW0, %IW1 with -4354, %IW10 with -4355, etc... Use BFC Copy with indexing to charge PLC inputs. Prefer asynchronous exchanges for port Modbus APRIL configuration (faster).

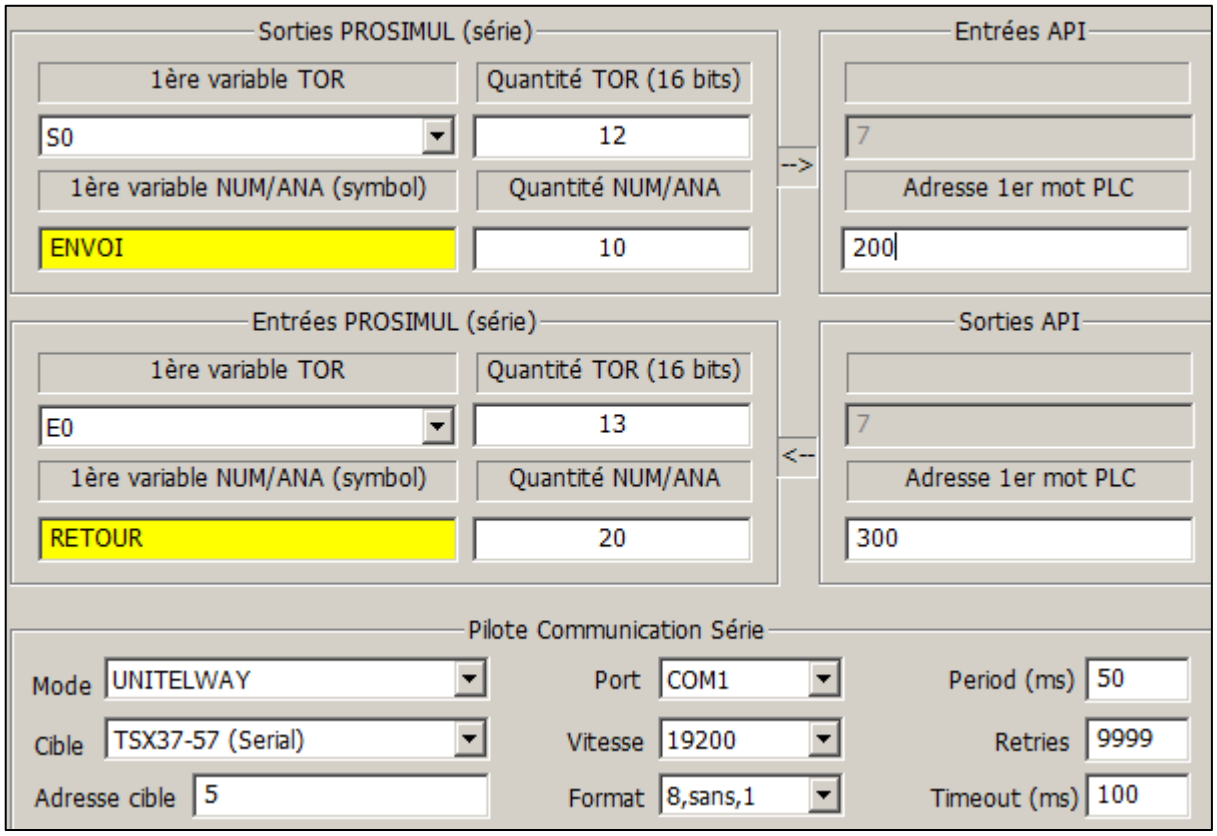
- Ethernet TCP/IP Target=Server (PROSIMUL is Client)
Select the used function code FC03 (Read) + FC16 (Write) or FC23 (Write + Read).
Indicate the IP address of distant equipment. Optionally, slave number can be writed after IP address behind "/" character (ex: 192.168.2.65/**2** means slave 2). By default, address 1 is selected. Configure exchanges tables with the Modbus addresses of PLC. Adjust the timeout for server response in millisecond. IP address, read or written words address and quantity of words must be the same in the server response. Bytes order can be configured (0=Low first , 1=High first).

- Ethernet TCP/IP Target=Client (PROSIMUL is Server)
Select the used function code FC03 (Read) + FC16 (Write) or FC23 (Write + Read).
Indicate the IP address of distant equipment. Configure exchanges tables with the Modbus addresses open for read / write request. Adjust the timeout for client request in millisecond. IP address, read or written words address and quantity of words must be the same in the server response.
The initial connection is expected before 100 x Timeout after launching Run mode.



Unitelway Serial Protocol

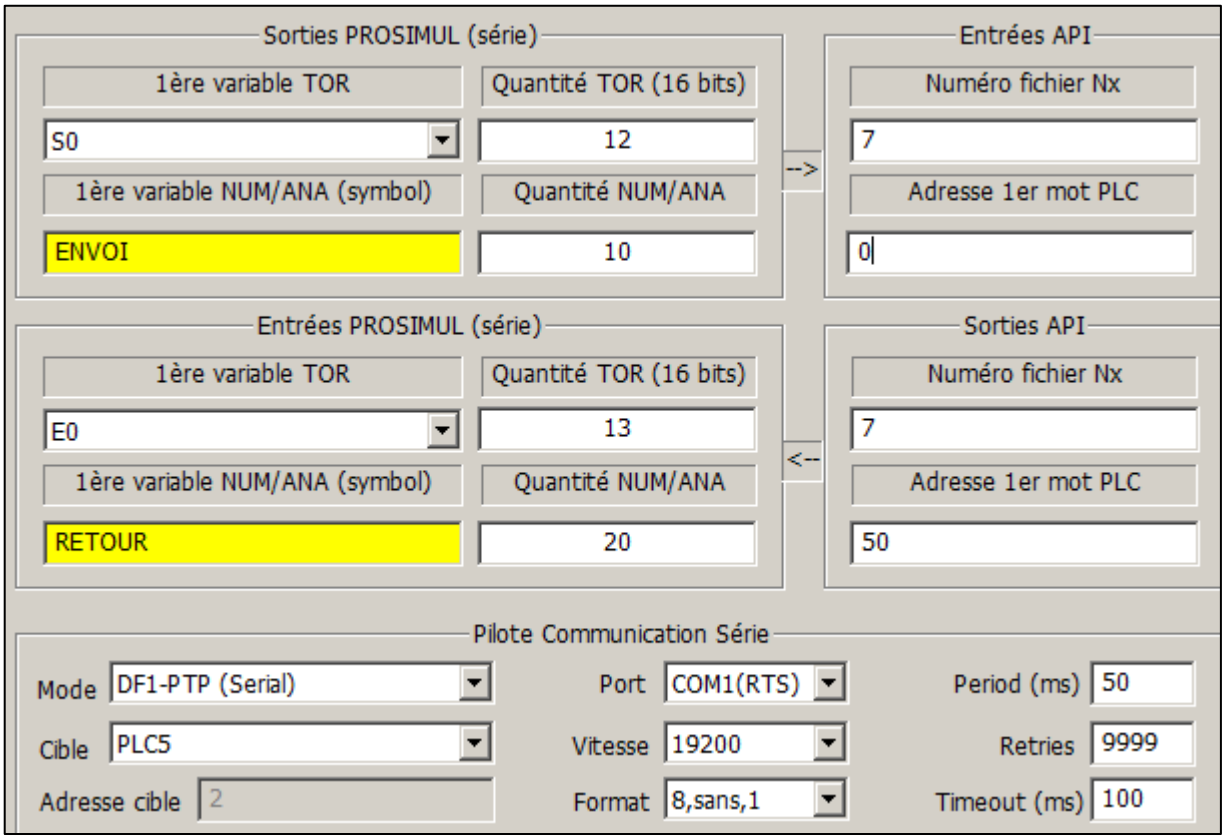
Prosimul is **Master** of the exchanges. Indicate the reached automat type TSX27-47 (Unitelway_1) or TSX37-57 (Unitelway_2). Indicate the number of reached slave **S**. Configure exchanges tables.



Note : to configure automats TSX37-57 in slave (Ad0 =S) it is necessary, in more of the software configuration via PL7-Junior, to use a TSXP ACC1 box.

DF1 Protocol – Serial Point to Point

Indicate the reached automat type (PLC2, PLC3, PLC5, PLC5XX, SLC5XX).
Configure exchanges tables (see automat specificity below).



For **ControlLogix** platforms :

. Produce a " tag " of type of integer table in the desired length (ex : N7 with type INT[50]) via the RSLogix5000 software.

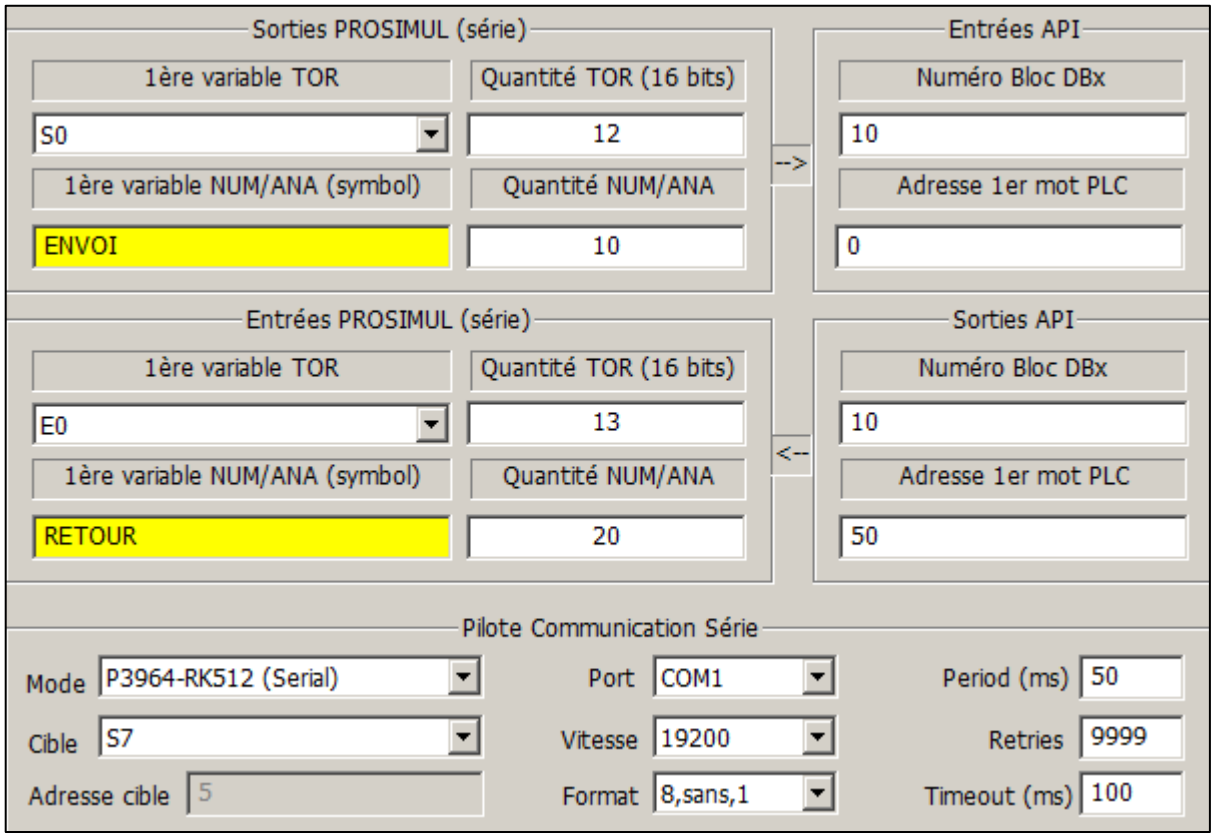
. Configure option " **PLC2/3/5, SLC Mapping** " via menu " Logic->Map PLC/SLC Messages ". Indicate the tag produces in the reserved zones for Mapping. For automats PLC5/SLC the number of reached file must be indicated.

*For automats **SLC5XX** the file of exchanges will be obligatorily N9.*

*Note : Configure serial port side automat with **BCC** error detection and not take the numbers of identical message (message 0 always used).*

P3964-RK512 serial Protocol

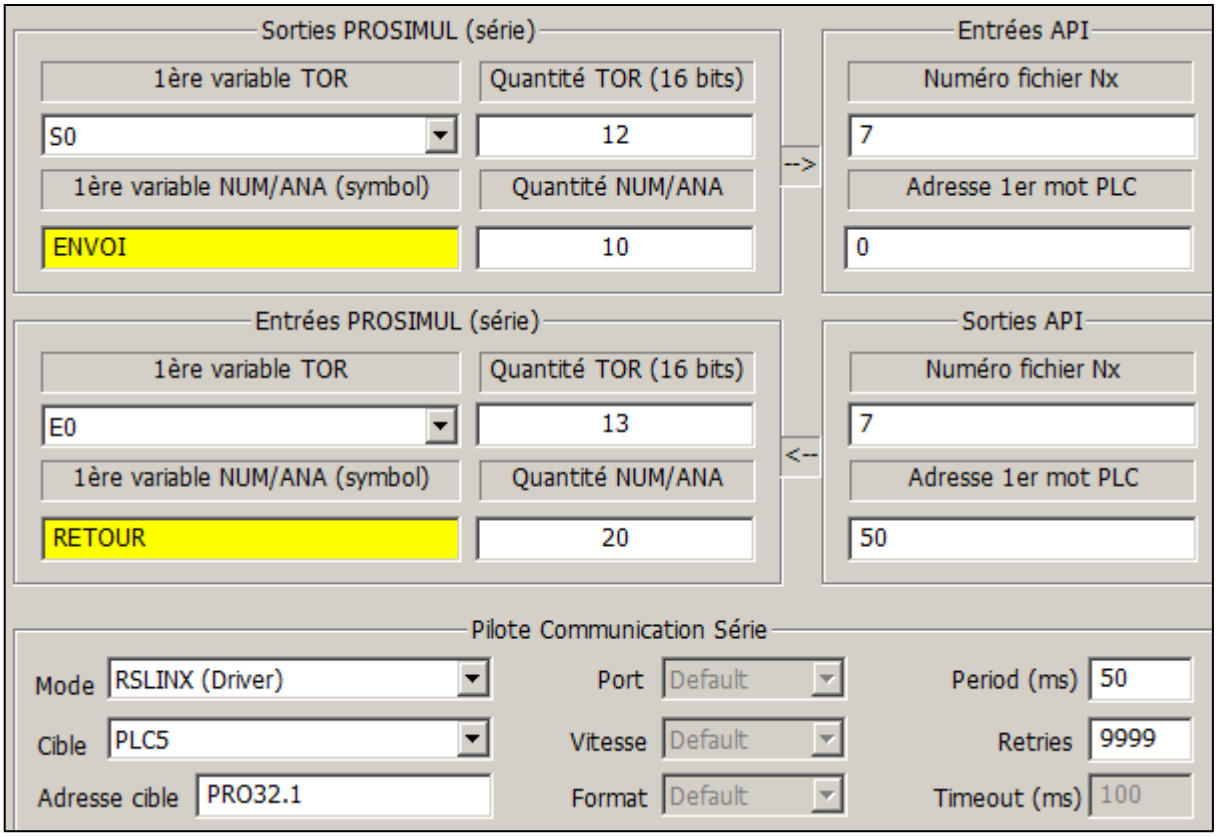
Indicate the reached automat type (Siemens S5 or S7).
Configure exchanges tables (see automat specificity below).



*Note : to simplify the recovery of the words to **S5 PLC** side (in DBi block), the numeric words have their low bytes and high bytes reversed (arrive in the good location of DR and DL of the dB). Binary I/O arrived on the opposite and can be sent or directly resulting from images words EW and AW.*

Rockwell – RSLINX Driver

On RSLINX, configure a Driver (DF1, DH+...) with a **Device Name** as choosed in the field “Target Address”
On PROSIMUL, indicate the reached automat type (PLC2, PLC3, PLC5, PLC5XX, SLC5XX) and target address for this Driver. The target must be designed by a **NAME_X** (where X is the number of the PLC).
Configure exchanges tables.



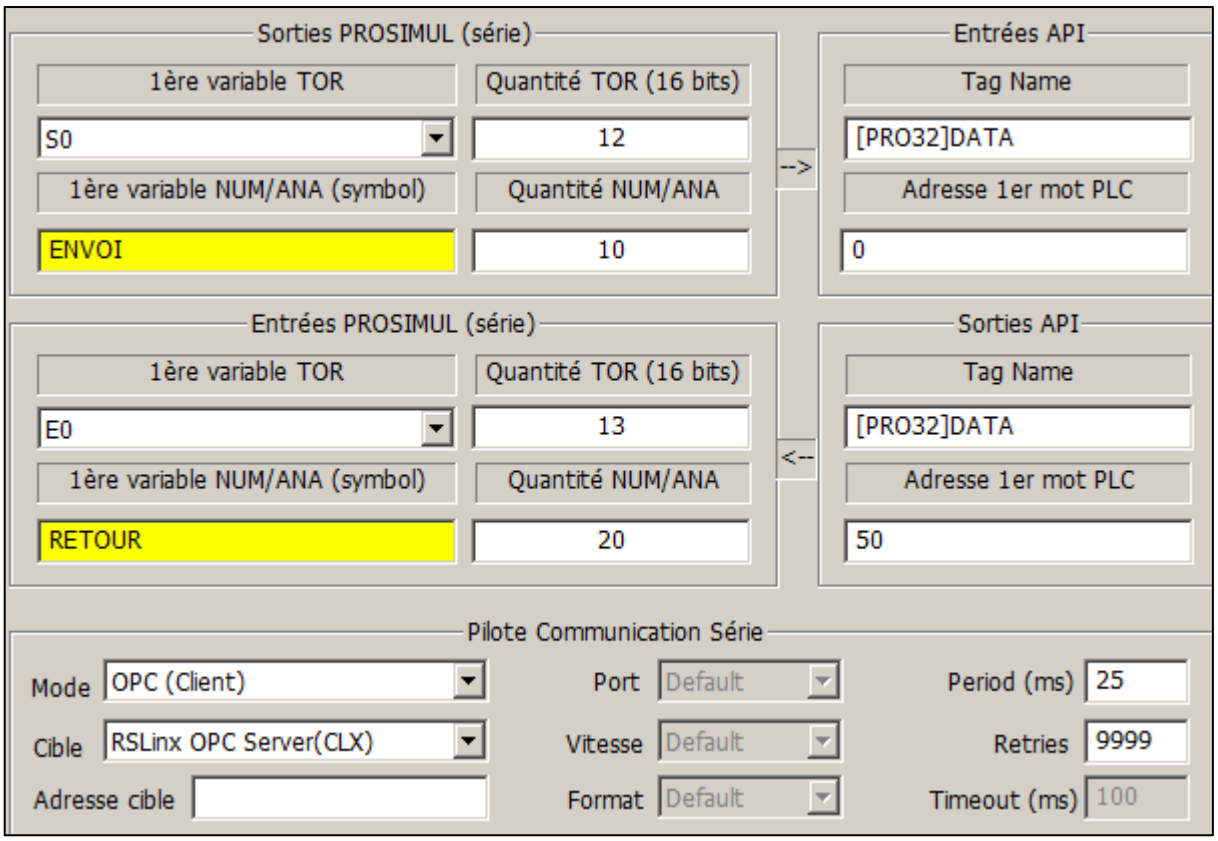
For **ControlLogix** platforms :

. Produce a " tag " of integer table type in the desired length (ex : N7 of the type INT[50]) via the RSLogix5000 software.
. Configure option " **PLC2/3/5, SLC Mapping** " via menu " Logic->Map PLC/SLC Messages ". Indicate the tag produces in the reserved zones for Mapping. For PLC5/SLC automats, reached file number must be indicated.

OPC Client

1 – With PROSIMUL, select the desired server.

- **RSLinx OPC Server** : OPC server for Rockwell SLC/PLC or CLX,
- **Schneider-Aut.OFS** : OPC server for Schneider (Modbus / XWAY) PLC
- **OPC.SimaticNET** : OPC server for Siemens S5/S7 PLC.
- **APPLICOM.OPCServer** : multi-protocols OPC server (Applicom/Woodhead) for many PLCs (Schneider/Siemens/Rockwell/…).
- **BR.OPC.Server_AR** : embedded OPC server for B&R PLC.
- **Kepware.KEPServerEX.V5** : multi-protocols KEPWare OPC server.
- **Matrikon.OPC.Universal.1 / Matrikon.OPC.Siemens.1** : MATRIKON OPC servers.
- **CoDeSys.OPC.02** : OPC server for CODESYS software (PLC emulator).



Optionally you could indicate in the "**Target address**" the IP address of the remote OPC server that you want to use. For local server this parameter must be clear.

2 - Configure selected OPC Server (objective : create a target named **PRO32** for example).

- **RSLINX OPC Server** (Rockwell) : DDE/OPC Menu -> create a " Topic " named **PRO32** and pointing on the selected automat via the network navigator. Adjust " Polled Message " parameter (tab : Data Collection).

- **Schneider-Aut.OFS** (Schneider) : use the " OFS Configuration Tool ". Create one " Alias " named **PRO32** pointing on the target address with selected protocol. Adjust period of min group (tab : Alias) and sampling speed in reception (tab :Comm).

- **OPC.SimaticNET** (Siemens) : with STEP7, create the architecture of your project (network, sub-network, S7 station, gateway so necessary, and only one and single SIMATIC PC Station which will be your Prosimul station). Configure this SIMATIC PC Station which must contain at least an OPC Server with S7 active connection (adjust the period) and a communication card (general IE). Create a S7 connection named **PRO32** between the target station and OPC server station. Locate the .XDB file resulting from the configuration of SIMATIC PC Station (tab : Object Properties Configuration). Use the tool " Station Configurator " of SimaticNET to import XDB configuration file. Check that SIMATIC NET services is launched.

- **APPLICOM.OPCServer** (multi-protocols) : use Console software. Insert the communication card to access the target (ex SW1000ETH for Ethernet link). Configure the needed protocol (Modbus/S7/…) for the selected channel (0 by default). Insert a communication server (Topic) named **PRO32** pointing on the target address. Adjust the max simultaneous requests (Messagerie Parameters) and the refresh period (Topics configuration).

- **BR.OPC.Server_AR** : « to complete »

- **Kepware.KEPServerEX.V5** : create a « Channel » using correct « Device Driver » and adjust its parameters. Create a « Device » named PRO32 for example and adjust its parameters. Create 2 « Tags » (1 for inputs and 1 for outputs) typed 16 bits word fields and adjust « Scan Rate » parameter.

- **Matrikon.OPC.Universal.1 / Matrikon.OPC.Siemens.1** : create an « Object » with the correct « Device Driver » (ex : MODBUS) et régler ses paramètres. Create a « Target object » named PRO32 for example and adjust its parameters (IP address, port, timeout, …). Create 2 « Alias » (1 for Inputs and 1 for Outputs) as array of 16 bits integer and adjust the « Update Rate » in milliseconds.
Ex : Alias « Read » « MODBUS.PRO32.**4** :301 :10 » (10 words starting at address 300 (301-1) in PLC Modbus R/W Registers (value 4)).

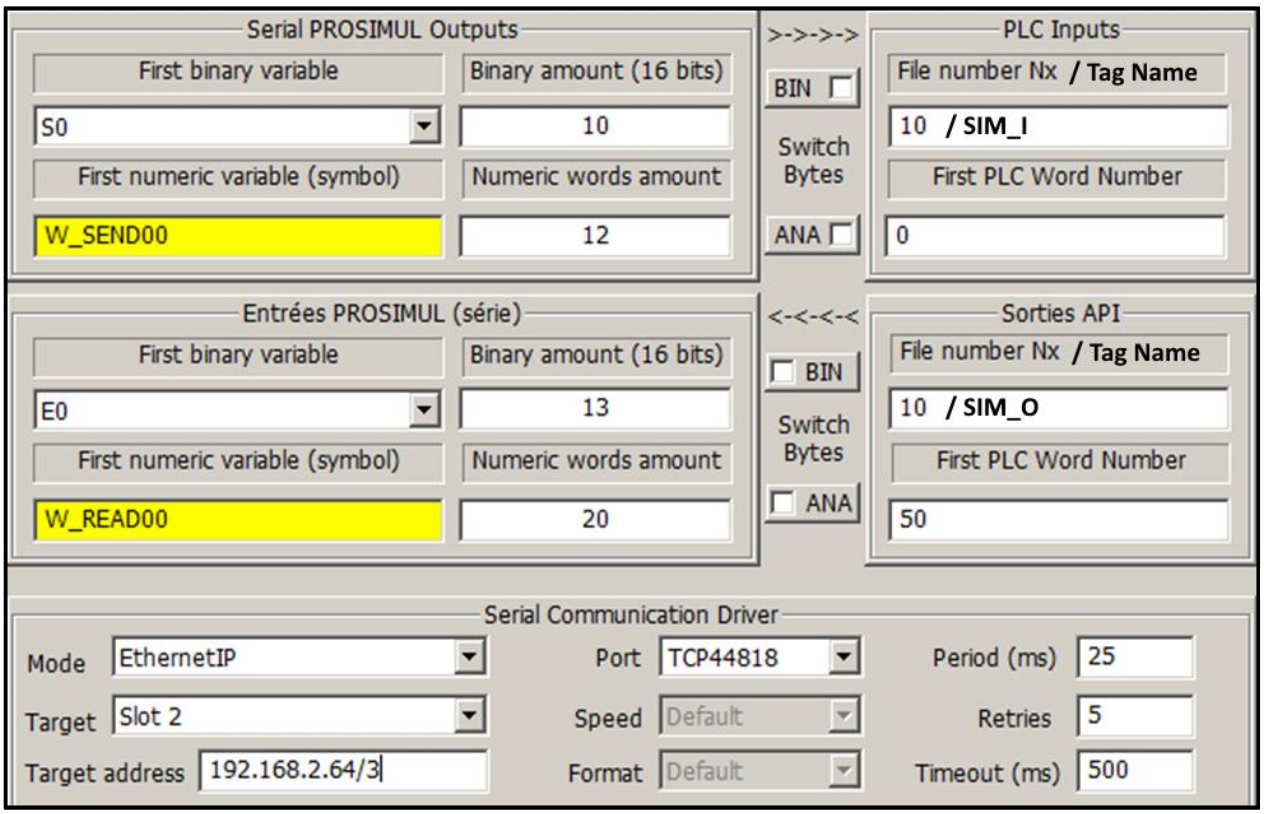
- **CoDeSys.OPC.02** :
Start « PLCWinNT » program (delivery with Codesys).
With CODESYS software : configure target on « PLCWinNT ». Create global variable typed ARRAY [1..N] OF INT. Configure symbol file (options) with R/W rights access. Adjust communication on « Tcp/Ip Level 2 Route » driver (gateway=local). Do “Online”, « System access » then « Start ».
Launch « Codesys OPC Configurator », adjust Update Rate server and choose communication driver like Codesys communication.

3 – With PROSIMUL configure exchanges tables. In the exchanges table (fields DATA NAME) the target data must be given by name (see point 2 for explanation about “PRO32” symbol):

- for RSLinx OPC Server : « **[PRO32]VarName** » for example (VarName is the PLC Data Name read or write)
- for Schneider-Aut.OFS : « **PRO32_!%MW** » for example
- for OPC.SimaticNET : « **[PRO32]DB10.DBW** » for example
- for APPLICOM OPCServer : « **PRO32.VarName** » (VarName is the PLC Data Name read or write).
- for BR.OPC.Server_AR : « **to be completed** »
- for Kepware.KEPServerEX.V5 : « **ChannelName.PRO32.TagName** »
- for Matrikon.OPC.Universal.1 and Matrikon.OPC.Siemens.1 : « **Alias** »
- for CoDeSys.OPC.02 : « **PLCx :.VarGlobaleName** ». PROSIMUL table length must be the same as OPC array of int.

EthernetIP client driver (CIP-PCCC / CIP-TAGS)

Select target (CPU position) :
- « Direct UC » for Ethernet **for ENI modules** or integrated port on **SLC/PLC** (only for **CIP-PCCC**).
- « Slot 0..19 » for **CLX port or deported Module**. CPU must be in slot 0 to 19.



Enter the IP address of target on Ethernet. Optionally, after IP address, enter behind a "/" character the "Channel" number of connection (ex: 192.168.2.66/**3** for channel 3). By default channel 1 is selected. For subnetwork you can configure static route on the computer (route add).

Configure exchanges tables. Total of variable (binary + numeric) must be less than 250.

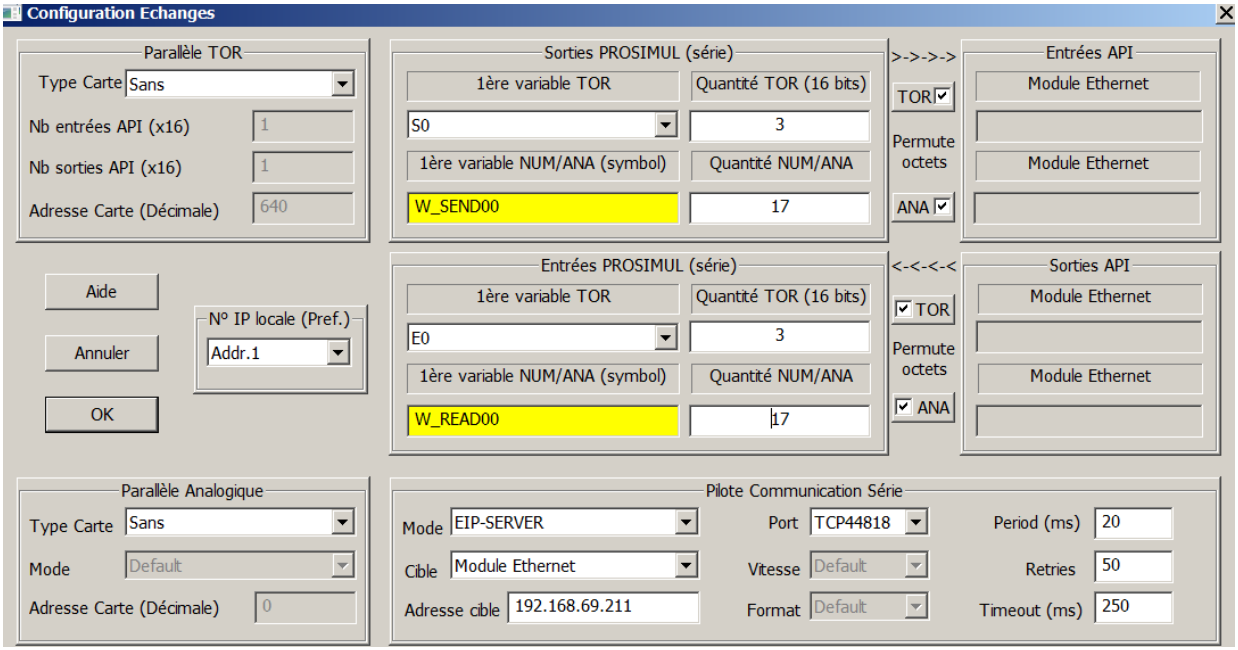
CIP-PCCC : only Nx files are accessible.
CIP-TAGS : only Tags are accessible.

For **CLX** platforms with **CIP-PCCC** :

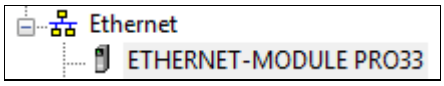
. Produce a " tag " of integer table type in the desired length (ex : N7 of the type INT[50]) via the RSLogix5000 software.
. Configure option " **PLC2/3/5, SLC Mapping** " via menu " Logic->Map PLC/SLC Messages ". Indicate the tag produces in the reserved zones for Mapping. For PLC5/SLC automats, reached file number must be indicated.

EthernetIP server driver

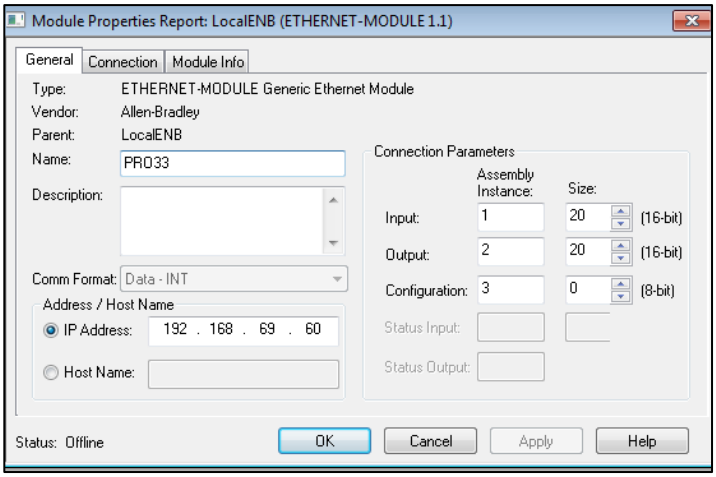
This driver allow to answer to configured **CLX Ethernet Module**.

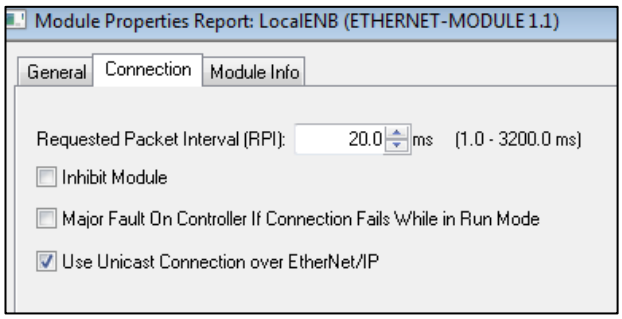


- **In PROSIMUL**, fill in IP address of PLC client and TCP port used (default 44818). Optionnaly, fill in behind IP address (separated by « / » character) the UDP port used (default 2222).
  If few connections are necessary with **few PROSIMUL applications** on the same computer, few IP addresses must be configured on Ethernet card. Those local addresses must be given as parameters in Preferences file and selected in exchange configuration.
  Configure also exchanges tables. **Total of variable (binary + numeric) must be less than 250.**
  It is desirable that application will be configured with a priority level 6 (realtime).



- **In PCL program,** configure the Ethernet Module that will sequence communications. Fill in IP address of PROSIMUL application and configure sizes of I/O datas **(16 bits)** identically as PROSIMUL configuration (ex : 17+3 = 20).





In the tab « Connection » verify that « **Use Unicast Connection** » checkbox is checked and fill in I/O Request Packet Interval.

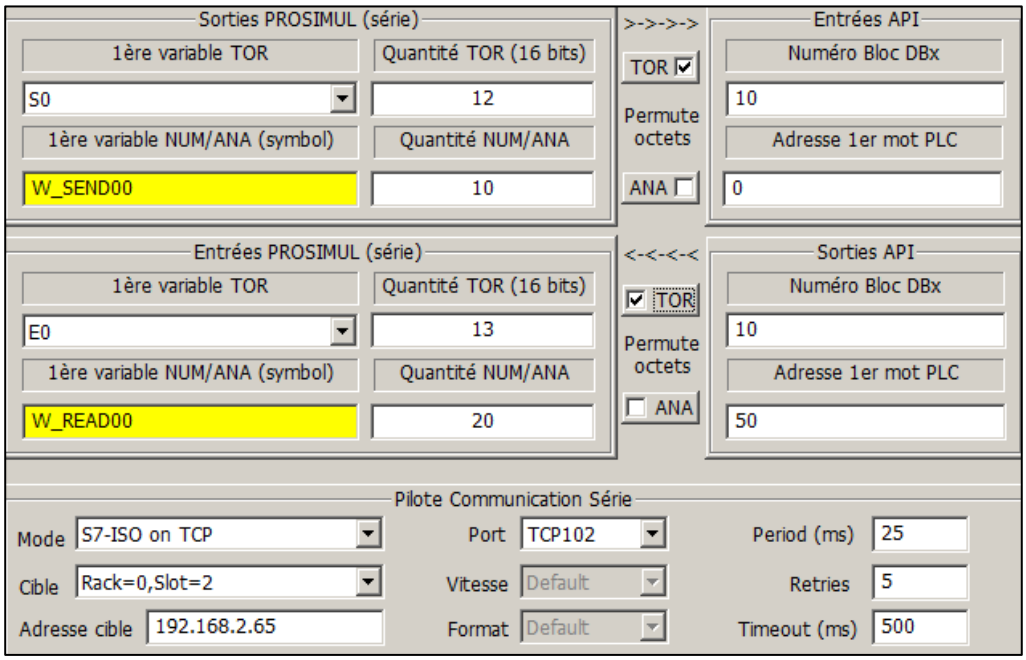
S7-ISO on TCP driver

Select the target CPU slot number.

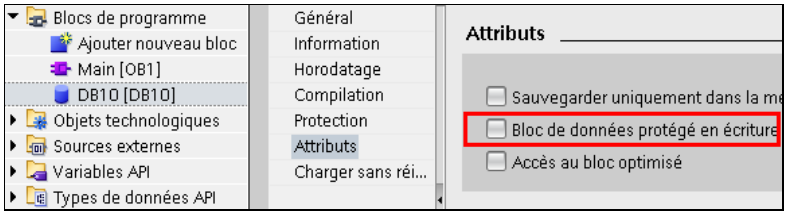
Select IP address of Ethernet way. If necessary configure static route (route add).

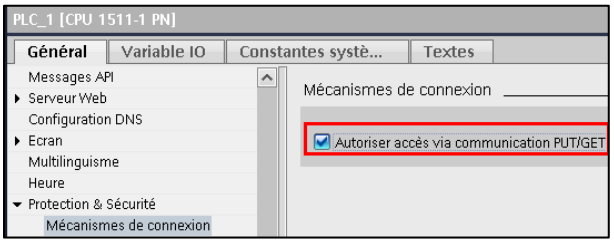
Configure exchanges tables.

Only **DBx** blocks are accessible.



For **1200-1500 series** "Optimised access" attribute mustn't be selected.
And option « GET/PUT acces authorization » must be selected in "Protection & Security "/ "Connection mechanism" configuration of CPU.





How to use PLCSim emulator

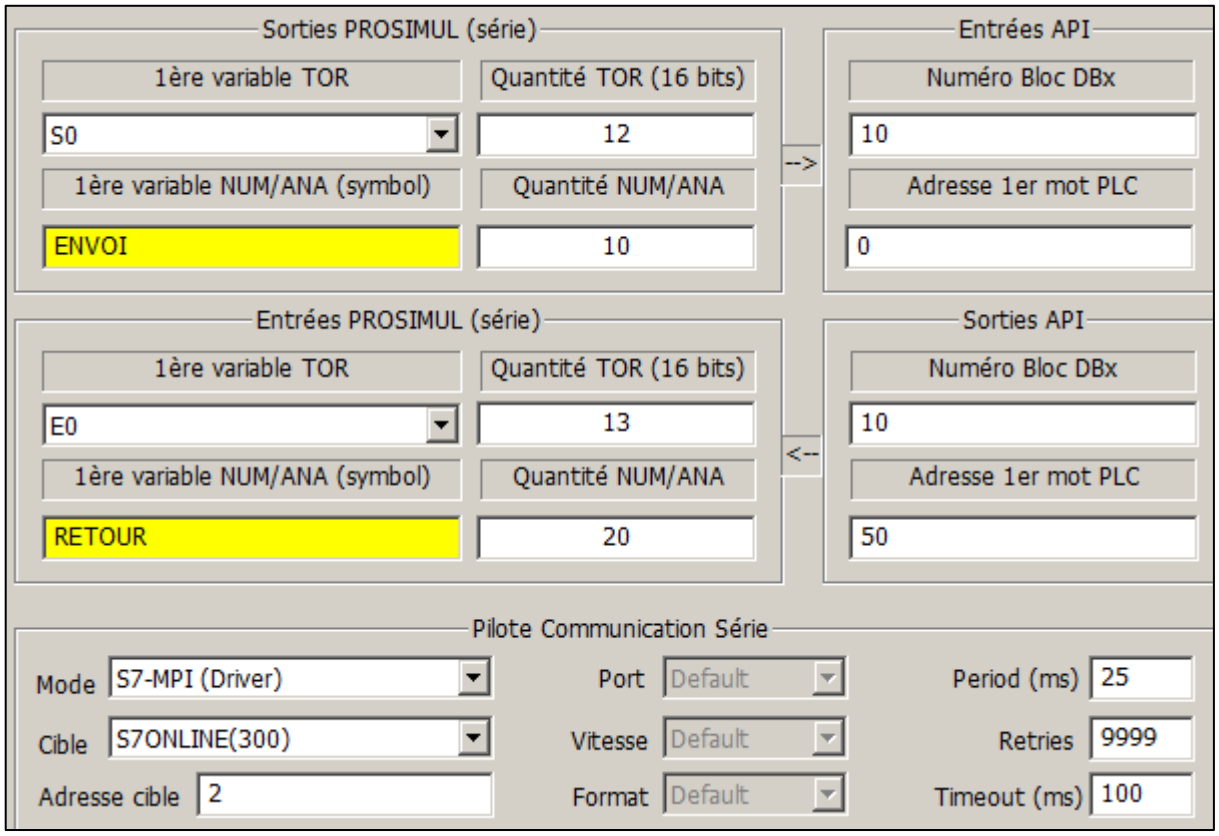
PROSIMUL can be used in connection with PLCSim emulator.
1 – Before starting any SIEMENS software, start NetToPLCSim V1.2.4.
2 – Start STEP7 or TIA-PORTAL and load your application in PLCSim (driver Ethernet).
3 – Configure or load configuration for PLCSim in NetToPLCSim
Ex : PC :192.168.2.212 → PLCSim :192.168.2.215 / Port 102
4 – Start NetToPLCSim server.
5 – In PROSIMUL, configure driver S7-ISO on TCP with IP address configured in NetToPLCSim (ex: 192.168.2.212).

Applicom Driver

The exchanges will be carried out with the Applicom DATABASE (physical card or Direct-Link software). In addition, cyclic exchanges must be programmed with PCCYCL tool. The exchanges runs by PCINIT command.
Configure exchanges tables by indicating the acceded card number and the DATABASE words addresses.

S7-MPI Driver (thru PG/PC interface)

The exchanges will be carried out via configured **S7ONLINE** connection by the " interfaces PG/PC Parameter " of the Control panel of STEP7.
Select PLC model : S7-200 (only DB1 is available) or S7-300/400.
Indicate the reached automat number on MPI or PROFIBUS.
Configure exchanges tables.

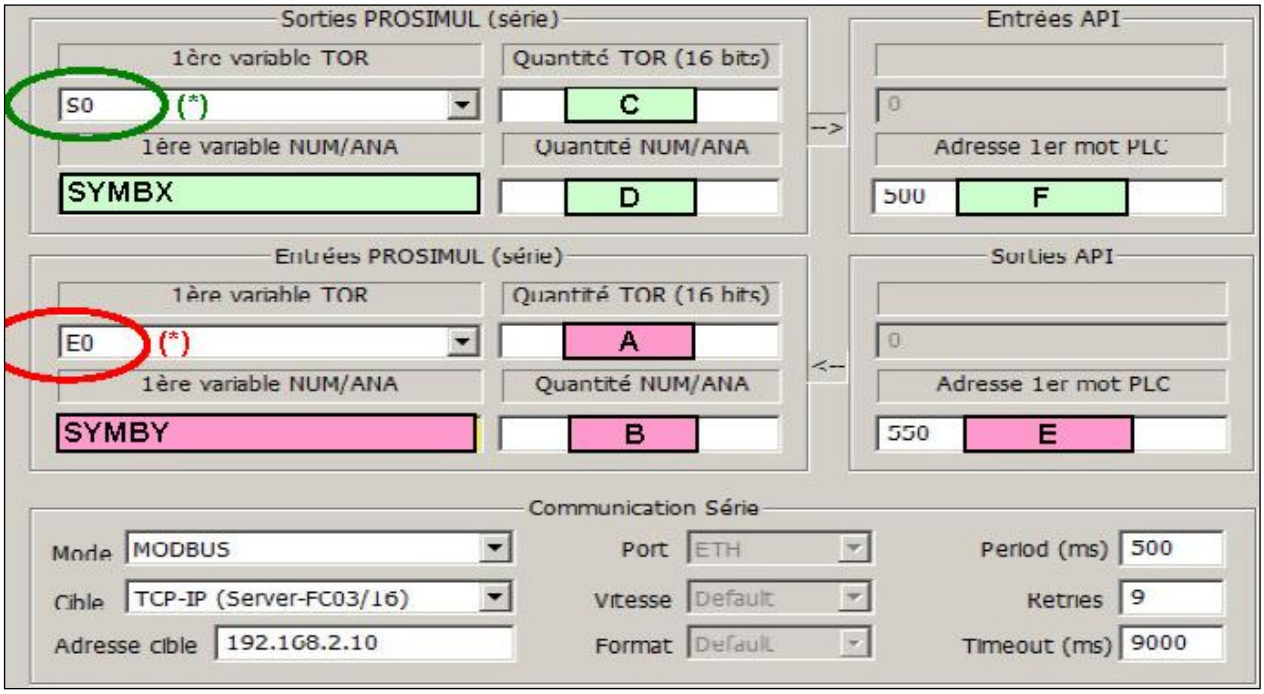


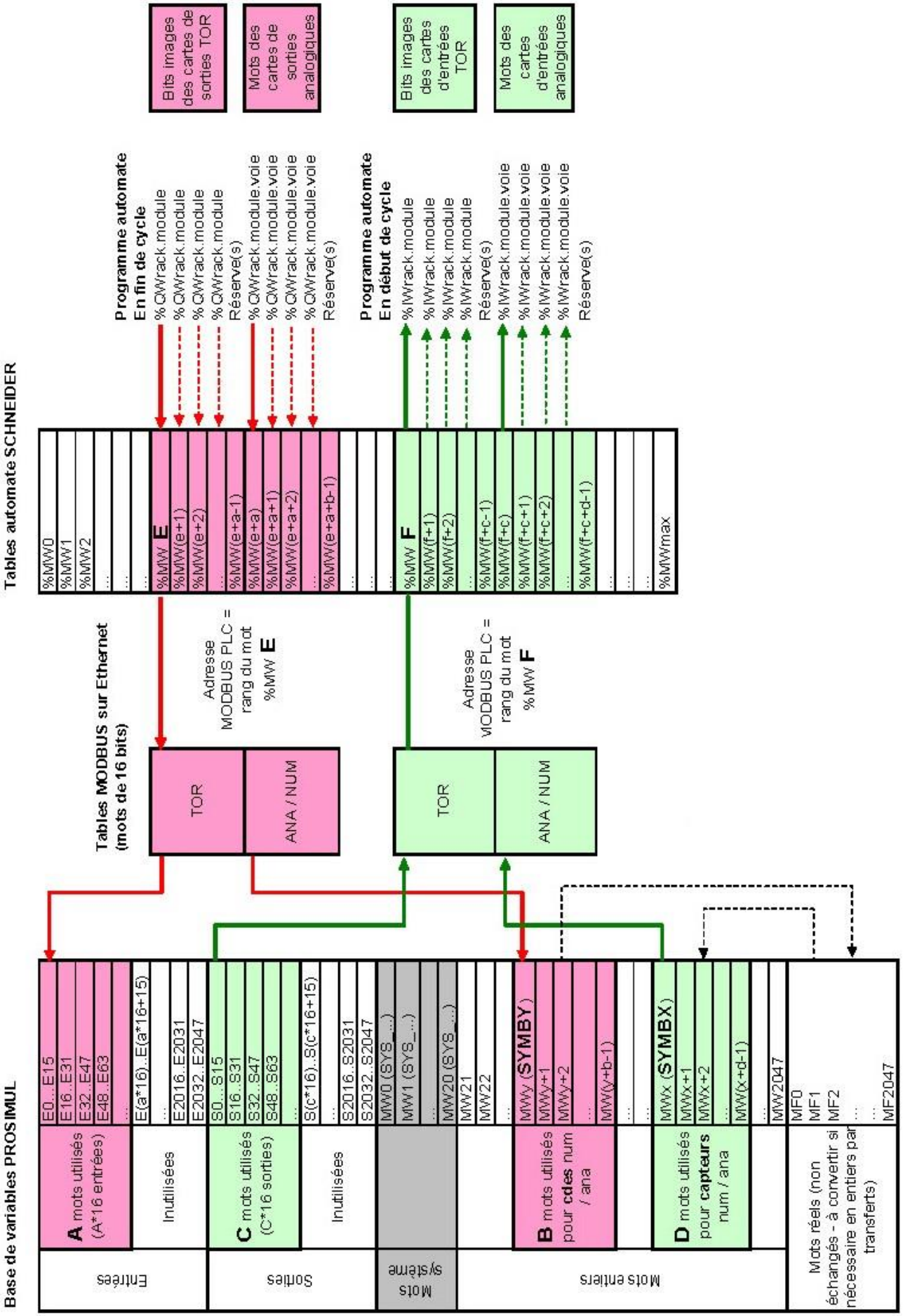
Exchanges tables applications

Exchanges will be carried out via exchanges tables having to be configured.
PLC outputs will be read in the PLC thru a single table including first binary variables and then numeric words.
PLC inputs will be written in the PLC thru a a single table including first binary variables and then numeric words.
See graphic next page.

Tables Configuration

Parameter	**Inputs PLC Zone**	**Outputs PLC Zone**
Number of BINARY words	Quantity **C**	Quantity **A**
Number of ANA words	Quantity **D**	Quantity **B**
ANA Table Prosimul	Associated symbol with the word **Y**	Associated symbol with word **X**
Data block number DBx	For SIEMENS	For SIEMENS
File number for Rockwell Nx	For ROCKWELL	For ROCKWELL
Card number	For Applicom	For Applicom
Data Name	For OPC client	For OPC client
PLC word number	Address **F**	Address **E**




Bits images des cartes de sorties TOR
Mots des cartes de sorties analogiques
Bits images des cartes d'entrées TOR
Mots des cartes d'entrées analogiques
Tables automate SCHNEIDER
Programme automate En fin de cycle
%QWrack.module
%QWrack.module.voie
Réserve(s)
Programme automate En début de cycle
%IWrack.module
%IWrack.module.voie
%MW0
%MW1
%MW2
%MW E
%MW(e+1)
%MW(e+2)
%MW(e+a-1)
%MW(e+a)
%MW(e+a+1)
%MW(e+a+2)
%MW(e+a+b-1)
%MW F
%MW(f+1)
%MW(f+2)
%MW(f+c-1)
%MW(f+c)
%MW(f+c+1)
%MW(f+c+2)
%MW(f+c+d-1)
%MWmax
Tables MODBUS sur Ethernet (mots de 16 bits)
Adresse MODBUS PLC = rang du mot %MW E
Adresse MODBUS PLC = rang du mot %MW F
TOR
ANA / NUM
Base de variables PROSIMUL
Entrées
Sorties
Mots système
Mots entiers
A mots utilisés (A*16 entrées)
Inutilisées
C mots utilisés (C*16 sorties)
B mots utilisés pour cdes num / ana
D mots utilisés pour capteurs num / ana
Mots réels (non échangés - à convertir si nécessaire en entiers par transferts)
E0..E15
E16..E31
E32..E47
E48..E63
E(a*16)..E(a*16+15)
E2016..E2031
E2032..E2047
S0..S15
S16..S31
S32..S47
S48..S63
S(c*16)..S(c*16+15)
S2016..S2031
S2032..S2047
MW0 (SYS_...)
MW1 (SYS_...)
MW20 (SYS_...)
MW21
MW22
MWy (SYMBY)
MWy+1
MWy+2
MW(y+b-1)
MWx (SYMBX)
MWx+1
MWx+2
MW(x+d-1)
MW2047
MF0
MF1
MF2
MF2047

RUNTIME

The launching of the application is done via " **Run** " command of the principal menu.
Configuration options of exchanges and application must be adjusted as a preliminary.

On **RUN command** SYS_INIT_APPLI variable becomes true during first cycle.
On **STOP order** SYS_CYCLE_PROSIMUL variable take value **-1** during last cycle. It allows variables backup for example (see WRITEINI).

Binary actions are available in RUN mode from each screen thru function keys associated with binary variables. Specific force and adjust functions are available from Graphic display, Variables list, Ladder, Litteral, Grafcet and Fonctions display.

Available menu commands

Appli Init ([ALT][CTRL][F05]) : for application intialization.
Chart Init : for Charts initialization.
Time Init : for process and communication times max reset.

Force -> Reset ([ALT][CTRL][F09]) : cancel all active forces.
Force -> List : access to forced variables list

Dictionary : allow to switch between languages present in the dictionary..

Priority : allow to dynamically modify priority of PROSIMUL Windows Threads.

Run / Stop : outing Run mode is possible by pressing [**ESC**] key twice (<1s).
In Developer Mode, starting with [SHIFT] key cause to hide Explorer.

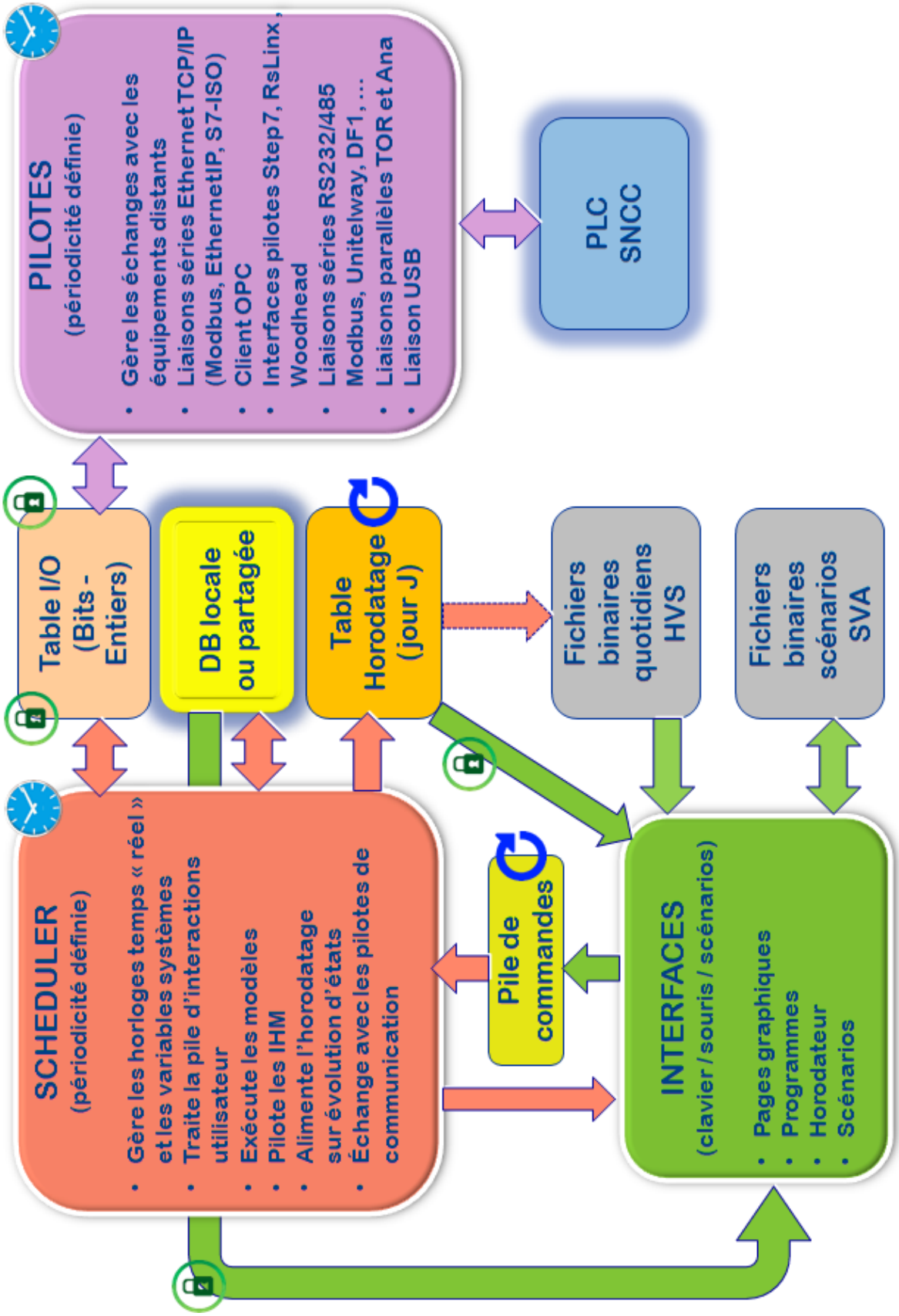
Window : allow to switch easily between opened PROSIMUL applications (like Master/Slaves applications). **Screen select** move main window to next screen if available.

Lock / UnLock : to lock or unlock action on application. If protection is configured password must be given.

Status bar : some states are displayed each second on the status bar.
RUN state timer : 00:00:40 [F15] Uc[13.6%]
+ Number of forced variables : **F[Fxx]** → if superior to 0 background becomes yellow.
+ Processor usage in percent.

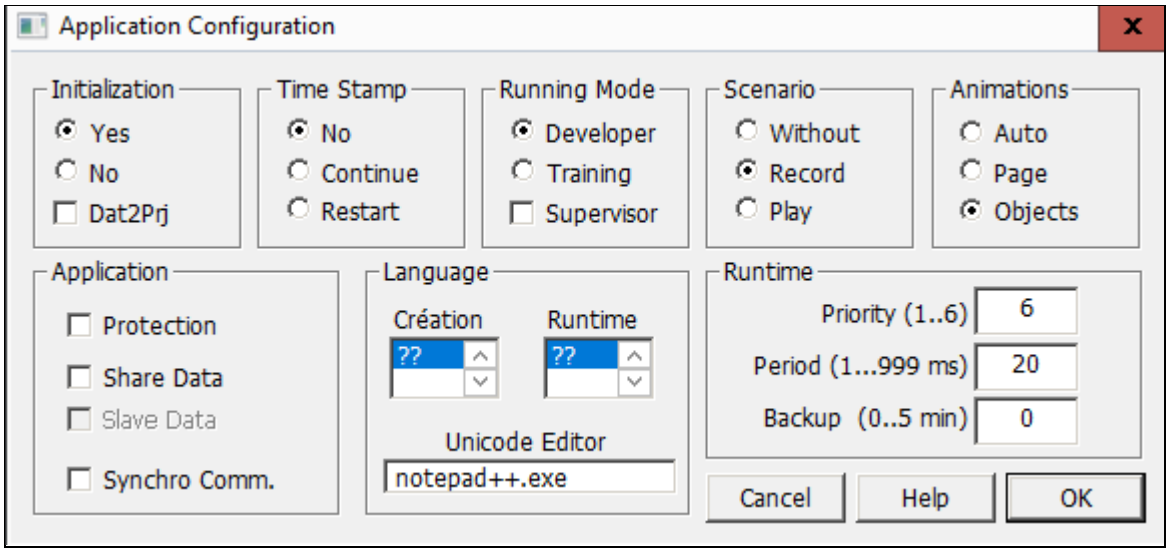
Current and max program and communication periods : P[250<251ms] C[240<260ms]
(in case of communication error, background becomes red). « Time Init. » command reset max times.

Visudyn (submenus) : access to monitored variable list or to monitoring if active (accessible also thru menu « Tools » « Monitoring » or thru right click on blank zone of Graphic page).


PILOTES
(périodicité définie)
• Gère les échanges avec les équipements distants
• Liaisons séries EthernetTCP/IP (Modbus, EthernetIP, S7-ISO)
• Client OPC
• Interfaces pilotes Step7, RsLinx, Woodhead
• Liaisons séries RS232/485 Modbus, Unitelway, DF1, ...
• Liaisons parallèles TOR et Ana
• Liaison USB
PLC
SNCC
Table I/O
(Bits - Entiers)
DB locale
ou partagée
Table
Horodatage
(jour J)
Fichiers
binaires
quotidiens
HVS
Fichiers
binaires
scénarios
SVA
SCHEDULER
(périodicité définie)
• Gère les horloges temps « réel » et les variables systèmes
• Traite la pile d'interactions utilisateur
• Exécute les modèles
• Pilote les IHM
• Alimente l'horodatage sur évolution d'états
• Échange avec les pilotes de communication
Pile de
commandes
INTERFACES
(clavier / souris / scénarios)
• Pages graphiques
• Programmes
• Horodateur
• Scénarios

Application Configuration

Access to the application configuration is made from application navigator " Configuration →Application ".



- **Initialization** : possibility of initializing the variables with their initial values or launching the application with the current values (tests resumption). "**Dat2Prj**" check box allows to transfer values from current directory APPLINAME.DAT file to internal (B/W/R) values of project (when opening SIM file and if variables have same symbol and aren't forced).

- **Monitoring** : possibility of validating or not the recording on change of state of the configured variables via monitoring menu . In positive, APPLINAME_date.HVS file is created each day and could be exploited. The daily file can be written continuously or be restarted.

- **Scenario** : possibility of recording or replaying the scenarios. Those will contain all manual actions made on the application (adjustments/forcings/...).

- **Running Mode** : it is possible to choose between two modes : Developer or Training.
TRAINING MODE: graphics objects with « Dev.only » parameter selected are not displayed. The Explorer isn't visualized too and the graphics pages can be accessed by the [PgUp] et [PgDn] keys or the system variable SYS_PAGE_PROSIMUL. When opening .SIM application by double clicking, the application start automatically in RUN mode.
DEVELOPER MODE: starting with [SHIFT] key cause also to hide Explorer.
Supervisor mode : avoid to refresh background picture (for faster time cycle). Graphic menu isn't displayed. Superimposed objects mustn't move on those pictures.
[PgUp] et [PgDn] commands doesn't work and page navigation must be programed thru SYS_PAGE_PROSIMUL variable.

- **Process priority** : this parameter allow to adjust the Windows priority of PROSIMUL Threads.
- 1 : minimal priority for Runtime and Communication
- 2 : priority below normal for Runtime and Communication
- **3** : **normal** priority for Runtime and Communication
- 4 : priority above normal for Runtime and Communication
- 5 : maximal priority for Runtime and Communication
- 6 : « Time Critical » priority for Runtime and for Communication – 1ms precision is targeted for program cycle but **cpu usage increase**.

- **Process period** : this parameter allow to adjust the time cycle of Prosimul (from 0 = max to 999). By increasing this value you can reduce the time globally used by Prosimul on the computer. It's advisable to adjust "Process" period to the half of "Communication" period

- **Animations** : this parameter allow to choose the way the objects are refresh to optimize graphic view time when data changing. By "Page" all objects are redraw (use it for page with large pictures), by "Objects" only changing objects and their background are redraw. "Auto" mode select the best mode according to the 40 first animations. In "Page" mode or for objects on invisible background, antialiasing is quality less but display is faster.

If option "Page" is selected and a limit screen size too (graphic Grid tool), synoptic are auto-sized depending from the used screen.

- **Languages** : to adjust the Creation and Runtime languages according of the languages found in the dictionnary. If two differents languages are choose, then a translation is done when starting Run mode.

- **Unicode Editor** : to configure the Unicode editor file used for opening exported data (for sample : Wordpad, Word, Excel, Notepad++, …).

- **Protection** : to enable or disable application protection. If protection is chosen, a password is necessary to open a project (6 character maxi).

- **Share Data** : Allow to share database between several .SIM running applications, 1 master and 2 slaves (**Slave Data**) maxi. The running applications can operate R/W operations on the same database but only master application will give the database symbols (normally identical on all running applications).

- **Synchro Comm.** parameter allows to synchronize communications between each application (round token). This parameter is operational for native Ethernet driver only.

- **Process backup** : this parameter allow to select the database period backup (0=no 1…5). If backup is selected, variables are stored in APPLI_NAME.DAT file and can be used to initialize another version of simulation (the same or containing same symbols).

Graphic Visualizations (access by SYS_PAGE_PROSIMUL=0 to 15)

On RUN, page 0 displays itself. Changes of pages can be operated via the Application Navigator or automatically via system variable " SYS_PAGE_PROSIMUL " taking values 0 to 15.

- The associated functions keys with the symbols are operative.

Right and Left **click** can be **inverted** by putting value 1 in « **Runtime.Invert_Mouse_Click** » parameter of preferences.

- For the objects with forcing option **:**

. **a left click** on an object causes **the forcing** on its 1st animation variable. If this one is a binary type, the complement is directly make. If it is word type (integer or real) the seizure of its forcing value is proposed. In both cases the variable is not modifiable by program.

. **a left click** on an forcing object **cancels** its **forcing**. A search of forcing variables is possible via the dialog box of the variables.

. a **right click** or an **associated left click with [SHIFT]** on an object acts on its 1st associated animation variable. A complement is realized if variable is a binary type or the seizure of its value is proposed if it is word type. Its **forcing** (if present) **is cancelled** (the variable can change by program). If « Cmd_Mode » parameter is >= 100 and first variable is a boolean, variable is complemented during time of this parameter (time in ms).

. **a left double-click** on an **curved** object allow its printing.

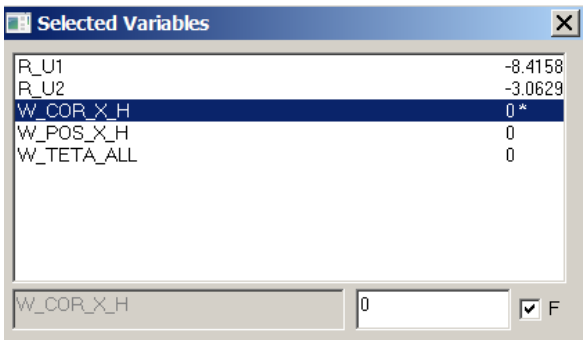
. **a click with [ALT]** on an object causes **forcing** from its 1st animation variable. The variable is not modified (value is **solidifying**).

- For the "SWITCH_3P" object : a click on left part of object cause the first associated variable to be set and the second reset, a click on right part cause the second variable to be set and the first reset. A click on middle position reset the two variables.

- For the "SWITCH_3P" object : a click on left part of object cause the first associated variable to be set and the second reset, a click on right part cause the second variable to be set and the first reset. A click on middle position reset the two variables.

- The "CURSOR" object can be drive by left and right arrows or moving cursor or click on associated value.

A right **click on an empty zone** give access to the list of variables selected thru the Monitoring function (see Monitoring).



Variables Vizualisations

- Associated functions keys with the symbols are operative.

- Variables are visualized in dynamics.

- Current values can be modified (forced or not).

- Forcing can be activated (check box). In this screen the modification of value of forced variables remains possible.

Ladder Visualizations

- Associated functions keys with the symbols are operative.

- Instructions can be visualized in dynamics (instructions access by **double-click**). True binary variables have their information text on green bottom. For words variables, current value is displayed instead of information text. The variable can be reached by one click on its information text .

- Command " Next " and " Previous " remain active.

- The command “VisuDyn” permit the dynamic visualization of variables used in current program. A double-click on variable suppress the selected variable from the list. Value can be changed by validation of new value in the edit zone. Forcing can be modified by clicking on the check box.

Literal Visualization

- The command “VisuDyn” permit the dynamic visualization of variables used in current program. A double-click on variable suppress the selected variable from the list. Value can be changed by validation of new value in the edit zone. Forcing can be modified by clicking on the check box.

Grafcet Visualizations (access by SYS_PAGE_PROSIMUL=20 to 27)

- Associated fonctions keys with the symbols are operative.

- Graphics are “real time” visualized in :

  . active step on green bottom
  . valid transition in green
  . forced associated variable in red.

- **Mouse Action**

  . **a simple left click** on a step/transition causes **the forcing** of its associated variable. A complement is directly made.

  . **a simple left click** on a forced step/transition **cancels** its **forcing**. A search of forcing variables is possible via the dialog box of the Variables.

  . a **right click** or **left click associated with [SHIFT]** on a step/transition acts as complement on its associated variable. Its **forcing** (if present) **is cancelled** (variable can be change by program).

  . **a click with [ALT]** on a step/transition causes **the forcing** of its associated variable. The variable is not modified (value is **solidifying**).

Functions Visualizations (access by SYS_PAGE_PROSIMUL=30)

- Associated functions keys with the symbols are operative.

- a simple **left click** on a block " **sub-page** " causes the entry in the under-page.

- a simple **right click** from **a sub-page** causes the exit towards the high level.

- I/O values are displayed and blocks " lamps " animate their outputs in green if the threshold in inputs is reached.

- **double-click** on "**Curved** " block function causes the displaying of the associated curves in inputs according to the maximum indicated. **A right click** closes this displaying. A second double-click allow its printing.

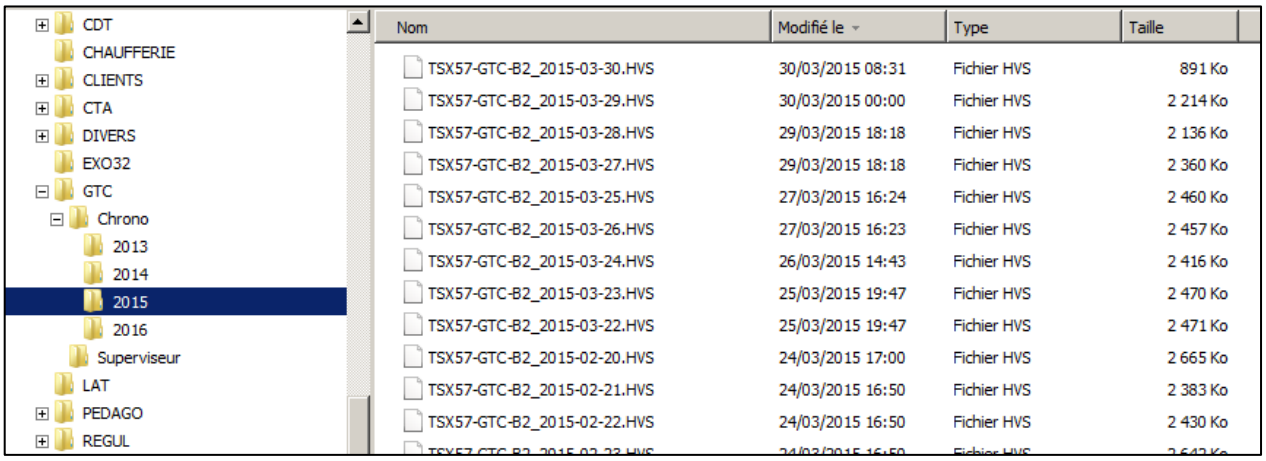
Connexions Veyon

If parameter « **Runtime.Minimize_if_Veyon** » is set to 1 in preference file (PROSIMUL.INI) and if an event (EraseBkGnd) appear and Veyon Master is connected (on **port 11100**), applications will be minimized to avoid screen conflicts.

MONITORING

Function accessible via the Navigator " Tool -> Monitoring " allows to select a list of variables (100 maxi) to be recorded in every change of state. Recording option must be declared with the precondition in application configuration.

APPLINAME_YYYY_MM_DD.HVS files are created daily in this case.
If "**Chrono**" subdirectory exists, files are stored and read from it.
If "Chrono\YYYY" (by year) subdirectories exist, files are stored by year and read from it.



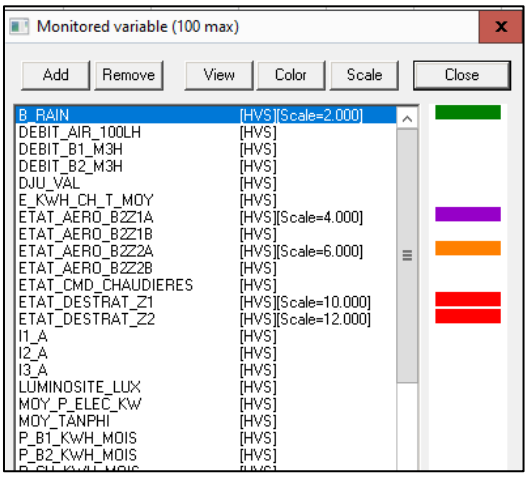
Variable references are daily included in HVS files.
Reference is made by Symbol and type of variables. If modification occurs in .SIM application after HVS storage, file « NOMAPPLI_SYMB-HISTORY.txt » contain history of modification. If this file exist (it can be deleted), modifications are included in displayed HVS files.

By default, in running mode, the current day file is loaded (if exist) and is dynamically displayed (only last 10.000 values).
Monitoring page can be called if SYS_PAGE_PROSIMUL equal 31.

Monitoring Menu

ScreenShot : displayed curves are stored to Windows Clipboard.

List Management : activate a dialog box allowing to select recorded and viewed variables.
→ Add / Remove : recorded variable selection
→ Tag [HVS] : variable exist in file
→ View (or double-click) : display or not the variable
→ Tag [Scale] & color : variable is currently displayed
→ Color : choice of display color
→ Scale : choice of display scale (ex : 0,01)



<< Day or ← : to see previous data file day (if non-existent search is done until 10th previous days).
Date : selection of a *.HVS file previously created .
Day >> or → : to see next data file day (if non-existent search is done until 10th following days).

Variables : selection in consigned variables displaying in *.HVS file. One coefficient and one color of displaying are proposed at the selections.

View -> Grid : display or hide background grid.
View -> Vertical Bar : display a vertical bar when cursor moved.
View -> Horizontal Bar : display a horizontal bar when cursor moved.
View -> Large pen : display values with 2 pixels pen.
View -> Point to point : display and export values as unlinked single points.
View -> Real Time : display values on realtime or relative time from origin.
View -> Mean Values : allow to do a mean function for N consecutive values (0=off).
View -> Up Threshold : seizure a maximum threshold to displaying (higher value if not).
View -> Down Threshold : seizure a minimum threshold to displaying (smaller value if not).
View -> Extremely points : display extremely points (squares located on curves).
View -> Inflections : displaying inflection points (circles located on the curves).

Parameters -> Store : save all displays and variables selected in a file named APPLINAME_*.INI.
Parameters -> Load : load a existing APPLINAME_*.INI file. Existing lists are accessible below this submenu.

Tools -> Polynomial : if <u>only one variable is displayed</u> this function calculate the polynomial coefficients of degree 2 to 12 according to least squares Method. These coefficients can be used in a " Polynomial " block function (In the lock of the dialog box indicating the result, the polynomial can be paste in the selected function page <u>before</u> this operation).

Export → Day : creation of a **APPLINAME_YYYY-MM-DD.HVX** file for all present values on the interval of selected time (see <u>Zoom</u>) with the selected Mean Value.
Export → Month : creation of a **APPLINAME_YYYY-MM.HVX** file for all the month including the current display day with the selected Mean Value.
Export → Year : creation of a **APPLINAME_YYYY.HVX** file for all the year including the current display day with the selected Mean Value.

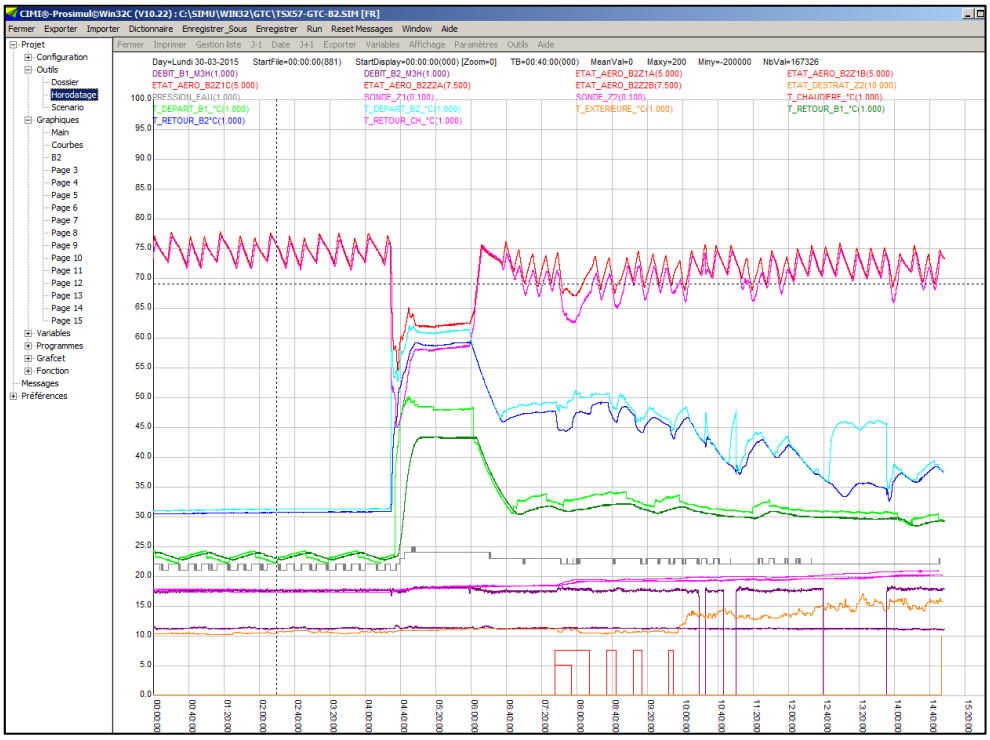
Export format :

Year	**MonthDay**	**Time**	**MilliSecond**	**Symbol**	**Value(10)**	**Forçage**

Where Value(10) mean that exported values are average of 10 values.
An utility under Excel can be provided to analyze data.

Displaying values

- a setting on ordinates scale is carried out automatically according to selected thresholds (initially maxima of the recorded values).



- time basis, regulated automatically and corresponding to the abscissa, is reported in top of the screen.

- symbols of the selected variables are indicated in their respective colors in top of the screen.

- movement of the cursor on the grid is reported like X=abscissa Y=ordinate in the right zone of message.

*Variables and selected displaying options are memorized booth in *.HVS file and in the global file "NOM_APPLI.INI" which is read for each new day viewing.*

Moving and Zooming Time

A **mouse selection of a screen zone** allow to select a range time. This function is available on 10 levels reported in top of the screen (Zoom=0... 9).

Right click or [ESC] command causes a Zoom out of time.

In Run mode, time scale is automatically shifted according elapsed time.

Left and right keys allow to shift time from one time base.

The **mouse wheel** allows to increase or decrease displayed day.

SCENARIO

Recording or reading option must be declared with precondition in Application before passage in RUN.

In the **recording** phase, a **APPLINAME.SVA** file is created. This contains all manual operations made on variables (variables bases, functions keys, forcings...).

Scenario is acceded via the navigator " Tools -> Scenario " which allows to modify recorded values in the form of **APPLINAME.TXT** textual file (**tabulation separator**). Values and variables can be modified (particularly to paste monitored values). Text format is compatible (times & values) with the text format of Monitoring export and is :

Time MilliSecond Symbol Value Forçage(Yes=FORC)

Example

Time	MilliSecond	Symbol	Value	Forçage
00:00:02	000	W_POLYMER_AFF	1	
00:00:03	000	R_HEAD_PC_EDT_SP	70.0	
00:00:03	100	O_BP_HEAD_ADJUST	1	FORC
00:00:03	200	O_BP_HEAD_ADJUST	0	
00:00:04	000	O_P_42MB_W_RUN	1	
00:00:08	000	O_BP_M_SS2_START	1	FORC
00:00:08	200	O_BP_M_SS2_START	0	
00:00:20	000	R_V_FCV5201A_SP	25.0	FORC

Edit : allow to modify current scenario (NOMAPPLI.SVA) thru text file NOMAPPLI.TXT. When TXT file is closed, scenario NOMAPPLI.SVA is built with it.

Load : allow to modify any TXT file correctly formatted and to use it as current scenario.

*.SVA file can **be replayed** automatically (Application Configuration / Scenario / Play). The time system coefficient is apply to time of recorded values. All operations present in file will be executed.

IMPORT / EXPORT

Prosimul allows exchange of textual files (.TXT) for the symbols definitions (with initial values) and Graphics+Models files (.BIM) for exchanges between applications (.SIM). The same command import/export will be used for the two uses according to the opening or not of the variables dialog box.

Variables Exportation(since Variables box)

- Select in the variables dialog box, variables to be exported.
- Choose the command “export” in the principal menu.
- Give a file name (.TXT extension)
- Exported file is viewed with Notepad - Windows (notepad.exe).
- File format :

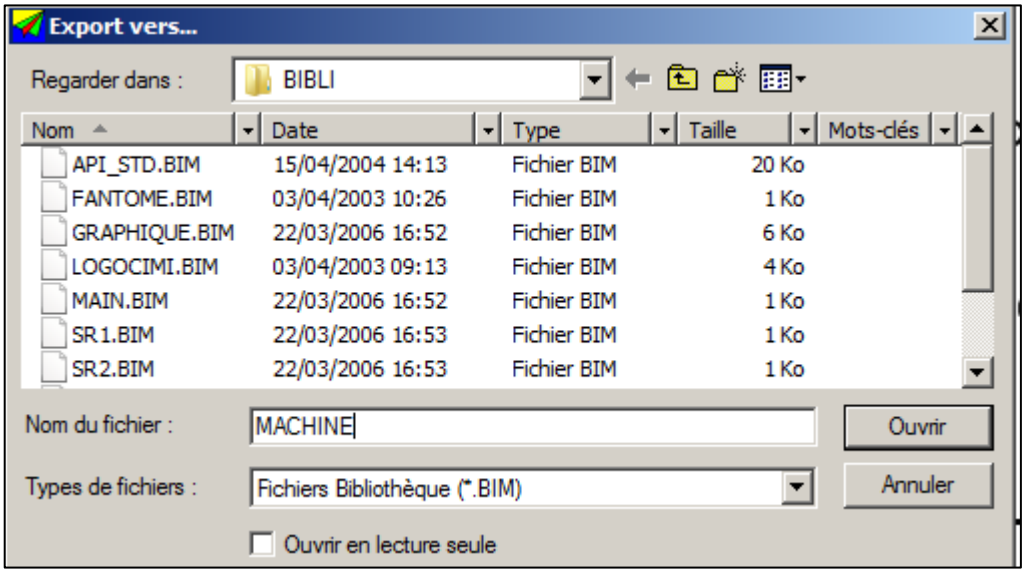
  SYMBOL**[tab]**value (real values recognized by separating point of the decimals).

Variables Importation (since Variables box)

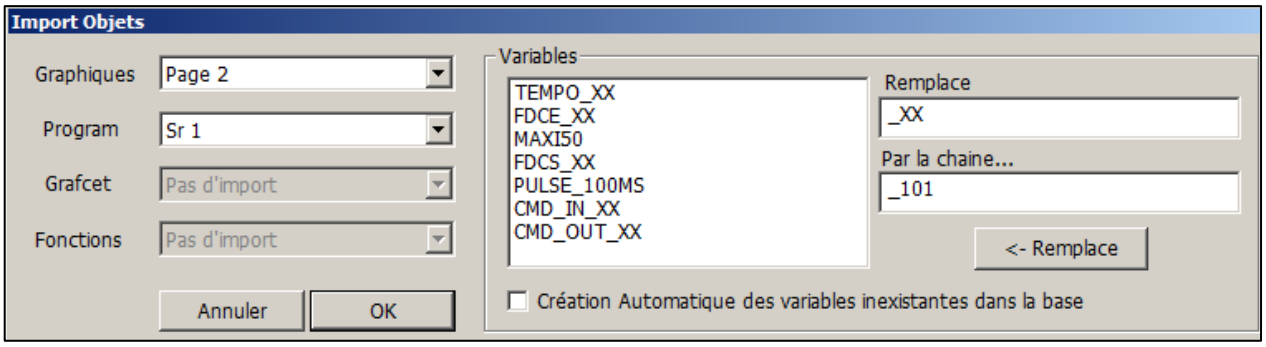
- Select in the variables dialog box, the first variable receiving the importation.
- Choose the command “import” in the principal menu.
- Choose a file name (.TXT extension).
- If some symbols already exists behind the first selected, you can choose between cancelling operation or overwriting variables. A detection of identical symbols is executed. If the initial value exists (SYMBOL**[tab]**value) this one is recovered.

Graphics and Models Exportation (.BIM)

- Export all **selected objects** through the different editors. This function is available for the Graphic/Ladder/Grafcet/Functions editors.
- To export graphics, Ladder instructions, Grafcet and function blocks in the same file .BIM, repeat the selection / export operations in each editor and choose the same export file.
- All selected components will be exported in a binary file (**.BIM**) containing also associated symbols with the objects. You can replace objects in the .BIM file by repeating the selection/export operations. In that case you must confirm this replacement through a message box. In all case be careful to export objects that use the same Variables DataBase.



Graphics and Models Importation (.BIM)



- Import on **chosen pages,** selected elements contained in .BIM file (graphic and models).

- It's possible to avoid of elements by selecting "No Import" option.

- Import of Grafcet and Functions require empty pages selection.

- The import operation fail if the number of objects by editor exceed the maximum.

- For graphic pages, imported objects are grouped by default.

- **a creation/replacement of the present symbols** in .BIM file is proposed through a dialog box. You can :

. Import first all the necessary variables in database via .TXT file and simply validate.

. Create or modify all or part variables. Double-click on each variable in list will project in the variable dialog box on the adequate variable type. Create or select the selected symbol (right click call back the ten last symbols used).

. Replace a string in all symbols of the list (ex : _ 01 by _ 02). In this case, use the "<- Replace" button after complete the fields " Replace " and " By String ".

. Choose the "Automatic creation of inexistent variables on database" option. In that case all inexistent symbols will be created on free variables at the end of database. The variables types are respected. The used Grafcet steps are replaced by new steps.

CONVERTER

DOS File existence

- Only .IMA files (graphic), .CFG (variable), .ANI (animations), .ACT (Ladder), .XTR (ladder Grafcet), GRF (Grafcet) and FCT (blocks function) are necessary for the conversion of DOS applications.
It's possible to permute the black and white colors (take care of Ladder programs that use codes of colors).

DOS Applications Constraint

- Variables " BIT A 1 ", " TOP DIXIEME ", " TOP CENTIEME " and " INIT GRAFCET " must exist on bits 0 to 3 of DOS application.

- Bits SYS_TOP_DIXIEME and SYS_TOP_CENTIEME are now square signals. Envisage the use of a rising front (**P#**) in the converted programs.

- Variables " RIEN ", " HORLOGE ", " CYCLE PROSIMUL ", " PAGE PROSIMUL " and " COEFF TEMPS(%) "must exist on words 0 to 4 of DOS application.

- the B1023 bit must be free of use.

Variables

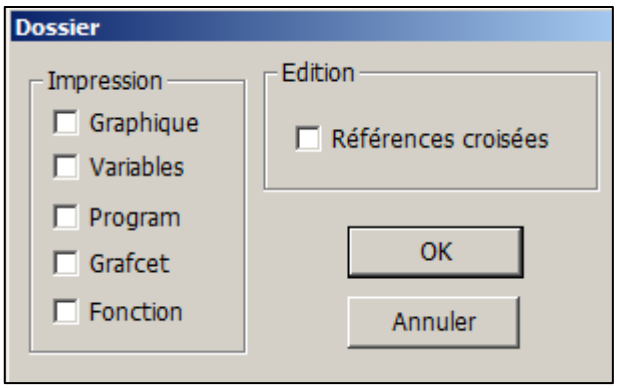
- Variable " BIT A 1 " " is replaced by the binary value 1 (0 for its complement).
- Variable " RIEN " is replaced by the real value 0.

Ladder conversion

- As soon as only 10 instructions remain available in the " Main ", instructions are introduced into the SR (1, 2...) and an unconditional call (CALL) is operate since the " Main ".

- Relative JUMPS are substituted by labels (Be careful : however with SR division which can cause loops caesuras).

PRINT REPORT BOX



- The report are printed selected elements in the zone "print"' (graphic, variables lists, Ladder programs, Grafcet and/or Functions) on the selected printer and edit Xref with notepad if selected.

- Access to Report mode can be done only in STOP mode.

- Printer and orientation can be selected.

EVOLUTIONS

V12.07

12.07A4 (04/08/2026)

NEW **DataBase** : copy Tags is possible in Run mode (for searching tool or for clipboard).

12.07A2 (20/07/2026)

NEW **Communication** : new driver « ENIP (CIP-TAGS) » - Data addressed by Tags.

NEW **Graphics** : new object « TANK_TETA » (rotative – product expressed in volume).

NEW **Graphics** : new object « TABLE_TETA » (rotative bargraphe).

CORR **Graphics** : sometimes page background color will not be displayed correctly.

12.06C2 (15/03/2026)

NEW **Graphic editor** : new object « Line_Bezier » .

12.06C1 (22/12/2025)

MOD **ENIP driver**: UC parameter is available up to slot 19.

12.06B2 (15/11/2024)

NEW **Stuctured Text** : new functions « ConnectToTcp », « Disconnect », « ModbusReadRegister ».

CORR **Monitoring** : corrective data Exports.

CORR **Function blocks** : fix of links between blocks.

12.06B1 (22/10/2024)

NEW **Graphics** : 2 new objects used for measurement (Measure and Surface).

CORR **Function SendMail** : body of message is missing (V12.06 only).

CORR **Modbus TCP** : configuration problem for Modbus client target (V12.06A2 only).

12.06A2 (10/07/2024)

NEW **Monitoring**: new tactile commands ← ESC →.

NEW **Modbus client**: FC04 request is now usable in addition to FC03 and FC23.

CORR **Graphics**: correctives of superposed objects draw (only for V12.06A1).

CORR **VarDyn Box**: corrective of system variable display.

12.06A1 (21/06/2024)

NEW **Structured Text**: new instruction GetTcpPortState.

MOD **Graphics**: GDI and GDI+ calls are rewrite to avoid problem with VEYON.

CORR **Graphics**: some correctives applies on dialog boxes (properties, colors, …).

CORR **Stuctured Text**: minor correctives for InfoApp(4 ;0) et MsgLog(;) functions.

V12.05

12.05A9 (12/12/2023)

MOD **Monitoring** : new version of Management box.

CORR **Graphics** : old Groups (converted application) haven't correct variable type.

12.05A3 (15/11/2023)

CORR **Littéral - Box** : some OS do not use multi-instance DLL and 12 box will be enabled (DLL is modified).

NEW **TimeStamp** : new function "ScreenShot".

MOD **Videos** : new version of Pro32Video.exe (V12.05) that run only with PROSIMUL V12.05.

CORR **Structured Text** : function "Replace_all" work on all ST programs.

MOD **Structured Text** : new parameters for InfoApp et MouseCtrl functions.

V12.04

12.04A6 (13/09/2023)

NEW **Scenario** : new function "Load".

NEW **Graphics** : new object « LISTBOX2 » (animation of border/text color).

NEW **Function blocks**: new blocks "Share" and "Screw".

NEW **Runtime**: in Developer Mode, running application with [SHIFT] key will cause to hide explorer.

CORR **Structured Text** : « Search Next » function stop to search if empty program is found.

CORR **Graphics** : corrective of random bug of Objects Property Box.

NEW **Graphics** : new curve « CURVE_PC » (values given in %, with or without / x-values).

NEW **Littéral** : nouvelles fonctions de conversion « Fahrenheit » ←→ « Celsius ».

NEW **Ladder** : nouvelle commande « Coller+Suivant ».

NEW **Littéral** : la fonction « InfoApp » peut retourner les origines X, Y du synoptique.

NEW **Littéral - Box** : possibilité de gérer 10 exécutables externes.

NEW **DialogBox** : nouveau bouton (code 64) avec Bitmap (256 couleurs max).

V12.03

12.03A3 (12/01/2023)

CORR **Graphics** : corrective of new Rotative conveyor display.

CORR **Graphics** : corrective of curve display after program visualization.

12.03A2 (05/01/2023)

MOD **Pilotes Ethernet :** parameter « Retries » is changed for parameter « Delay before retry (s) ». Parameter « Period (ms) » become « Connexion timeout(s) » for server drivers. Ethernet drivers restart in case of timeout.

NEW **Graphics** : new object rotative conveyor.

CORR **Structured text** : keyword « then » now detected when starting a line.

CORR **Runtime** : application activity is now detected to avoid to close all application in case of crash.

CORR **Ladder** : shortcuts « Shift » « / P N F » now doesn't operate inside comments.

NEW **Grafcet** : down links are now usable (only below transition).

CORR **Structured text :** return values are corrected for instructions GetSX, GetX1, …

V12.02

12.02C2 (23/05/2022)

CORR **Ethernet drivers** : corrective of "Retries" parameter (**-1**) for English version.

12.02C1 (17/05/2022)

MOD **Ethernet drivers** : if parameter "Retries" is **-1**, driver stay active even if comm errors occur.

12.02B6 (04/04/2022)

NEW **Graphics** : new tool "**Screenshot** ".

NEW **Graphics** : new tool for Resizing all objects.

NEW **Graphics** : new parameter for Box for vertical alignment on frame of windows.

NEW **Graphics** : new 3 ways valve.

MOD **Graphics / Tanks** : if first character of comment is | tank is closed.

CORR **EIP server** : communication can't restart if max attempts is reached (software must be closed).

CORR **Structured Test editor** : problem with BackSpace and Return (characters could be deleted).

CORR **Runtime** : problem with right click (unforc) on VALEUR_T2 et LISTBOX objects.

12.02A5 (08/11/2021)

CORR **Strutured Text** : "Replace all" function doesn't operate correctly sometime.

CORR **Ladder Timer** : if BT=0, a loop stop program execution.

CORR **Modbus TCP** : FC23 timeout fixed.

12.02A4 (11/05/2021)

NEW **Communication** : a local fix IP (integated in SIM application) can be configured.

NEW **Structured Text** : after a Run search operation, last viewed program is displayed when coming back in Stop mode.

NEW **Structured Text** : new function « Replace all ». Search and replace all occurrences of text in all Structured Text programs.

NEW **Structured Text** : a double dash « -- » at end of line allow to write a comment.

NEW **Runtime / Graphics** : Right and Left clicks can be inverted by putting value 1 in parameter « Runtime.Invert_Mouse_Click » of preferences.

NEW **DataBase** : I/O addresses can be visualized in X.Y format by putting value 1 in parameter « DataBase.IO_HexaMode » of preferences.

NEW **DataBase**: keys [Ctrl]C/V/X/Insert and [Shift]Suppr/Insert are now working (copy/paste/del).

NEW **Monitoring**: possibility to show or hide grid.

MOD **Graphics** : boxes Colors/Font/Styles : only modified or re-validated parameters have effect, especially in case of multiple selection.

MOD **Graphics** : magnetism function no longer modify point 2. This one can be positionned on grid by moving point 2 lonely (with key [CTRL] or mouse).

MOD **Structured Text – Window Variables** : double-click no longer delete variables from list.

MOD **Graphics** : mouse wheel drive now « Zoom+ » and « Zoom- » functions. To modify dimensions of multiple objects, you must group them and move point 2.

CORR **Structured Text** : an unsuccessful search brings back to initial program.

CORR **Graphics – Animation** : in case of symbol renamed if Properties box is opened, new symbols used in box are modified. Parameters /#, P#, …, are kept except in case of new symbol selection.

CORR **Communication** : S7-ISO-TCP driver doesn't start (for V12.01D4 only).

CORR **Graphics** : rare cases of bug when displaying out of range float value (Value object).

V12.01 (Upgrade)

12.01D4 (16/03/2021)

NEW Timestamps : values can be displayed and exported as single points.

NEW **Structured Text** : MsgLog function accept one variable (%MWx% or %MFx%) each line.

NEW **Runtime** : display each 5s of % CPU power used by application in Run mode.

MOD **Runtime** : with priority 6, program will not wait for period but only skip it schedule place. This guarantee a 1ms time cycle precision but use more cpu power.

CORR **Structured Text** : dynamic variables in Window are now sorted (english version only).

CORR **Runtime** : of time precision (1 ms) for Runtime and Comm tasks for **Windows 10**.

CORR **Communication** : all drivers. Momentary fault of transmission to PLC in rare cases.

12.01C2 (07/01/2021)

NEW **Structured Text** : new parameters 6 & 7 for StopApp function.

MOD **Runtime** : time cycle management is enhanced (runtime and exchanges).

MOD **Function Blocks** : Multiplier and Order1 functions have Initial and Minimum values.

CORR **Timestamps** : of exported timestamp values with "relative time" selection.

CORR **Timestamps** : bug possible in case of many changes of data list.

CORR **Graphics** : sometime, « Cursor » object doesn't display well formatted values.

CORR **Structured Text** : WinExec function with Param2 <> 1.

12.01B2 (09/10/2020)

NEW **Communication** : **Modbus** TCP-IP : FC01 / FC15 request added.

CORR **Communication** : **DF1** serial driver.

CORR **Runtime** : forcing binary value fail if object accept multi-format variable.

12.01A5 (03/08/2020)

NEW **Graphics** : new object « OBJ2D » in group « Rotation 2D ».

CORR **Windows 10** : access to 64 bits registry for Windows 10.

CORR **Structured Text** : sending email. Numeric values where omitted.

12.01A1 (20/01/2020)

NEW **Communication** : local IP address used can be selected in Exchange configuration.

NEW **Communication** : new driver EthernetIP server use to answer to Rockwell Ethernet Module.

NEW **Graphics** : new graphic 3D object (format .ASE, .DAE, .OBJ) using PRO3D.exe viewer.

MOD **Communication** : Ethernet drivers are optimized (all protocols).

MOD **Structured Text** : function « SendMail » use now SendMail.exe software.

CORR **Timestamps**: minor display faults corrected.

CORR **Project explorer** : doesn't display correctly new graphic or program pages.

DEL **Conversion** : conversion of MS-DOS application is no longer available.

DEL **Keys** : parallel keys are no longer recognized.

V11.06

11.06H6 (14/09/2020)

CORR **Communication** : **DF1** serial driver.

CORR **Runtime** : forcing binary value fail if object accept multi-format variable.

11.06H1 (10/12/2019)

CORR **Runtime** : bug when starting Run mode after "Structured Text" program was opened then closed.

CORR **Structured Text** : multiple beeps when compiling "Structured Text" program or viewing it in run mode.

11.06G5 (17/04/2019)

MOD **Graphics**: « Clean » tool is improved.

CORR **Communication**: Ethernet drivers : long delay before restarting communication is fixed.

CORR **Graphics**: « Sound » object (*.wav) now work correctly.

V11.06G3 (21/02/2019)

NEW **Runtime** : Monitoring page can be called if SYS_PAGE_PROSIMUL equal 31.

NEW **Runtime / Functions** : Zoom In and Out actions are stored in scenario.

V11.06G2 (21/01/2019)

CORR **Project Explorer** : point of insertion for new intermediate pages.

CORR **Graphics**: right-click fault if zoom is different of 100%.

V11.06F1 (18/11/2018)

MOD **Graphics**: animation colors of new object is "page color" by default.

CORR **Runtime**: groups initially hidden when starting application aren't displayed.

V11.06E4 (29/08/2018)

NEW **Function Blocks** : P/I/D values of controllers can also be written in manual mode.

CORR **Ladder**: application may dysfunction if a comment is located before end of program.

V11.06E2 (07/08/2018)

NEW **Function Blocks** : P/I/D values of controllers can be transfered in DataBase.

MOD **Communication** : S7-ISO-TCP driver. UC slot 1 is now accessible. Configuration must be changed.

CORR **Graphics** : display of real value for "Symbol" or "Value" tables is corrected.

V11.06D1 (16/06/2018)

NEW **Graphics**: new command "Paste Next" that use paste with next variables in Database if root is the same (except system variable).

MOD **Graphics** : "Dialog Box" object. Code Box 32150 mean that text separator is tabulation.

V11.06C3 (04/05/2018)

NEW **Graphics** : new graphic object elliptic "Line".

CORR **Communication** : OPC client : connection to server is continuously tested to avoid bug if user shutdown OPC server.

CORR **Graphics** : background color of dashed line.

11.06B2 (15/01/2018)

NEW **Ladder / ST**: new instructions to open / close dialog box (DialogOpen, DialogClose).

NEW **I/O Cards** : integration of new analog outputs cards « IPCDAS / PIO-DAxU » with « UNIDAQ » driver. « PISO-P32A32U » binary I/O cards must now use « UNIDAQ » driver.

CORR **Ladder** : bug when closing (X) Ladder dynamic instruction dialog box.

11.06A3 (04/12/2017)

CORR **Graphics** : « ListBox » display / use in « Page » mode.

CORR **Help** : page calls to PDF help file.

11.06A2 (17/11/2017)

NEW **Graphics** : new object like user « Dialog Box » describe in file text .TXT.

NEW **Runtime** : Application configuration : new mode « Supervisor » which avoid to refresh background picture (for faster cycle time) and suppress « Graphic » menu in Run state.

NEW **Graphics** : new object like « ListBox ».

NEW **graphics** : « Value » objects can display "Time" value depending from predefined format (idem Text object). Time value is expressed in milliseconds starting at 0 hour.

NEW **Ladder / ST** : new instructions for mouse position control.

NEW **Graphics** : new object « Value_T2 » which value can be directly modified on object.

NEW **Graphics** : for all objects with binary command drived by« Cmd_Mode ». Value 21 mean to always force to 1 and value 20 mean to always force to 0.

NEW **Graphics** : « TEXT » object can receive « Cmd_Mode » value equal to 10 to allow text modification (29 characters max) directly by clicking on object position. Les valeurs 30 et 31 permettent d'afficher l'heure et la date système.

NEW **Graphics** : 2 new pumps + 2 new motors.

NEW **Function Blocks** : new block « Average ».

MOD **Function Blocks** : gain of « Conveyor » block is splitted on each step.

MOD **Graphics / Application** : additional colors (16 to 31) are stored in .SIM file.

CORR **Function Blocks** : initial values for conveyor block.

CORR **Structured Text** : instructions GetSX, GetSY, …, for moving objects.

V11.05g2 (15/01/2018)

- **Corrective** : bug when closing (X) Ladder dynamic instruction dialog box.

11.05g (06/10/2017)

- **Création** : Objet « Rotation2D » : pour les objets « Rotation2D » type 3 et 4 création d'un « Mode contour » (1=contour non affiché).
- **Modification** : Éditeur graphique : la fonction « Undo » est basculée sur la combinaison de touches [CTRL][Z] (idem Microsoft Office). La touche [ESC] sert à désélectionner tous les objets.
- **Modification** : Pilote EIP : la limite maximum du total des I/O échangées passe de 127 à 254.

V11.05f (01/08/2017)

- **Corrective** : « Moist Air » instructions : all values of pressure are standardized in Pascal unit
- **Corrective** : Runtime : math errors are intercepted before application crash. Stop mode is automatically called if more than 4 errors occur in the same cycle (see LOG file).
- **Corrective** : Dialog box position in multi-screens mode.
- **Creation** : graphic editor : new object "3 ways valve".
- **Corrective** : object.
- **Creation** : new functions « TahHsRs(kJ ; kg/kg as) » and « TrPv(Pa ;0.0) ».
- **Corrective** : Ladder / Structured Text : PVapeurPatmRs function adjusted.
- **Corrective** : rarely, key could be unrecognized at RUN to STOP state changing (V11.05e only).
- **Corrective** : Modbus TCP-IP driver with FC23 request. Some null outputs state could be send between correct states.

11.05e

- **Modification** : in RUN mode, dialog box limit their freeze time (especially when left click remains pressed).

11.05d

- **Corrective** : rarely, one new state of variable could be non-transmitted to PLC.
- **Corrective** : "read-only" attribute on .SIM file forbid any save operation (manual or automatic).
- **Corrective** : in Run mode, if SYS_PAGE_PROSIMUL attempt to display empty page, an error message is displayed and action is blocked.
- **Corrective** : copy of variable on animation dialog box dysfunction if « OK » button is used to close the dialog box after copy operation.
- **Corrective** : valve objects : corrective of animation (pixels could stay on screen).
- **Modification** : dictionary : if an active dictionary exist, labels modification of « Text » object are stored in active language of dictionary.

11.05c
- **Creation** : new “Frigo” functions
- **Corrective** : Y movement for GROUP object are corrected (inverted).

11.05b
- **Corrective** : upgrade : value format prior than V11 is preserved.
- **Corrective** : Graphic editor (English version only) : fail to create new object.
- **Creation** : ST : F1 and F2 allow to introduce / suppress begin and end of commentary.

11.05a
- **Creation** : graphics : new object « Notepad » allowing to display text file.
- **Corrective** : Getxxx functions for cylinders are limited by stem outlet.
- **Corrective** : fonction « Undo ».
- **Creation** : Pro32xx.EXE could be launched with M1 à M5 parameters (screen 1 to 5).
- **Creation** : graphics : new “Vidéo” object working with Pro32Video.EXE player and “VideoLoad” instruction.
- **Creation** : Ladder and ST : new function “ScreenParam”.
- **Corrective** : runtime : minor unsightly display in “Objects” mode.
- **Corrective** : print report incomplete.
- **Creation** : font size of Ladder and ST editors can be adjusted thru preferences.
- **Creation** : colors for Ladder list can be adjusted with preferences.
- **Creation** : a « User » palette with 16 colors can be used. Colors could be adjust thru « Preferences » in project explorer and are stored in PROSIMUL.INI file. In the Color Dialog Box, a « Palette » button allows switching between « System » or « User » palette of colors.

V11.04b (17/01/2017)

- **Corrective** : EIP driver : CID number is coded on 2 bytes with low byte of IP client address. (in case of multiple client PC communicating with same IP server).
- **Corrective** : Runtime : in case of dialog box opening, graphic animations stay actives.

V11.03d (15/12/2016)

- **Corrective** : of dialog boxes in english version (boxes that always stay on topmost position).
- **Corrective** : graphic editor : minor correctives of display un “page mode”.
- **Creation** : new integration of parallel I/O box Sealevel 8227 on USB port.
- **Corrective** : OPC driver : application fault if configured with a uninstalled OPC server.
- **Corrective** : graphic editor : on page change with background color, flickering problem is solved.
- **Corrective** : Structured Text Editor : program backup could be partial under virtual system.
- **Creation** : projects are protected from double use on the same computer.

V11.02 (09/11/2016)

- **Corrective** : ENIP driver : added “Stop” session and “Restart” session functions.
- **Creation** : Graphic editor : Line object can be non-displayed by giving -1 to Display or/and Clean value colors.
- **Creation** : Graphic editor : « Paste Object Only » function (paste without animation parameters).
- **Corrective** : “Group” animation (CACHE & X/Y couldn’t work simultaneously).
- **Creation** : Ladder/Litteral :
  new instruction « % » : like MOD (modulo)

new instruction « >< » : Abs Near of… : OP1><OP2 equal Abs(OP1)<=OP2
ex : B_CAPTEUR = (W_POSITION – 100) >< 10
→ B_CAPTEUR is true if W_POSITION is include between 90 and 110
operators « + , - , *, / , % » could be use in logical equation in Literal langage

- **Modification** : Ethernet drivers : « Sleep » time becomes variable (10% of communication period). This modification is done for powerless computer.
- **Modification** : Ethernet drivers : received data can be largest than 255 bytes.
- **Creation** : Graphic editor : display and magnetism steps can be adjusted separately.
- **Corrective** : Graphic editor : minor display correctives for boxes “Colors”, “Styles” and “Fonts”.
- **Modification** : Graphics/Ladder/ST/Chart : unused items aren’t displayed in Project Explorer. A **right click** on the root of items allow to access « **Add** » menu.
- **Creation** : Ladder/ST : « Call » instruction can operate with program names. Ex Ladder : CALL#MYPROG Ex ST : Call ‘MyProg’ (between simple quotes).
- **Modification** : Ladder/ST : available programs are up from 10 to 16.

V11.01 (09/10/2016) - Upgrade

- **Modification** : Modbus/EIP/S7-ISO server : communication period apply only for the read operation of PLC outputs. The write operation of PLC inputs is done immediately if inputs are detected updated in PROSIMUL application. The communication cycle time is improved on average by 30%.
- **Creation** : local IP address (for Modbus-TCP/EIP/S7-ISO-TCP) could be selected depending from number of application in PROSIMUL.INI.
- **Corrective** : “EIP driver : communication may not restart after a communication error.
- **Modification** : “Synchro communication” parameter is now available for EIP driver.
- **Corrective** : Literal : array of numeric variables can be greater than 100 elements.
- **Corrective** : Literal : bug may occur when passing en Run mode if a syntax error is detected.
- **Modification** : Literal editor : maxi characters per line is up to 1024 and maxi operand number is up to 15 operands per line.
- **Corrective** : Chart : error of symbol location when building chart.
- **Corrective** : bug may occur when replacing objects in .BIM library.
- **Creation** : Graphic editor : “Group” object have 2 animations : “Hide/display” and/or “Move X, Y position” of grouped objects.
- **Creation** : Graphic editor : function Zoom+, Zoom- available.
- **Corrective** : Graphic editor : « Cursor » object could take values out of max limit.
- **Modification** : Graphic editor : “Undo” function works with 5 levels.
- **Creation** : Graphic editor : « Background » function.
- **Modification** : Graphic editor : straight line selection zone is limited around the line.
- **Modification** : multi-screen selection is available by moving windows or with « Window→ Select screen » command.
- **Creation** : screen resolution change is managed (when new screen is selected).
- **Modification** : main window can be resized.
- **Modification** : serial communication state is visualized OK with green background in right position of status bar. On communication error, background status become red. If any variable is in forced state, background becomes yellow. Status display format is :
hh :mm :ss [max execution cycle duration] [F: number of forced var] [C: exchange count]..

V10.30 (11/07/2016)

- **Modification** : EthernetIP driver : files and variables above 254 are accessible.
- **Modification** : links of function block are redraw.
- **Creation** : Runtime : auto-sizing of graphics if parameter "Animations" is on "Page" (Configuration/Application) and screen limit is selected (Grid tool).
- **Creation** : Graphic editor : [ALT] key allow to select successively backgrounded objects.
- **Creation** : Graphic editor : new tool for objects group rotation (right click).
- **Corrective** : occasional error when calling open/save file dialog (V10.29 only).
- **Corrective** : error may occur when calling Visudyn menu.
- **Creation** : Graphic editor : new objects « Table_Bargraphe ».
- **Création** : Ladder / Literal : new function AtgPi(x ;y) returning angle between -180° to +180°.

V10.29 (01/02/2016)

- **Creation** : when starting simulation, SYS_INIT_APPLI pulse to 1 during the first cycle.
- **Creation** : Application Configuration : current DAT file can be used to initialize unforced variables.
- **Creation** : in RUN mode, a backup of variables is done each 5 minutes in a file named APPLI_NAME.DAT.
- **Modification** : standard "File dialog box" are used.
- **Corrective** : printing on Windows 64 bits is now efficient.

V10.28 (04/12/2015)

- **Modification** : All communication drivers : high and low bytes can be switched for digital or analog values.
- **Corrective** : Object "curve auto" – Display error if Y scale is equal to 1.
- **Corrective** : Runtime stop – Error could occur if Ethernet communication fail.
- **Correctives** : Monitoring – List of variables + Data export + Data suppression.
- **Corrective** : Graphics – Decimal values in curve object were not correctly displayed.

V10.27 (09/09/2015)

- **Modification** : Graphic Editor – "Clean" menu. Type of animation variable is fully controlled.
- **Corrective** : bug can occur when creating Literal program (V10.26 only).
- **Modification** : S7-ISO-TCP driver : high and low bytes are switched for analog values.
- **Corrective** : of communication retries when timeout occur (S7-ISO-TCP and ETHERNETIP).

V10.26 (30/06/2015)

- **Creation** : Ladder/Litteral : function StopApp can display other Prosimul application.
- **Creation** : Ladder/Litteral : StopApp function can stop other Prosimul application.
- **Creation** : Ladder/Litteral : new InfoApp function (to collect information of other loaded Prosimul applications).
- **Creation** : Graphic editor : new triangular « Mobile » object horizontally oriented.
- **Creation** : Ladder/Literal: for information instructions (GetSX, GetLeftX, …), second operand can be either number of object or label of object with quote delimiters like GetSX(0;"Tapis1").

- **Creation** : Function Blocks: in Run mode dynamic I/O value are displayed.
- **Creation** : SYS_PAGE_PROSIMUL variable can call Grafcet pages (value 20 to 27) and Function Blocks (value equal 30).
- **Creation** : numeric values can be input with graphic dialog box.
- **Creation** : Graphic Editor : new "Curve" object that allow to show time on abscissa.
- **Corrective** : graphic problem may occur in case of short PROSIMUL period (< 10ms).
- **Corrective** : problem may occur when ending PROSIMUL in case of long serial timeout (> 500ms).
- **Corrective** : Function blocks may dysfunction in case of short PROSIMUL period (< 10ms).
- **Modification** : GetDateTime function with first parameter equal to 10 return elapsed milliseconds in the day (since 0h).

V10.24-25 (30/05/2015)

- **Creation** : Ethernet TCP/IP drivers : TCP port can be selected.
- **Modification** : Print Report : for PDF printer one file is used for all outputs.
- **Creation** : Master/Slaves applications : Modbus client communication can be synchronised → see Application configuration, parameter "Synchro Data".
- **Modification** : Print Report : for PDF printer one file is used for all outputs.
- **Creation** : "Cmd_Mode" variable for graphic object binary animation : if value is >= 100, click on object now generate a pulse >= 100ms (only for new created objects).
- **Modification** : Ethernet IP configuration : "Channel" of target module can be selected behind IP address (ex : 192.168.2.66/3).
- **Modification** : if 2 Function blocks "Conveyor" are linked (FWD->FWD or REV->REV), flows (FWD or REV) are synchronised on the same time cycle.
- **Modification** : Graphic editor : "Table_Fill" and "Table_Value" can be oriented top to bottom or bottom to top.
- **Corrective** : crash may occur when passing from Stop to Run or Run to Stop mode.
- **Corrective** : Grafcet and Function Blocks editors. Minors bugs when selecting objects.
- **Corrective** : Graphic editor: problem may occur when creating "Viewmeter" or "Cursor" objects.

V10.22 – 23 (31/03/2015)

- **Modification** : Help : HLP format is no longer supported. Only PDF file is updated.
- **Modification** : Modbus protocol: slave number can optionally be configured after IP address (ex: 192.168.2.65/2 means slave 2).
- **Modification** : Modbus protocol: high byte or low byte order transmission can be configured thru new field. 1= High Byte first (Normal), 0=Low Byte first. Modbus applications must be reconfigured.
- **Corrective** : Runtime. Animations fault if "Initialization" option is not selected..
- **Creation** : Graphic editor. New command "Duplicate" (=Copy/Paste).
- **Creation** : Graphic editor. Curve object now store 1000 latest values.
- **Corrective** : Graphic editor. Return value of animated GIF.
- **Corrective** : Graphic editor. Bug when creating new object (English version only).

V10.20-21 (08/01/2015)

- **Creation** : Commands [Ctrl][C/V/X] are available for all editors.
- **Corrective** : "Cursor" object. Left/Right commands do not work properly.
- **Creation** : New function block PID standard (the first model has parallel structure).

- **Creation** : When opening APPLI.SIM project, APPLI.BAT file can be executed (if existing).
- **Corrective** : BODE function (only for V10.19 version).

V10.19 (09/09/2014)

- **Corrective** : Ladder : functions TBCOP and NCOP (only for V10.18 version).
- **Creation** : Ladder/Literal : new functions for flow treatment : « DeltaP », « Flow »..
- **Corrective** : Literal editor : 2nd operand of functions can be a result of operators.

V10.18 (23/07/2014)

- **Creation** : Ladder/Literal : new functions for moist air treatments :« DeltaZero », « PVapeurPatmRs », « SVapeurTah », « PVapeurTahHr », « RoAirPasTabs », « RoVapeurPvTabs », « RsPvPas », « HsTahRs », « HkDtRs ».
- **Modification** : Ladder/Literal : instructions set is standardized and execution time optimized.
- **Creation** : Graphic editor : new object Desk « Bargraphe » with N piloted level.
- **Corrective** : Grafcet editor : “renumber” function modify existing labels in Literal language.
- **Corrective** : Variables editor : “increment” function modify existing labels in Literal language.

V10.16-17 (23/06/2014)

- **Creation** : interface with CoDeSys OPC server is integrated.
- **Corrective** : Monitoring. Limitation of 10.000 dynamic display values (application may crash for wide quantity depending from configurations).
- **Corrective** : PID function block. Min saturation did not work properly.
- **Corrective** : Starting PROSIMUL without any configured printer could crash software.
- **Corrective** : Monitoring : time display period greater than 24 hours is corrected.
- **Corrective** : Monitoring: multiple access on stored data could fail (RUN mode).
- **Corrective** : Function blocks : some hazardous lines could appear when passing in STOP. Blocks links are modified.
- **Creation** : begin of test with Windows 64 bits (in STOP mode without communication).

V10.15 (13/05/2014)

- **Creation** : Graphic editor : [L] key and Grid menu allow to display a screen user limit (no, 800x600, 1024x768, 1280x1024).
- **Corrective** : Ladder : instruction “EBM” (writing bit of word – obsolete).

V10.14 (01/04/2014)

- **Creation** : Graphic editor : [0] key allow to modify all animation colours for selected object.
- **Creation** : Monitoring: Parameters lists can be loaded and stored thru “Parameters” menu. Existing lists are accessible in this menu.
- **Creation** : Monitoring: Data files are search at +- 10 days around current date.
- **Creation** : Graphic editor : new object “Animated GIF”.
- **Creation** : New function “StopApp” (read Lock state and allow to stop / restart application). See

Literal chapter.

V10.12x → V10.13 (06/03/2014)

- **Corrective** : OPC configuration : PLC target name with S/R/P/N/U letter.
- **Corrective** : Graphic editor : TAB_SYMB display float values in wrong format.
- **Corrective** : EthernetIP driver : Low and High bytes inversion and bug for socket initialization.
- **Corrective** : Monitoring function could modify exchange configuration.
- **Creation** : Festo Usb Easyport access (64DIO+16AI+8AO max) is integrated.
- **Creation** : New function "MsgLog" use to write user message to PROSIMUL.LOG file. See Literal chapter.

V10.11 (31/01/2014)

- **Modification** : variable inputs. New shortcuts (keyboard and mouse right-click).
- **Corrective** : Literal : "for … next" loop with decremented index.
- **Creation** : when passing in STOP mode, SYS_CYCLE_PROSIMUL variable is set to -1 during one program cycle (LADDER/Literal). This can be used for stop mode detection.
- **Creation** : in case of Communication error, SYS_COMPTEUR_ECHANGE variable is set to -1 and can be tested.
- **Modification** : Monitoring : HVS files directories can be organised by years (\Chrono\YYYY) .
- **Corrective** : OPC client : bug when starting RUN mode two times.

V10.10 (10/01/2014)

- **Corrective** : Unicode texts memory allocation. Could cause PROSIMUL "crash".
- **Creation** : Graphic editor. New keyboard shortcuts.
- **Creation** : Graphic editor. New "PIPE" objects with border and background colour animations.

V10.06/07/08/09 (02/01/2014)

- **Corrective** : text and value default parameters (font, alignement) are store in .SIM project.
- **Corrective** : graphic editor : pictures with null size (RATIO_X.Y) can be selected in Stop mode.
- **Corrective** : runtime period is adjust to 10ms for new application and can't be configured below 1ms.
- **Corrective** : of import/export function (bug with 10.05 version only).
- **Correctives** : Grafcet and Function Blocks editors (cells moving and cursors).
- **Modification** : function blocks « Intégrator », « Sommateur » et « Polynôme » have new parameters « Minimum » and « Initial value ».
- **Modification** : Monitoring : left and right keyboard action allow to move time from one time base before or after current display.
- **Creation** : graphic object « TABLE_SYMB » (section « Various ») allow to display a table of symbols and values that be force in RUN mode.
- **Modification** : for graphic objects « TABLE » with "force" parameter enabled, in Run mode, the forced value in the table depends on the position of the mouse click.
- **Modification** : Ladder editor : wheel mouse operate.
- **Modification** : Ladder/Literal editors : main program can be written in Ladder or Literal language.

- **Creation** : Function editor : new bloc « Split-Range » (Warm/Cold or Quick/Precise functions).
- **Modification** : New exchange configuration dialog box + Ethernet frames up to 1024 bytes.
- **Creation** : Graphic editor : new object Desk "PUSH-BUTTON" (with two circles).
- **Modification** : Graphic Editor. By shifting [CTRL] key, user can select objects by zone without selecting background objet.
- **Modification** : Graphic Editor. [M] key toggle magnetism function.

V10.05 (01/08/2013)

- **Creation** : "**S7-ISO on TCP**" communication driver to access S7 PLC thru Ethernet way.
- **Corrective** : of "Copy / paste " function in graphic editor.
- **Creation** : Monitoring function : displayed variables and display parameters can be store or read from different .INI files.

V10.04 (28/06/2013)

- **Creation** : "Copy/Paste" functions of editors work now with ClipBoard. It available copy/paste operation between different .SIM applications with same databases.
- **Corrective** : of Monitoring dynamic visualization (bad time scale and sometimes crash of application).
- **Creation** : Graphic Editor : texts can be draw with Fixed Vertical Size or Automatic adaptation to Horizontal Size depending from len of chararcters.
- **Creation** : Monitoring function : Export function can work for current display day or for all month or year including the current display day. An utility under Excel can be provided to analyze data.
- **Creation** : Graphic Editor : a right click with selected Value objects allow to reformat all values.
- **Modification** : Runtime start on the last displayed graphic page.

V10.03 (23/05/2013)

- **Creation** : Monitoring function : if a subdirectory "Chrono" exists, HVS files are store and read from it by priority.
- **Creation** : Monitoring function : selected variables and options are stored in the global file NOM_APPLI.INI which is read (if exist) on each day viewing or when reopening TimeStamp function.
- **Creation** : Monitoring function : a "Mean Values" tool is created (0=off).
- **Corrective** : Integer values are displayed with the selected number of digits.
- **Modification** : Bode function is improved.
- **Creation** : slave applications (Share database) are automatically close when master application get stop.
- **Creation** : Function block PID with parameters Kp, 1/Ti, Td, Td/N and limits for integrator saturation.
- **Modification** : Max variables database is increase to 2048 instead of 1024.

V10.02 (02/04/2013)

- **Corrective** : minors bugs (Grafcet editor, Function blocks, CURSOR graphic object with integer association, ...).

- **Corrective** : exchange configuration thru S7-ONLINE driver.
- **Creation** : daily HVS timestamp file can be reinitialized on each running start.

V10.01 (22/03/2013) - Upgrade

- **Creation** : **Monitoring function**.
HVS files are stored by days continuously. Selected displayed variables are transmitted from day to day. File name must be regular : NAMEAPPLI_YYYY-MM-DD.HVS.
The timestamps menu allows to navigate to next or previous days or to load a specific date. By default, the current day is loaded and dynamically displayed for the current day. 100 variables (25 in previous version) can be monitored and are ranked by alphabetic order.
- **Modification** : **VisuDyn** tables (FORCE / LITERAL and LADDER language / TIMESTAMP LIST). Those tables are now ranked by alphabetical order.
- **Creation** : SendMail function is created (see Literal language chapter).
- **Creation** : WinExec function is created (see Literal language chapter)
- **Creation** : GetDateTime function is created (see Literal language chapter)
- **Creation** : PROSIMUL applications (3 max) can work in master-slave mode (application configuration). One application is master of database and 2 others can communicate with other PLC for example). The exchanged binary I/O origin can be configured (Exchange Configuration).

V9.05 (XX/01/2013)

- **Corrective** : minors bugs of data display in timestamp function.
- **Creation** : graphic object “Analog-Viewmeter” is created.
- **Creation** : graphic object “Cursor” is created. This object can be controlled by right and left arrows or cursor movement.
- **Modification** : Timestamp file format is modified (file .HVS) to include Day and Time start of records.
- **Creation** : graphic editor : the mouse wheel associated with [CTRL] key allow a precise and homothetic resizing function.

V9.04 (12/10/2012)

- **Modification** : objects « Text », « Panel », « Pipe », « Motor », « Pump » can be animated with X,Y movement.
- **Creation** : OPC MATRIKON (Matrikon.OPC.Universal.1) Server access is available.
- **Creation** : EthernetIP driver (Rockwell). Ethernet access to PLC with integrated port or ENI module. Access is available only to Nx file (for Tags please configure mapping).
- **Creation** : A Lock Menu is available in Run Mode. If the application is protected, the password must be typed. All commands are locked.
- **Modification** : when dictionary is imported, inexistent translations are labelled “-----Not Found----“ in PROSIMUL.LOG.

V9.03 (19/10/2011)

- **Creation** : access to OPC server Kepware.KEPServerEX.V5.

V9.02 (06/05/2011)

- **Corrective** : OPC server communication name was incorrect for Rockwell.

V9.01 (15/04/2011)

- **Corrective** : correctives of some minor display faults (objects containing texts).

V9.00 (15/03/2011)

- **Creation** : .SIM application can be protected by password. This password is configurated in Application configuration dialog box.

V8.02 (xx/08/2010)

- **Creation** : remote OPC server access. Choose the OPC server name and fill the « Target Address" field. WARNING: all existent OPC configurations must be revalidate (see Chapter Communication/OPC in this documentation).
- **Creation** : MODBUS TCP-IP access can use FC03 + FC16 or (new) FC23 (R/W multiples registers – unavailable on many PLC).
- **Corrective**: in some case, after grouping objects operation, the graphic page could generate Windows crash. A new menu is create to clean old application (Graph / Clean).
- **Corrective**: for centred or right justified text.

V8.00/V8.01 (25/04/20100-10/05/2010)

- **Creation**: graphic objects label becomes **Unicode** coded. They can store international font characters. A multilingual **dictionary** (see Main Menu) can be exported or imported for different Runtime language views. Creation and Runtime languages are declared thru Application Configuration in Navigator. Applications must be converted (Upgrade V7->V8).
- **Creation**: New MODBUS-TCP/IP server thru WinSocks (waiting for client connexion).

V7.30/31 (18/07/2008)

- **Creation**: Graphic editor – The "Cmd Mode" (0=one push to force object, 1=continuous action to force) works with "Text" objects.
- **Creation**: Ladder and Literal editors – New instruction "WriteIni(OP1;OP2) that allow to change the value and the initial value of OP1 with the current value of OP2..
- **Creation**: Exchange – New protocol "MODBUS-TCP/IP" direct to PLC thru WinSocks.
- **Modification**: Runtime – dynamics variables lists are sorted by types.

V7.29x (21/04/2008)

- **Creation**: New menu “Window” that allow to switch between PROSIMUL applications.
- **Creation**: The mouse wheel is operating in Graphic and Variables pages.
- **Creation**: Graphics – the windows can be moved by mouse wheel and keyboard (direction / Home / End keys).
- **Creation**: Variables editor – positions on each tables are memorized.
- **Corrective**: Ladder – the unforc instruction “U#BIT” operate correctly (with logic conditions).

V7.28x (17/03/2008)

- **Creation**: Communication – OPC and DataBase interfaces with Direct-Link SW1000 (Applicom / Woodhead) validated.
- **Corrective**: “Dev.Only” graphic objects are displayed only in development mode.
- **Corrective**: “graphic editor – foreground function fail if more than 400 objects on page.
- **Modification**: literal editor – double-click selection select entire word.
- **Corrective**: “graphic editor – problem with colour display for “table” objects.
- **Corrective**: Bug when quitting Run mode if the mouse move on graphic page that use grouped objects.

V7.27 (Bêta1 – Bêta2)

- **Corrective**: Report function (problem if graphic page isn’t opened).
- **Corrective**: Application crash when comments exist in Ladder in certain cases.
- **Corrective**: Bug when quitting Run mode if the mouse move on graphic page that use grouped objects.
- **Creation**: Graphic editor: new object “IMAGE” in the “Others” family (BMP/GIF/JPG/PNG/TIF/EMF format accepted). This object can be moved, resized and rotated.
- **Creation**: Graphic editor: a variable named “Cmd Mode” is defined for “Desk” and “Pipe” objects. It allow to force the in two modes (value 0=push one time to force, 1: continuous action to force).
- **Creation**: Graphic editor: the size modification of a graphic objects group is possible thru Property dialog box.
- **Creation**: new functions Int(X ; Y) and Float(X ; Y) usable in Ladder and Literal.
- **Creation**: new functions that allow to get dimensions, position and coordinates of graphics objects (GetSX, GetSY, GetX1, GetY1, GetX2, GetY2, GetLeftX, GetTopY, GetRightX, GetBotY). Used syntax : Get…(page number, object number) using information obtain with “Object Properties” dialog box (in title bar).
- **Creation**: Graphic editor: new object “Conveyor” in “Rotation_2D” family.
- **Creation**: Graphic editor: new objects “TABLE_FILL” and “TABLE_VALUE in “Others” family. They allow to display some values with different colours (like FILL object) and with contents. The first element of the table must be the table size.
- **Modification**: Function blocks : the internal values of “Delay” and “Conveyor” blocks can be writed to Database thru the VALUES parameter.
- **Corrective**: Instruction of table copy in Literal editor . Ex: TAB1[N] = TAB[N].
- **Corrective**: Grafcet – erased cells could be associated with non null variables.
- **Corrective**: Literal – writing 0 value to bit array can produce bad result.
- **Modification**: Application Configuration – The graphic mode for drawing objects can be choose

according to the best time response; by “page” all objects are redraw; by “objects” only evolving object (and their background) are redraw.

V7.26 (19/04/2007)

- **Modification**: New parameters X and Y offset for “2D Rotation” objects.
- **Creation**: New function hypotenuse (Hypo) for Ladder and Litteral language.
- **Creation**: New graphic object “HINGE” in graphic editor.
- **Modification**: For real values the characters “.0” are automatically added if non existent.
- **Corrective**: Literal Editor – Screen movement corrected.
- **Modification**: Graphics objects can be sized in “Animation Box”.
- **Corrective**: Literal Editor – Logn instruction corrected.
- **Modification**: Grafcet and Functions Editors : blocks can be moved by mouse.

V7.25 (03/04/2007)

- **Corrective**: Problem of memory size in Runtime mode.
- **Modification**: Double clicking on .SIM application if configured in “Training mode” launch directly the application in RUN mode .
- **Modification**: The “No printer found” is now redirected to PROSIMUL.LOG file.

V7.24 (29/03/2007)

- **Corrective**: Problem of memorization of Literal Compilation solved when RUN mode start.

V7.22/23 (15/03/2007)

- **Modification**: Graphic Editor – Animation of distributor T1 modified.
- **Modification**: Graphic Editor – Animation of jack actuator – The background colour is used to fill the emptying part.
- **Modification**: Literal Editor – Instruction MOD is tested for null modulo.
- **Modification**: Literal Editor – The presence of separator “:” is tested in program structure.
- **Modification**: Literal Editor – The commentary can include few lines. It begin at the line including (* and finish after the line containing the two attached characters *)
- **Creation**: The Runtime period can be adjust thru the Application Configuration DialogBox.
- **Corrective**: OPC client is modified to avoid any starting problem on XP system.
- **Corrective**: Inaccessible functions with Runtime Key are locked.

V7.21 (17/12/2006)

- **Corrective**: Literal Editor – The variable replacement operate on variable that are preceded by tabulations.
- **Corrective**: Literal Editor – Some “End line” characters may be automatically corrected by editor. A message is display in that case

V7.20 (13/12/2006)

- **Creation**: Literal Editor – The colours of Comments, Variables, Operators, Keywords, .., can be selected thru the Preferences in the project navigator (or thru PROSIMUL.INI file loaded at the start of PROSIMUL) .
- **Modification**: The messages are now stored in PROSIMUL.LOG file (but not in Errors.txt file).
- **Modification**: Variables Editor – Functions "Cut/Paste/Suppress/Insert" are modified to allow operations between "Cut" and "Paste".
- **Corrective**: Literal Program backup modified (problem when passing in RUN mode after Validation).
- **Corrective**: Graphic Editor – The objects with "Dev.Only" attribute but without animation are draw correctly.

V7.16 (22/11/2006)

- **Corrective**: Variables Editor – Modifications, Insertions and deletions are correctly modifying Literal programs.
- **Corrective**: The "Communication Lost" message is now displayed blinking in the status bar.
- **Creation**: The different categories of messages are displayed with different colours.
- **Creation**: A right-click in a Variable Edit zone, allow user to go in the DataBase (last option in context menu).

V7.15 (31/10/2006)

- **Corrective**: OPC Client – The VARIANT Objects are initialised to limit memory usage.
- **Corrective**: Runtime – The execution of Function Blocks is reactivated.
- **Corrective**: Translate tool – The translation using .TRA file operate on graphic pages and by option on variables and programs.
- **Corrective**: The "Communication Loss" information stop correctly when quitting RUN mode.
- **Corrective**: Runtime Ladder Visualization – Problem when closing Dynamic Visualization of Timer configured with small Time Base.
- **Creation**: Literal program – As Variables are modified in database, Literal programs are automatically modified.
- **Creation**: The Process Priority can be adjust thru "Configuration Application" dialog box.

V7.14 (18/10/2006)

- **Modification**: In case of variables (not Words) insertion/deletion, the variables in the communication table are conserved.
- **Corrective**: OPC client – Code source is modified to avoid memory usage increase in RUN mode with communication.

V7.13 (06/10/2006)

- **Creation**: Literal editor – New command [CTRL][A] to select entire text.
- **Modification**: In case of program unforce command (U# in Ladder or UnF in Literal), the unforce action is done only if the variable is forced (to reduce display time).

- **Corrective**: In the “Variables” window (RUN mode), the selected variable is freeze to allow its value to be modified.

V7.12 (26/09/2006)

- **Corrective**: In RUN mode, in case of “INIT_APPLI”, problem of Literal program lost.
- **Corrective**: In RUN mode, the “Variables” window can slow down the exchange cycle.
- **Corrective**: In case of communication lost, the click out of the acknowledge box can freeze the menu commands -> cyclic message.
- **Corrective**: In case of variables insertion/deletion, the variables in the communication table are conserved.

V7.11 (09/09/2006)

- **Modification**: OPC exchange configuration : a list of installed OPC servers is displayed.

V7.10 (09/06/2006)

- **Modification**: LITERAL Editor – Compilation speed is improved.
- **Corrective**: LITERAL Editor – Problem of lost lines when passing from RUN to STOP.
- **Creation**: Graphic Editor – New object “LINK_CURVE” (mechanic family).
- **Creation**: Graphic Editor – New object “SWITCH_3P” in Panel family. This object can pilot 2 variables (XOR function) by clicking on left or right part of the object in RUN mode.

V7.08 (02/05/2006)

- **Creation**: In RUN mode, the list of variables selected thru “Timestamp” function can be dynamically visualized thru “Tools” “Timestamp” in Prosimul Explorer, or by right click on empty zone of graphic pages.
- **Corrective**: Literal – Priority of operators *, /, +, -.
Operators * and / have same priority and are executed from left to right. Operators + and - have same priority and are executed from left to right. The priorities are defined in the Literal section of the documentation.
- **Corrective**: Literal – Automatic spacing after commentary lines.

V7.07 (28/04/2006)

- **Creation**: In RUN mode, the list of forced variables can be accessible thru the menu command Force -> List.
- **Creation**: Ladder / Literal – In RUN mode the command VisuDyn of the Ladder / Literal menu permit the dynamic visualization of used variables in the current program.
- **Corrective**: Literal Editor – Detection of sentence with insufficient operators.
- **Corrective**: Literal Editor – Operators / and * are tested in inverse order.

V7.06 (04/04/2006)

- Modification: Graphic Editor –Animation Box / new display of Points coordinates.
- Corrective: Literal Editor – Bad interpretation of Boolean comparison without operator (execution is correct).
- Corrective: Variables Editor – impossibility to modify / force value in RUN mode.

V7.05 (28/03/2006)

- Corrective : Literal –operations (PLUS, MOINS) deliver correct values.

V7.04 (24/03/2006)

- Modification: Literal – Functions can be imbricate in themselves (maxi 10 operators by line).
- Creation: Graphic Editor – New “mechanic” object named LINK.
- Corrective: Graphic Editor – Animation of vertical valves.
- Corrective: Graphic Editor – Lines CPOS and CNEG when size is null.
- Corrective: Literal – Bad duplication of program when creating new SR.

V7.03 (15/03/2006)

- Corrective : Literal – Transfer INT to FLOAT with operators.
- Corrective : Literal – Seizure of the negative real values (ex : -1.0).
- Corrective : Literal – Put in 1 or 0 possible of the bits extracted from words (ex : M10 . 4 = 1).
- Creation : Ladder / Literal – functions ArcSinus (Asin) and ArcCosinus (Acos).
- Creation : Animation – A double-click on the box properties of Graphics objects allows to modify all animation colors.
- Creation : Converter – Possibility of choosing a permutation between black and white colors.
- Creation : Graphics – New types of objects : "mechanics".

V7.02 (08/03/2006)

- Corrective : Converter - Update according to the evolutions 7.01.
- Corrective : Import – Replacement of unitarian variable.
- Corrective : incorrect animation of objects " FILL " when the FRAME is informed in 0.
- Corrective : Variables – impossible selection of variables SYS_ in version 7.01 only.
- Modification : Variables import – Control doubloons and speechless characters.

V7.01 (27/02/2006) - Upgrade

- Corrective : Literal Editor – Interpretation of the logical equations with double parentheses.
- Corrective : RUN mode –« Priority - » command allowing to decrease to level 1.
- Corrective : Mode RUN –refresh screen problem on page changes.
- Corrective : passage in STOP mode with doubles Escape key.
- Modification : in upgrade case of .SIM file, characters replacements are made in the following way : characters "**(**" "**)**" "**,**" "**;**" "**:**" "**#**" "**.**" are replaced by the character "**_**"

characters "<" ">" "=" "+" "-" "*" "/" "**%**" "**[**" "**]**" are respectively replaced by the characters "**L**" "**G**" "**E**" "**P**" "**M**" "**X**" "**D**" "**R**" "**{**" "**}**"
- Modification : Communication with OPC client – the target datas must be indicated in textual form (ex : %MW for Schneider, DB20 for Siemens, or TAGNAME for Rockwell).
- Creation : by default, colors are created in variables, SYS_COLOR_BLACK…WHITE into integers 5 to 20.
- Corrective : Literal – « mathematical » operations (COS, SIN,…) made with variables of integer type deliver correct values.

V6.02

- Corrective : Variable Editor - variable insertion / deletion problem when a Literal program exists before a ladder program.
- Corrective : Literal Editor – variable insert function since Database.
- Creation : Literal Editor : list of last 10 variables in contextual menu.
- Corrective : Literal Editor : possibility of unforcing all types of variable (Unf)
- Corrective : interval functions ?1, ?2, ?5, ?10 in Ladder and Literal mode.
- Corrective : IMAGE object in movement can be forced.
- Creation : in Formation mode (application configuration) explorer is not displayed and page changes are made with [PgUp] and [PgDn] keys or the SYS_PAGE_PROSIMUL variable.
- Modification : Literal – FOR instruction works with an optional parameter STEP.
- Creation : Run – PROSIMUL process can be risen or lowered of Windows priority by the commands « Priority + / - ».
- Creation : Graphic Editor : 2 new objects of text type accepting 4 font colors (T_VISU) or 4 background colors (T_FOND).
- Modification : TEXT objects : if thickness of Style is greater than 1 text is displayed in bold.
- Corrective : Boolean and numeric operators – correct distribution of the intermediate results.

V6.01

- Modification : DF1-Direct driver - for automats PLC it will be no longer necessary to multiply the addresses of words by 2.
- Creation: the .SIM applications are now used with **versioning**. The oldest versions must be converted when opened. This conversion is not reversible.
- Correction : the translation files .TRA are now treated with or without blank line.
- Correction : the graphics objects with “Dev.Only” property can’t be forced no longer.
- Modification : in all serial drivers, the parameter “Retries” permit to configure the number of failed exchanges before the driver stop.
- Correction : for CONVEYOR block – The accumulation is now correctly treated is speed is null. The initial state is also correct. The speed values admissible are integer from –24 to +24 (0 = stop; 1 = one step by one period).
- Modification : the targets “PROSIMUL” and “PRO32” are now indicated for Rockwell driver and OPC client and can be modified.
- Correction : Graphic Editor : an object position larger that 999 is authorized.
- Correction : the initial state of SYSTEM variables is unlock.
- Modification : the call of function keys must be associated with characters « **{** », « **}** » in symbol.
- Creation : LITERAL language. The choice can be done between LADDER or LITERAL for all sub-routines Sri.
- Creation : Main Menu - Raz_Errors : enable the deletion of errors.TXT file.

V2005.12a

- Creation : software protection enabled with ALADDIN USB keys.