



*The professional software suite
for automatic control design and forecasting*



EICASLAB DEMO

User Manual

Version 1.0

www.eicaslab.com

How to contact EICAS Automazione S.p.A.

www.eicas.it	web site
support@eicas.it	mail support
+39 011 5623798 +39 011 5623088	phone
+39 011 4360679	fax
Via Vincenzo Vela, 27 10128 Torino Italy	address

Further information are available at EICASLAB web site www.eicaslab.com

EICASLAB DEMO 1.0

COPYRIGHT 2007 - 2014 by EICAS Automazione S.p.A.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from EICAS Automazione S.p.A.

Printing History: September 2008

First printing for EICASLAB DEMO 1.0

EICASLAB DEMO

The powerful DEMO version of the EICASLAB software suite

This manual aims at describing all steps necessary to run successfully the EICASLAB DEMO.

Getting started.....	3
<i>System requirements</i>	<i>3</i>
<i>Installation procedure.....</i>	<i>4</i>
<i>Uninstallation procedure.....</i>	<i>14</i>
<i>Running EICASLAB DEMO</i>	<i>16</i>
<i>DEMO projects</i>	<i>17</i>
<i>Overview of EICASLAB.....</i>	<i>20</i>
<i>Running projects</i>	<i>21</i>
<i>MASTER window menu</i>	<i>22</i>
SIMBUILDER.....	25
<i>The layout concept</i>	<i>25</i>
<i>System layout</i>	<i>26</i>
<i>Programming mode</i>	<i>27</i>
<i>SIMBUILDER window menu</i>	<i>28</i>
Scheduling management.....	33
<i>Schedule window.....</i>	<i>33</i>
<i>Buttons</i>	<i>35</i>
Popup menu	36
<i>Block popup menu.....</i>	<i>36</i>
<i>Layout popup menu.....</i>	<i>38</i>
Navigating the project	39
Graphical mode	40
<i>Libraries</i>	<i>40</i>
<i>Control blocks.....</i>	<i>43</i>
<i>Macros</i>	<i>44</i>

<i>Graphical Macro</i>	46
<i>C Macro</i>	52
<i>Macro upgrade</i>	61
<i>Subsystems</i>	63
C mode	64
<i>Control files</i>	65
<i>Functions</i>	66
<i>File Manager</i>	67
SIM	73
<i>Preparing User project for simulation</i>	74
<i>Running the simulation</i>	79
<i>Managing the simulation</i>	81
<i>Debugging the User project</i>	83
<i>SIM window menu</i>	86
POST	91
<i>Preparing project for post-processing</i>	92
<i>Record Data for POST</i>	95
<i>Running post-processing</i>	96
<i>Processing window</i>	97
<i>Processing window menu</i>	102
<i>Plotting window</i>	105
<i>Plotting window menu</i>	110
Appendix	115
<i>I/O Control variables for User project</i>	115
<i>POST variables</i>	116

Getting started

EICASLAB DEMO is a demonstration version of the EICASLAB software suite. It has been designed to provide you with all the basic features of the commercial version, together with a predefined set of examples (DEMO projects) regarding standard professional applications.

This document is addressed to briefly describe how to install the DEMO and how to run the examples. Additional information is presented in the official documentation of the EICASLAB commercial version (EICASLAB User Manual).

System requirements

Note

EICASLAB DEMO was **compiled for 32-bit Operating System**. Running it on 64-bit architectures might require the installation of 32bit libraries. In case of problems in finding the correct libraries please contact EICASLAB Support (support@eicaslab.com).

EICASLAB DEMO is designed for Linux and Windows OS. The following table shows the system requirements for installing EICASLAB for both the operating systems:

 Linux	 Windows
x386 processor architecture any Linux distribution kernel 2.6.x or later glibc 2.3 or later (www.gnu.org/software/libc) 512MB RAM 250MB disk space	x386 processor architecture Windows NT, 2000, XP, Vista, Windows 7 512MB RAM 500MB disk space

Installation procedure

EICASLAB DEMO can be downloaded for free from the EICASLAB web site www.eicaslab.com, depending on your operating system.

Installation file for Linux O/S

For Linux OS, the installation file is called `EICASLABDEMO1.0.tar.gz`.

The first installation step consists in extracting the content of `EICASLABDEMO1.0.tar.gz` in your working directory. Open a Terminal Console in the directory where you have downloaded the installation file and digit the command:

```
tar zxvf EICASLABDEMO1.0.tar.gz
```

The result is a directory called `EICASLABDEMO1.0` containing an `Install` script. In order to start the EICASLAB DEMO Installer you have to type:

```
./EICASLABDEMO1.0/Install
```

Note

Make sure you have enough permissions to extract and work with the content of `EICASLABDEMO1.0.tar.gz` file.

Windows installation procedure

For Windows OS the installation file is called `EICASLABDEMO1.0.zip`.

For installing EICASLAB in your PC, please download it from the login.eicaslab.com website and extract the downloaded package `EICASLABDEMO1.0.zip` in a working directory.

The result is a folder called `EICASLABDEMO1.0` containing the `Install.exe` file. Run it to start the EICASLAB DEMO Installer.

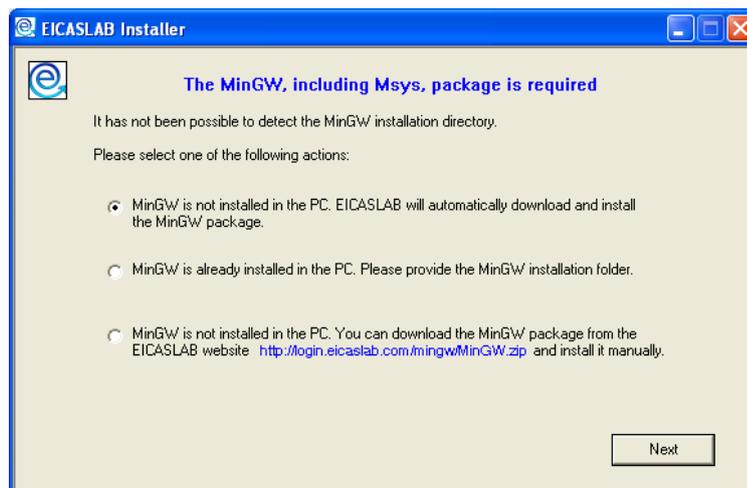
EicasLab DEMO for Windows requires the MinGW and Xming packages. During the installation phase EicasLab will automatically check if these packages are installed in your system and, if they are not present, it will ask you to automatically download and install them.

The MinGW and Xming packages can also be manually installed. All the instructions for manually installing them are described in the “[MinGW and Xming Installation's Guide](#)”, downloadable from the EicasLab DEMO web site.

In the following it will be described the automatic procedure that EICASLAB runs for installing the required packages if they are not found in your system.

Once the Install.exe file is launched, if the MinGW package, including Msys, is not found EICASLAB will show you the following window in which you can choose among the following items:

- **MinGW is not installed in the PC. EICASLAB will automatically download and install the MinGW package.** You can choose this item if you want EICASLAB to automatically install the MinGW package.
- **MinGW is already installed in the PC. Please provide the MinGW installation folder.** You can choose this item if the MinGW package is installed in your PC but EICASLAB is not able to detect the installation folder.
- **MinGW is not installed in the PC. You can download the MinGW package from the EICASLAB website <http://login.eicaslab.com/mingw/MinGW.zip> and install it manually.** You can choose this item if you want to install manually the MinGW package.



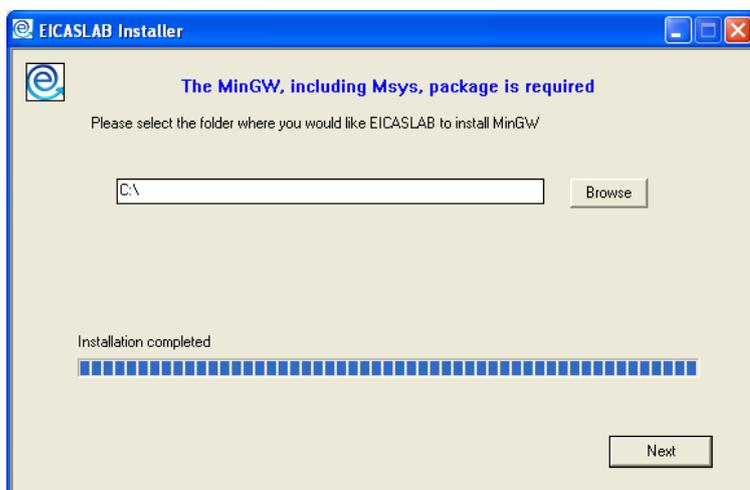
If you choose the first item the following window gets opened where you can provide the folder where you want EICASLAB to install the MinGW package.



By clicking the **Next** button the installation procedure starts and the following window gets opened.

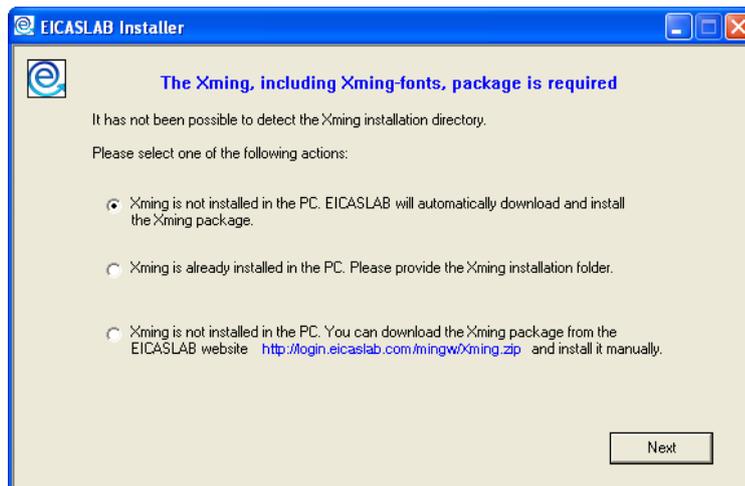


When the installation of the MinGW package is completed you can click the **Next** button.

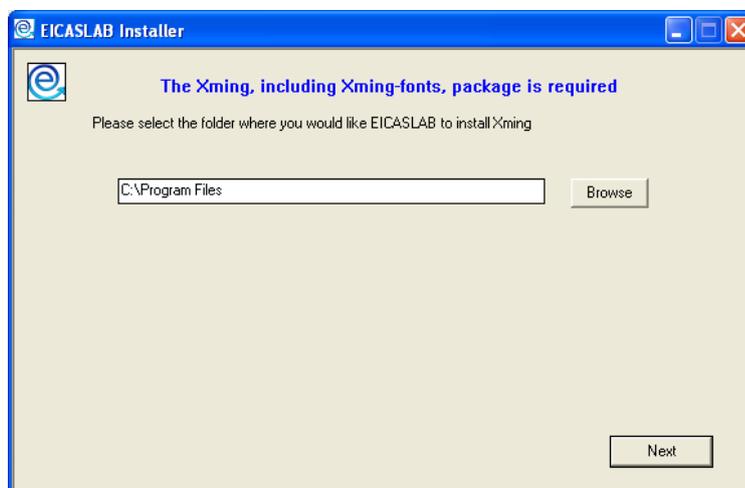


If the Xming package is not found in your system the EICASLAB installer shows you the following window where you can choose among the following items:

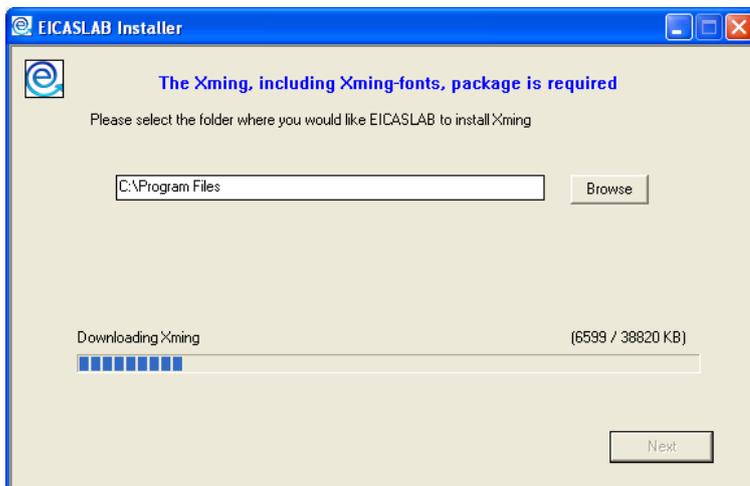
- **Xming is not installed in the PC. EICASLAB will automatically download and install the Xming package.** You can choose this item if you want EICASLAB to automatically install the Xming package.
- **Xming is already installed in the PC. Please provide the path of the Xming executable file.** You can choose this item if the Xming package is installed in your PC but EICASLAB is not able to detect the installation folder.
- **Xming is not installed in the PC. You can download the Xming package from the EICASLAB website <http://login.eicaslab.com/mingw/Xming.zip> and install it manually.** You can choose this item if you want to install manually the Xming package.



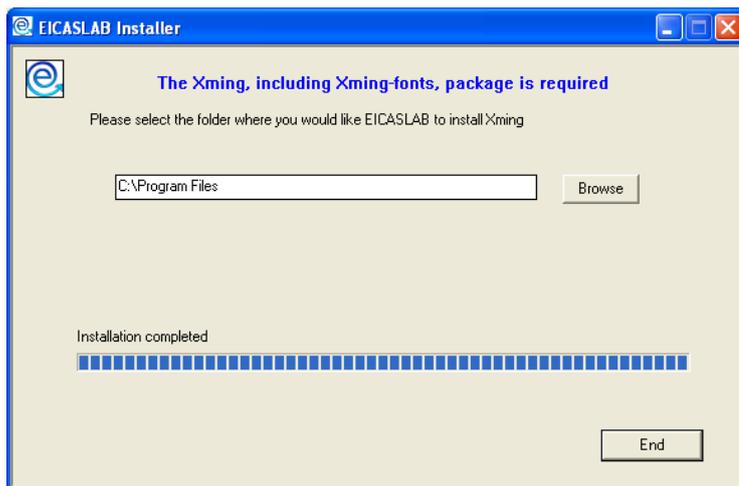
If you choose the first item the following window gets opened, where you can provide the folder where you want EICASLAB to install the Xming package.



By clicking the **Next** button the installation procedure starts and the following window gets opened.



When the installation of the MinGW package is completed you can click the **End** button in order to start the EICASLAB installation.



Note

When running Xming for the first time Windows firewall, if activated, could block the program. Please unblock it as shown in the following figure.



Starting EICASLAB DEMO Installer

EICASLAB DEMO Installer is the program able to install EICASLAB DEMO on your computer.



You can go on with the installation by using the **Next** button.

Note

In the following pictures we assume to work with a user called `myuser`, whose working directory is `/home/MyUser`.

License agreement

Before configuring your EICASLAB DEMO installation you have to read and accept all the terms of the EICASLAB DEMO license.

Enable the **I Accept** toggle button to go to the next step.



The EICASLAB License Agreement window will display an image that will ask to approve also third parties software licences. In fact EICASLAB relies on few applications and libraries provided by third parties under their own licence agreement. The licence text related to such applications and libraries are available by clicking the "View". By selecting "I accept" the user declares to agree with Terms and Conditions both of EICASLAB licence and of all the applications and libraries provided by third parties.

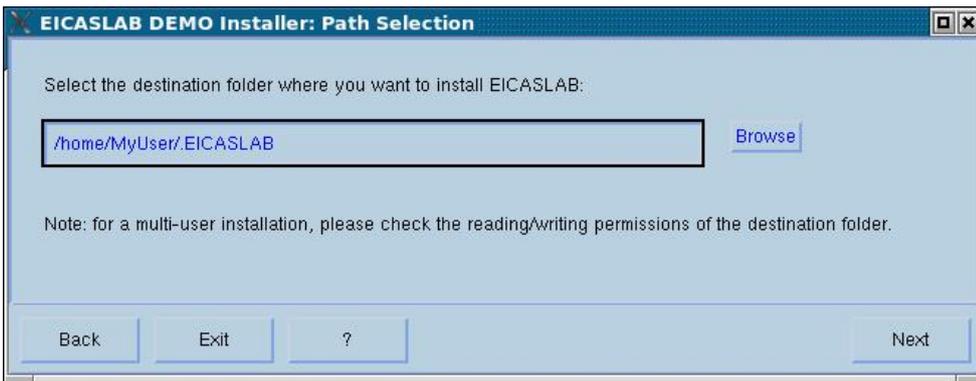
Path Selection

The next step regards the destination folder where you want to install EICASLAB DEMO.

As a default EICASLAB DEMO Installer creates in your home directory a new folder called `.EICASLAB` whose aim is to contain all the settings related to each EICASLAB release installed on your computer.

If you maintain the default, EICASLAB DEMO 1.0 will be installed in the folder:

`.EICASLAB/EICASLABDEMO1.0`

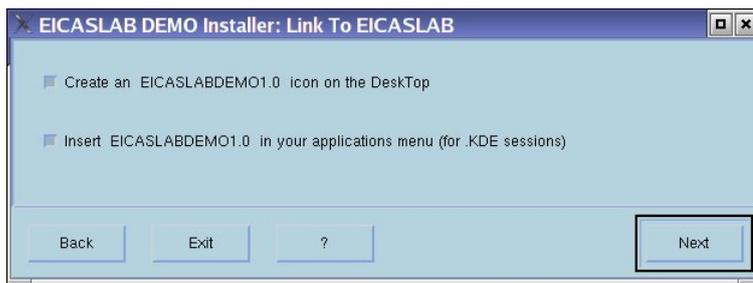


Link to EICASLAB

EICASLAB DEMO Installer allows you to quickly define links to EICASLAB DEMO.

It is possible to create an icon on the Desktop and a shortcut in the application menu for KDE sessions.

You can define also your working directory, that will be the default directory EICASLAB DEMO will use for opening user projects.



End of installation

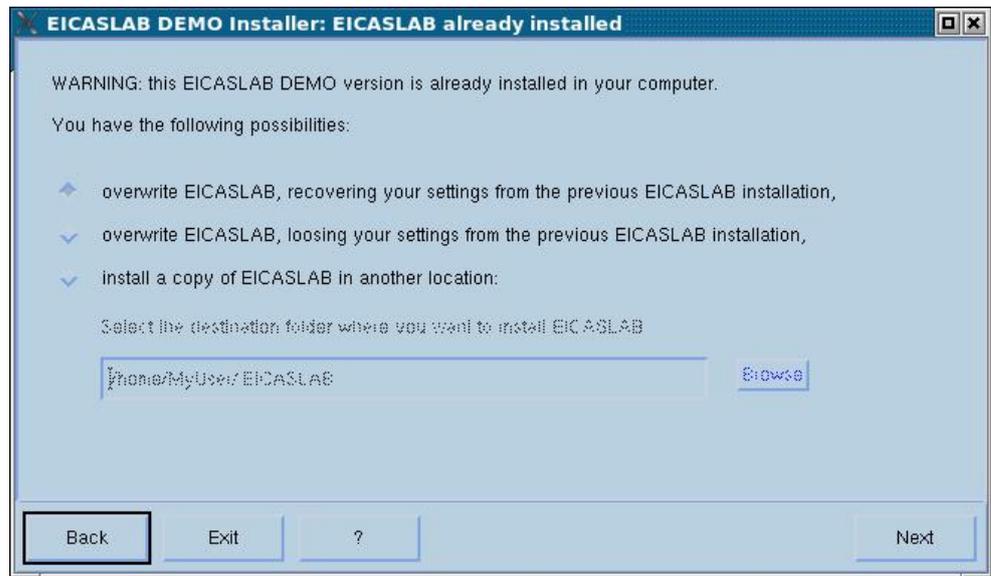
Once you have defined all settings, you have just to start the final installation step.



EICASLAB DEMO is now ready to be used.

Previous EICASLAB DEMO release

EICASLAB DEMO Installer is able to recognize the presence of previous installations of the same EICASLAB DEMO release : in such a case, after the license agreement, instead of the “*EICASLAB Installer: Path Selection*” window, the following window gets opened:



and you have the following possibilities:

- *overwrite EICASLAB, recovering your EICASLAB settings from the previous EICASLAB installation:* all your user settings will be recovered,
- *overwrite EICASLAB, losing your EICASLAB settings from the previous EICASLAB installation:* the old EICASLAB DEMO will be removed and replaced by a new one,
- *Install a copy of EICASLAB in another location:* If you install a copy of EICASLAB DEMO in a new location, the previous EICASLAB DEMO installation will be maintained, and the new one will be created. You will have two distinct and complete EICASLAB DEMO.

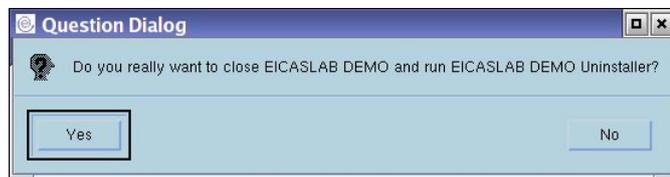
Uninstallation procedure

EICASLAB DEMO has a powerful uninstallation procedure that has to be launched by means of facilities available on the *Help* menu of MASTER.

You have to select the *Uninstall EICASLAB* item.

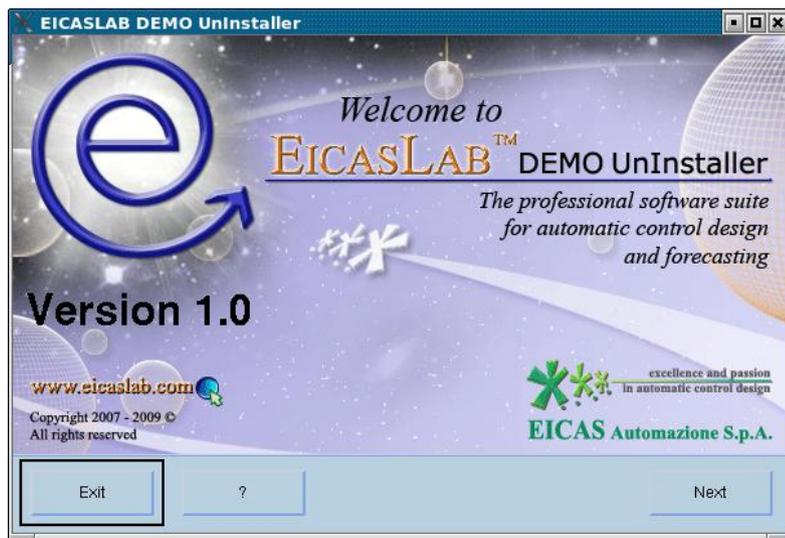


EICASLAB DEMO UnInstaller will start after closing EICASLAB DEMO.



Starting EICASLAB DEMO UnInstaller

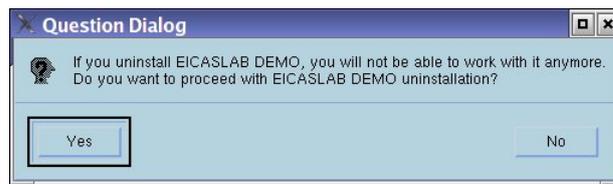
EICASLAB DEMO UnInstaller is the program able to uninstall EICASLAB DEMO on your computer.



You can go on with the uninstallation by using the **Next** button.

End of uninstallation

The uninstallation procedure has effect only after confirming your decision again.



EICASLAB DEMO has now been uninstalled.



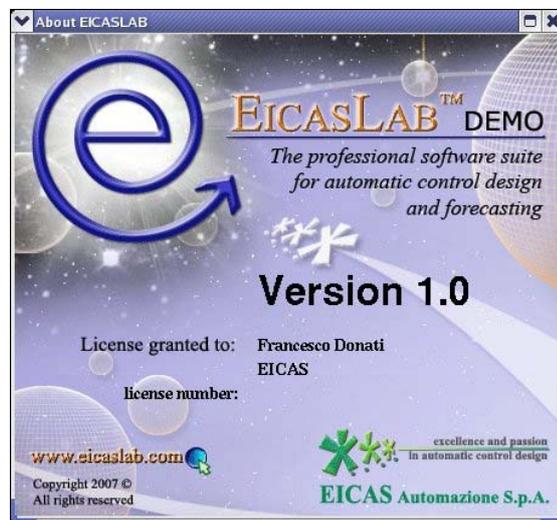
Running EICASLAB DEMO

EICASLAB DEMO for Linux can be launched by using all the quick links selected during the installation process, as:

- icons on the Desktop;
- shortcuts on the operative system application menu;
- symbolic links;
- alias.

Moreover you can launch EICASLAB DEMO directly by running the executable file EICASLAB. From a terminal you have to type the command:

```
/"Installation path"/EICASLABDEMO1.0/EICASLAB
```



EICASLAB DEMO for Windows can be launched by using the icon installed on the Desktop or the shortcut installed in the start menu.

DEMO projects

EICASLAB DEMO is supplied together with a set of test cases implementing the most common control algorithm techniques, applied to a “rotating table” single axis control application.

More information about the technical characteristics of each example is given in the document: `EICASLABDEMOTestcases.pdf`.

Project families

DEMO projects are subdivided into four main families according to the implemented control algorithms:

- *PID*: with a PID control algorithm;
- *Classical Model Based*: with a control algorithm based on state observer and state control;
- *EICAS Model Based*: developed by following the EICAS control design methodology (www.eicas.it);
- *User*: without any predefined control algorithm. Here you can develop your own control architecture.

The projects belonging to the first three families are locked: you can analyse the structure of the system, run the simulation and analyse the results, but you cannot make any modification.

On the contrary *User* project has been developed in order to allow you to build your own control algorithm and, as a consequence, to perform comparisons with the other predefined test cases.

Control architectures

For each control algorithm implemented, different control architectures have been considered:

- the control loop is closed by using the measurements taken on the motors, that is before the mechanical gear;
- the control loop is closed by using the measurements taken on the rotating load, that is after the mechanical gear.

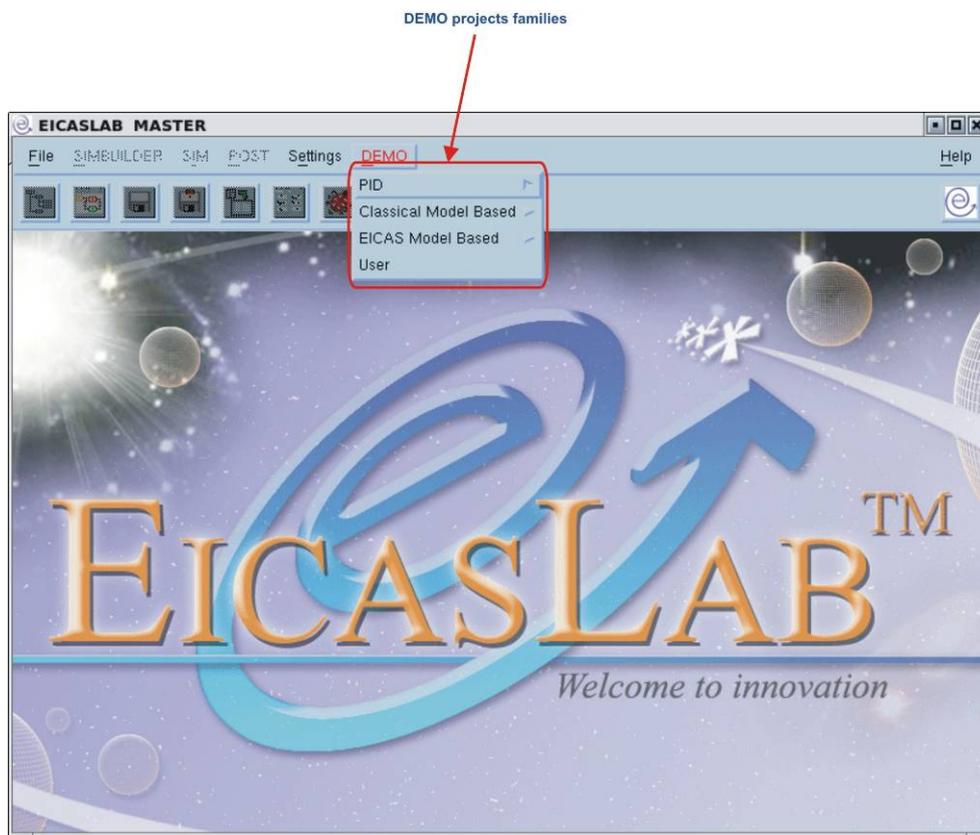
Moreover for the *EICAS Model Based* family also a hierarchical architecture has been implemented: the low level control loop is closed by using the measurements taken on the motors, the upper level control loop is closed by using the measurements taken on the rotating load.

As reported in the previous paragraph the *User* project does not contain any predefined control algorithm; in order to make you able to build the control architecture you prefer, both measurements taken on the motors and on the rotating load are available.

EICASLAB DEMO provides one project for every control architecture considered for each DEMO family.

Family	Control architecture – projects
PID	Motor measurement
	Table measurement
Classical Model Based	Motor measurement
	Table measurement
EICAS Model Based	Motor measurement
	Table measurement
	Hierarchical
User	Not predefined

They can be opened by using the *DEMO* menu of the MASTER window.



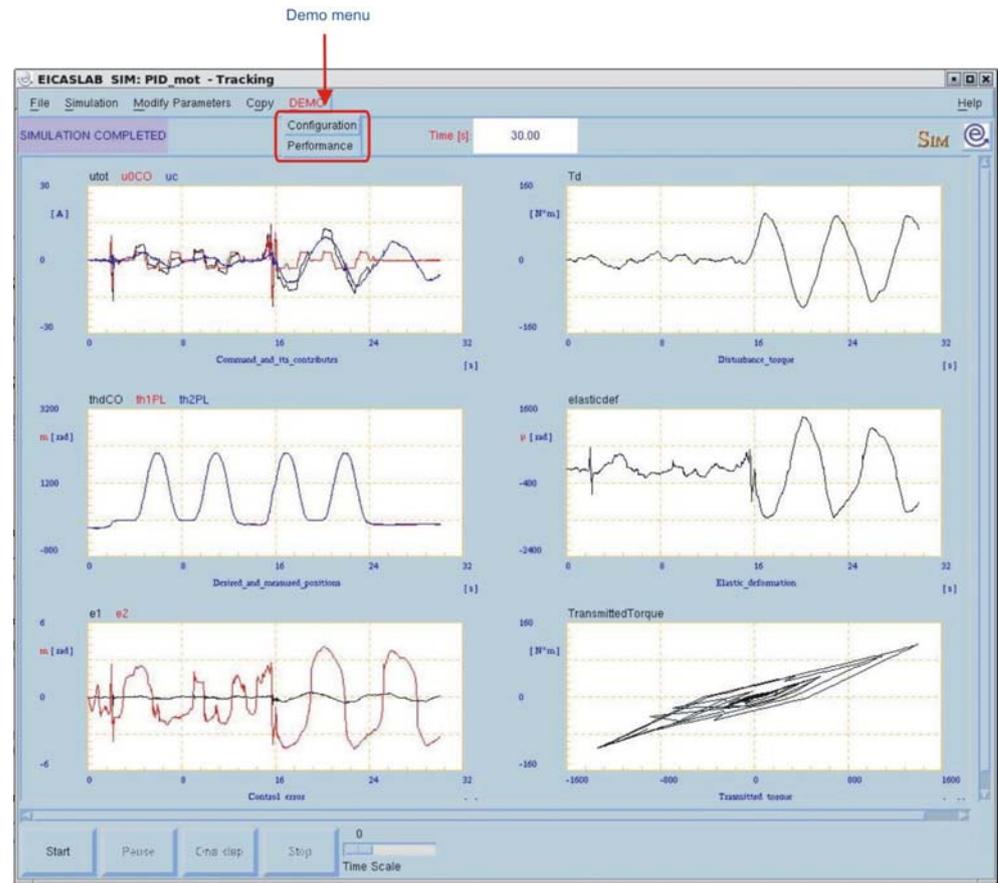
Operating modes

EICASLAB DEMO allows to analyse DEMO projects results in two different operating modes:

- *point to point* mode: the control system is required to rotate the load (of the rotating table) from the current angular position to the commanded angular position; the rotation must be carried out in the minimum time; once the commanded position has been obtained, it must be maintained until a new rotation command is received;

- *tracking* mode: the control system has to track reference trajectories.

Once chosen the DEMO project, the operating mode can be selected in the simulation phase, by using the *Configuration* item in the *DEMO* menu of the SIM window.



Here a table reporting all the possible combinations of families, control architectures and operating modes is available.

Family	Control architecture	Operating mode
PID	Motor measurement	Point to point Tracking
	Table measurement	Point to point Tracking
Classical Model Based	Motor measurement	Point to point Tracking
	Table measurement	Point to point Tracking
EICAS Model Based	Motor measurement	Point to point Tracking
	Table measurement	Point to point

		Tracking
	Hierarchical	Point to point Tracking
User	Not predefined	Point to point Tracking

Overview of EICASLAB

EICASLAB has been conceived and developed as a highly professional software suite supporting designers of automatic control system architectures.

EICASLAB supports the automation of industrial processes through powerful tools for modelling plants, designing and testing embedded control systems.

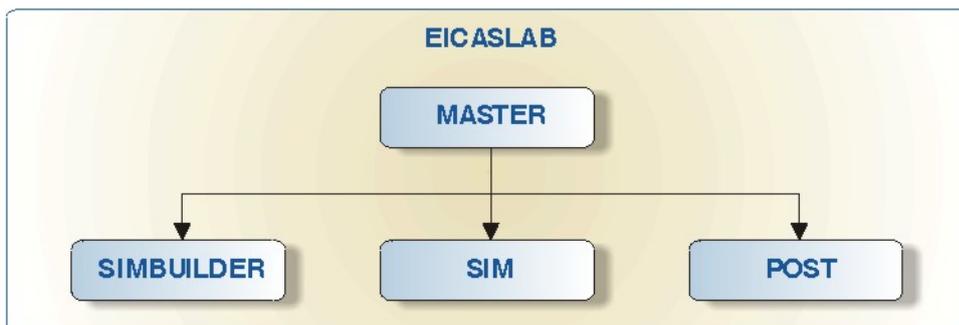
EICASLAB assists all phases of the design process of the control strategy: from system concept to finalised control structure.

The use of the EICASLAB envisages three different phases of work:

- automatic control and simulated environment *programming*;
- control trials *performing* in the above simulated environment;
- *post-processing* and analysis of the results obtained in the simulated trials.

Automated Algorithms & Code Generation, Multiprocessor architecture and Rapid Prototyping are part of the overall capabilities offered by the EICASLAB suite.

EICASLAB is composed of a main program, called MASTER, able to manage all the programming, performing and post-processing phases, by running three different tools: SIMBUILDER, SIM and POST.



Running projects

You can open a DEMO project by selecting it from the **DEMO** menu on the MASTER window.

EICASLAB DEMO allows you to:

- run SIMBUILDER, and analyse the system architecture. In the *User* project you can develop your control algorithm in *C mode* or in *Graphical mode*;
- compile the project, step necessary only for *User* project;
- run SIM, and start the simulation;
- run POST, and analyse the simulation data.

Here a short description of the functional buttons necessary to open SIMBUILDER, SIM and POST is presented.

Icon	Name	Description
	Run SIMBUILDER	Run SIMBUILDER, open the System Layout
	Save	Save your project, only for the <i>User</i> project
	Compile	Compile the project and generate the SIM, only for the <i>User</i> project
	Run SIM	Run SIM, performing your simulation
	Debug SIM	Make SIM run by means of a debugger
	Run POST	Run POST, in order to analyse your simulation results
	Quit Project	Close the project
	Exit	Close EICASLAB
	EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com

MASTER window menu

The top section of the MASTER window contains menus and related buttons able to drive you in the EICASLAB world.



File menu

Item	Description
Save	Save your project (only for <i>User</i> project)
Quit Project	Close the project
Exit	Close EICASLAB DEMO

SIMBUILDER menu

Item	Description
Run SIMBUILDER	Run SIMBUILDER, open the <i>System Layout</i>

SIM menu

Item	Description
Compile	Compile the project and generate the SIM, only for the <i>User</i> project.
Regenerate Makefile	Regenerate the Makefile used for the compilation (only for <i>User</i> project).
Compile options	Allow to modify the compile options and allow to link to pre-existing libraries (only for <i>User</i> project).
Run SIM	Run SIM, performing your simulation.
Debug SIM	Run the simulation by means of a debugger (only for <i>User</i> project).
Remove objects and SIM	Delete the objects generated during the compilation and the SIM executable file (only for <i>User</i> project).
Stop SIM	Stop and close the SIM: it is available only when the SIM is running.

POST menu

Item	Description
Run POST	Run POST, in order to analyse your simulation

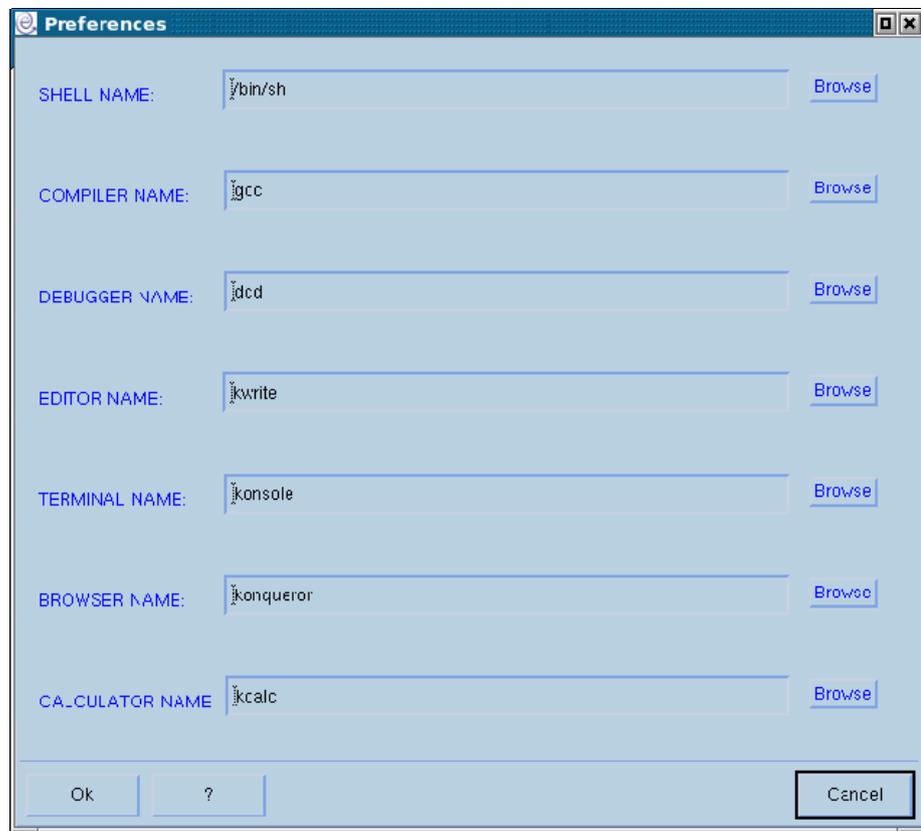
	results
--	---------

Settings menu

Item	Description
Preferences	Allow to modify your EICASLAB settings
Links and default Working Path	Allow to define new links to EICASLAB and to modify the default EICASLAB Working Path (default directory for the EICASLAB project selection)

“PREFERENCES” WINDOW

The *Settings* → *Preferences* menu opens the “*Preferences*” window:



This window allows you to define:

Item	Description
Shell name	Shell to be used by EICASLAB
Compiler name	Compiler to be used for the compilation process
Debugger name	Debugger for running the simulation in debugger mode
Editor name	Editor used to open the EICASLAB files

Terminal name	Terminal to be used as command line
Browser name	Browser necessary for managing files and directories, and for displaying the HTML documentation
Calculator name	Calculator to be used

DEMO menu

Item	Description
PID	Allow to select a project belonging to the <i>PID</i> family
Classical Model Based	Allow to select a project belonging to the <i>Classical Model Based</i> family
EICAS Model Based	Allow to select a project belonging to the <i>EICAS Model Based</i> family
User	Allow to develop the <i>User</i> project

Help menu

Item	Description
User manual	Run the EICASLAB DEMO User Manual
Short guide – EICASLAB in 6 steps	Run the EICASLABDEMO in 6 steps Manual
Technical note on DEMO Test Cases	Open the technical note on DEMO Test Cases
EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com
About EICASLAB	Display some information about version and licence of EICASLAB
On line support	Run the EICASLAB web site at the support page
Uninstall EICASLAB	Run uninstallation procedure.

SIMBUILDER

SIMBUILDER is the tool that allows the fine model construction and the control algorithm design. It appears as a white paper in which you can insert the blocks necessary to represent your system.

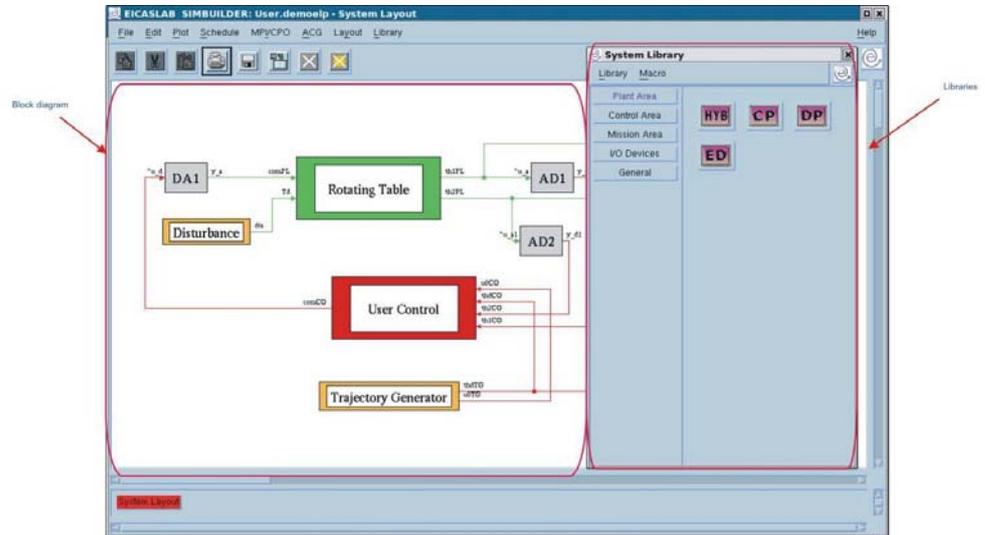
Each block can be programmed in several ways: it is possible to build a part of the project in a graphical way (*Graphical mode*) and to build another part by using a code representation (*C mode*); this choice is called *programming mode*.

The layout concept

SIMBUILDER is focused on the concept of **layout**: each block of the project can appear as a white paper with a related blocks library.

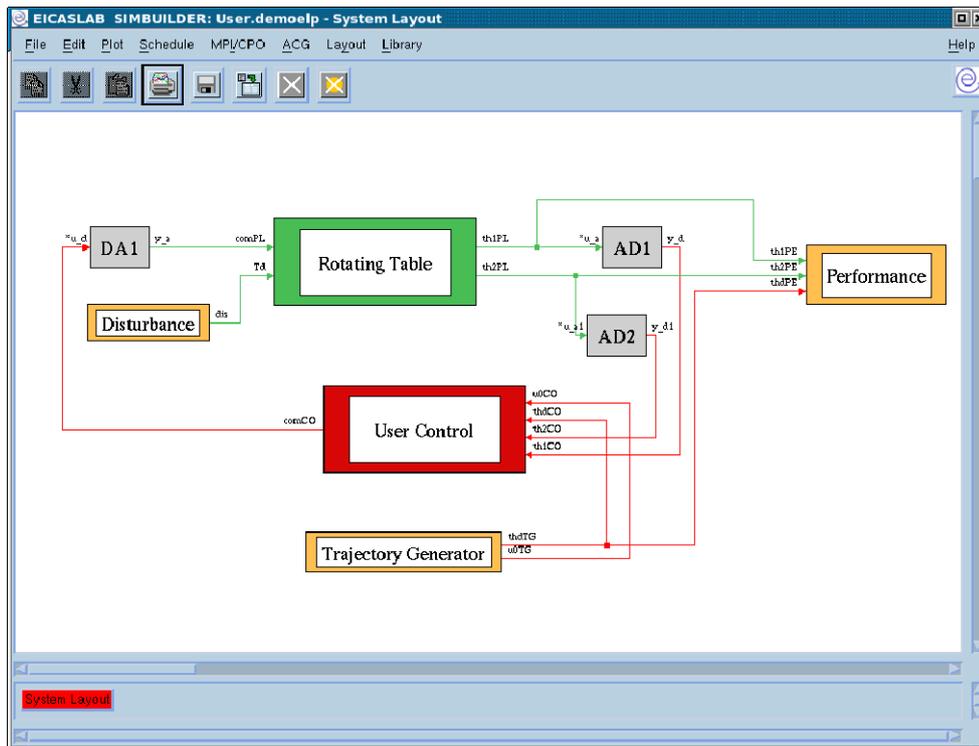
The layouts are organized in a **hierarchical** way: the layout of a specific block can contain other blocks represented by other layouts:

Each layout is related to a particular and **oriented library**: it contains only the blocks that can be inserted in that layout.



System layout

System Layout is the highest level of the project representation and it appears when you runs SIMBUILDER; this environment is realised in order to contain the three most important blocks of the system: the Plant Area, the Control Area, and the Mission Area.



The Plant Area is used to program the model of the plant. Continuous, discrete or hybrid plants can be simulated.

The Control Area is used to design the plant controller, which is conceived as a potential multi-processor system, where each processor may perform multi control functions.

The Mission Area is used to provide both the user references for the control and the disturbances that operate on the plant.

Moreover, in order to allow a complete simulation closer to the real world, it is also possible to insert in the System Layout, from the appropriate library, A/D and D/A converters. They can be used to simulate the real behaviour of the data transfer from the plant to the control and vice versa.

Programming mode

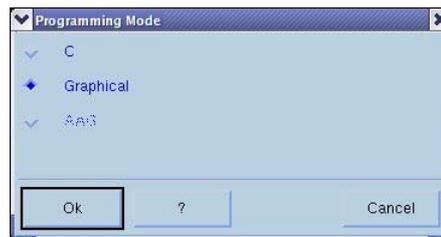
The structure of DEMO projects is already available. *PID*, *Classical Model Based* and *EICAS Model Based* projects are locked, you can only analyse the structure of the system, but you cannot make any modification; if you are interested in them, you can go directly to chapter “SIM”.

Only in the *User* project you can develop your own Control. In this case you can choose to program the Control Area by using different *programming modes*:

Graphical mode allows you to build your block by using a block diagram representation. The scheme will be composed by elementary blocks, available in the library window, connected by means of wired links.

C mode, on the contrary, gives you the possibility to build a block by inserting your C code in a complete structure of *.c* and *.h* files whose templates are provided and generated by EICASLAB.

This selection is available in the popup menu of the Control block at the **Block Programming** item.



SIMBUILDER window menu

The top section of the SIMBUILDER window contains menus and related buttons able to drive you in building your project.



In the following only items available in EICASLAB DEMO will be considered.

File menu

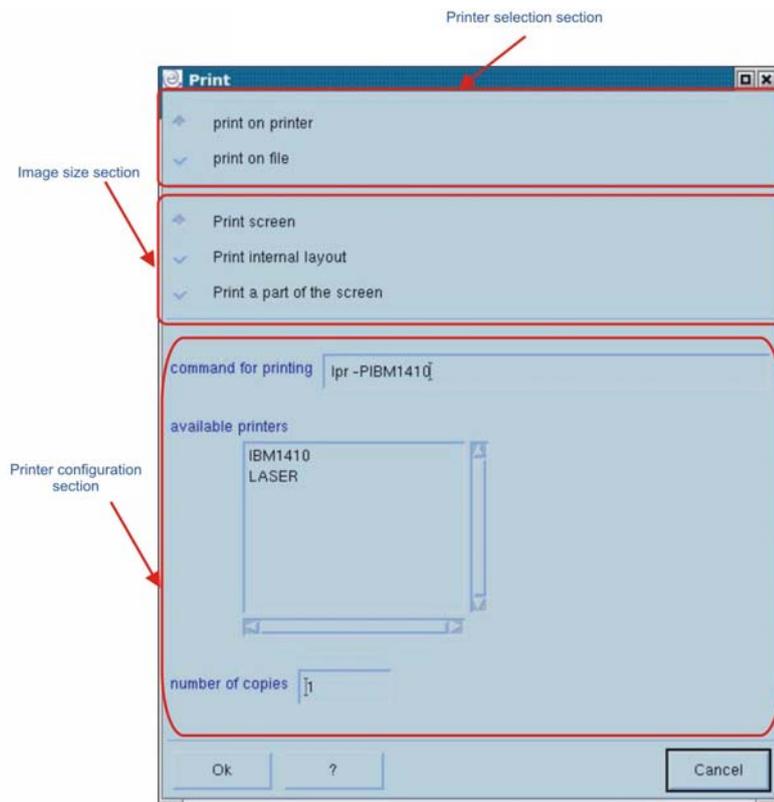
Item	Description
Save	Save the project (only for the <i>User</i> project)
Compile	Compile the project and generate the SIM, only for the <i>User</i> project
Print	Print the screen on a printer or on a file
Print Options	Pre-selection of print options (print all the layout or part of it; print on printer or on file)
Hide	Close the current layout window
Hide All	Close SIMBUILDER
Exit from the project	Close the project (and close SIMBUILDER)
Exit from EICASLAB	Close EICASLAB

PRINT

Print item allows you to print your layout window (or a part of it) on a printer or on a file. This facility is based on the import Linux command.

The “Print” window is subdivided in three main sections:

- *Printer selection section*
- *Image size section*
- *Printer configuration section*



Printer selection section allows you to select whether you want to print on a printer or on a file.

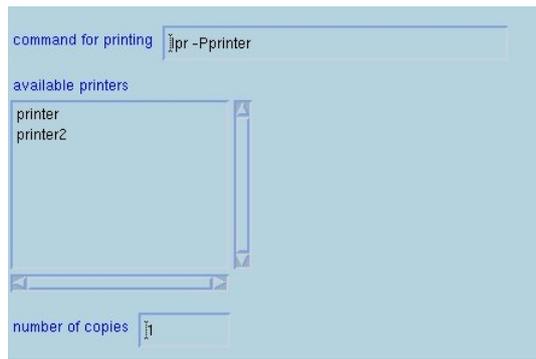


Image size section allows you to select what you want to print:

- *Print screen* item will print what you can see in the screen
- *Print a part of the screen* will print only a selection of the screen. The selection can be done by clicking and moving the mouse around the part of the layout you want to print.

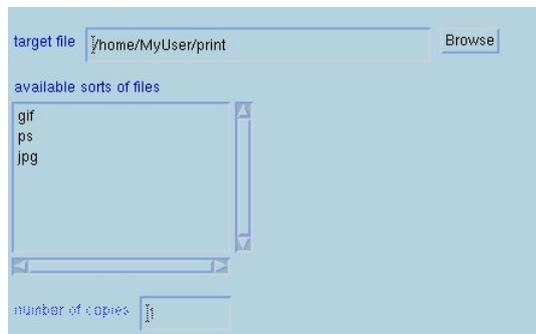
Printer configuration section reports all the options related to the printer you selected. If you choose to print on a printer, you can select:

- the Linux command to use
- the printer
- the number of copies



If you choose to print on a file, you can select:

- the name of the file
- the type (gif, ps or jpg)



The result of your print is an exact copy of what you see on the screen.

The file where you printed your layout (if you choose to print on file) is reported in the directory from which you run EICASLAB.

Edit menu

Item	Description
Refresh	Refresh the layout (F5 keyboard button)
Clear Area	Delete all the blocks in the layout (only for the <i>User</i> project).
Select All	Select all the blocks in the layout (only for the <i>User</i> project).
Deselect All	Deselect all the blocks in the layout (only for the <i>User</i> project).
Scale	Allow to change the dimension of the layout. The scale factor is between 40% and 200% (only for the <i>User</i> project).
Copy	Copy the selected blocks (only for the <i>User</i> project).

Cut	Cut the selected blocks (only for the <i>User</i> project).
Delete	Delete the selected blocks (only for the <i>User</i> project).
Paste	Paste the blocks previously copied or cut (only for the <i>User</i> project).
Rotate	Rotate the selected block (only for the <i>User</i> project).
Global Layout Data	Open the “ <i>Global Layout Data</i> ” window, where you can manage all the parameters of all the blocks present in the layout.
Block Setting	Allow to modify the Block Setting properties of the selected block, such as: Data, Name, Dimension, Structure, Input/Output. (only for the <i>User</i> project).
Block Programming	Collect all the functionalities concerning the block development (only for the <i>User</i> project).

Plot menu

Item	Description
SIM Plot Global	Display the current settings for the SIM Plot (only for the <i>User</i> project).
POST Plot Global	Display the list of the blocks that can have variables selected for POST plot: you can ‘deselect’ any of these blocks: in that case the POST file of the deselected blocks are not written during the simulation, and then the related variables, selected for POST, are not recorded.
Show variable in POST/SIM Plot	Put on evidence the blocks of the layout having variables selected for SIM Plot S and POST Plot P .
POST Plot Options	Set inputs/outputs/states of all the blocks in the layout for POST Plot (only for the <i>User</i> project).
SIM Plot Block	Set variables for SIM Plot (only for the <i>User</i> project).
POST Plot Block	Set variables for POST Plot (only for the <i>User</i> project).

Schedule menu

Item	Description
Plant Activities Schedule	Schedule the activities of Discrete Plant, Experimental Data, Elementary Mission and Mission. In EICASLAB DEMO you cannot change the Plant Activities Schedule.
Control Activities Schedule	Schedule the activities of Controls and the Converters. In EICASLAB DEMO you can only change the

	scheduling of the user control of the user project; you cannot change other schedulings. See paragraph “Scheduling management” (page 33) for more details.
--	--

MPI/CPO menu

Not enabled in EICASLAB DEMO.

ACG menu

Not enabled in EICASLAB DEMO.

Workspace menu

Not enabled in EICASLAB DEMO.

Layout menu

Item	Description
“depends on the available layouts”	Allow to move among all the available layouts

Library menu

Item	Description
Hide/Show	Hide and show the <i>library window</i> of the layout

Help menu

Item	Description
User Manual	Run the EICASLAB DEMO User Manual
EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com
About EICASLAB	Display some information about version and licence of EICASLAB DEMO

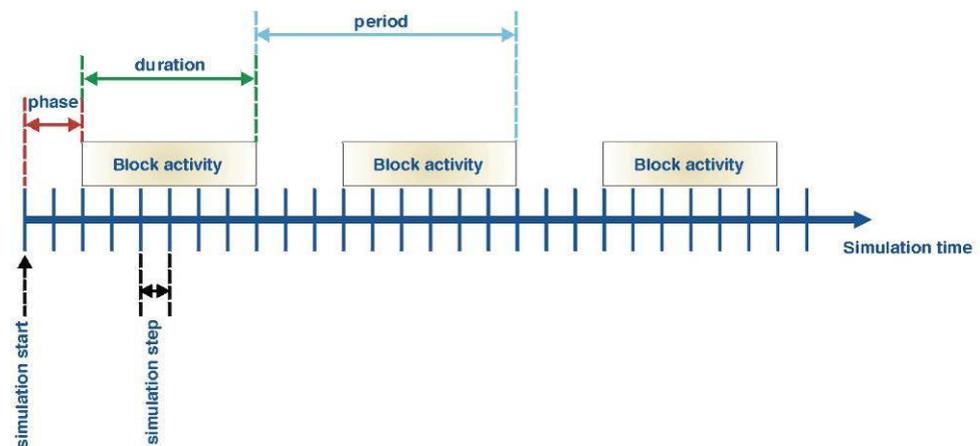
Scheduling management

EICASLAB simulations are based on a time scheduling algorithm that allows you to manage microprocessor control architectures and to perform simulations very close to the real behaviour of your system.

The scheduling concept regards the temporal management of the blocks representing your system: you can set the instant time at which every block must be executed.

You can view the scheduling of the blocks of the EICASLAB DEMO by using the *Schedule* → *Plant Activities Schedule* and the *Schedule* → *Control Activities Schedule* menus of the SIMBUILDER: they open the *Schedule* window indicating the names of the scheduled blocks, and their three schedule parameters:

- *duration*, defines the execution time of your block function;
- *period*, is the interval of time between two successive executions;
- *phase*, determines the time, from the beginning of the simulation, at which every block function starts; in this way it determines the phase displacement among your block functions.

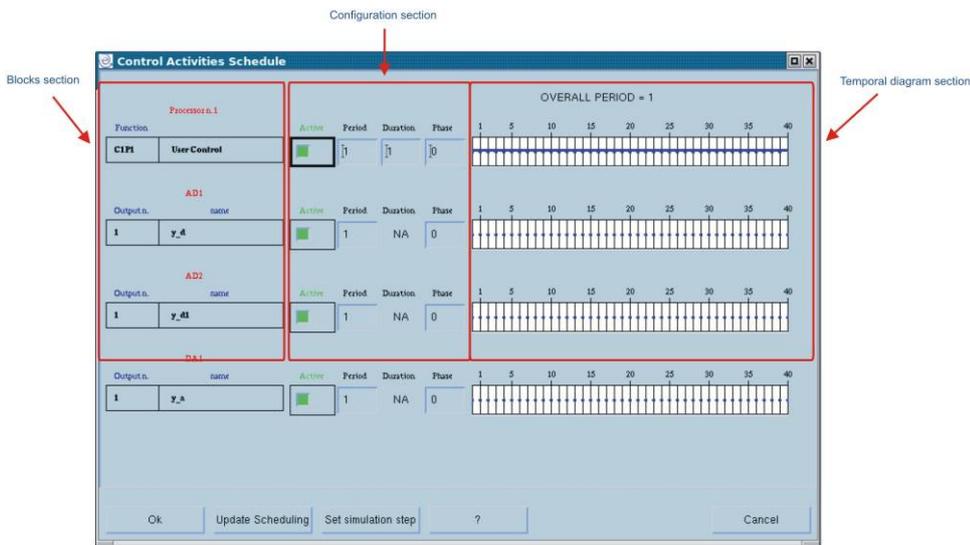


Each parameter is expressed in terms of simulation steps.

The period multiplied by the simulation step gives the sampling time of block function.

Schedule window

The “*Control Activities Schedule*” window of the user project is:



The “Control Activities Schedule” window is composed by three main sections:

- *Blocks section*;
- *Configuration section*;
- *Temporal diagram section*.

Block section reports the list of the blocks whose schedule is represented in the window.

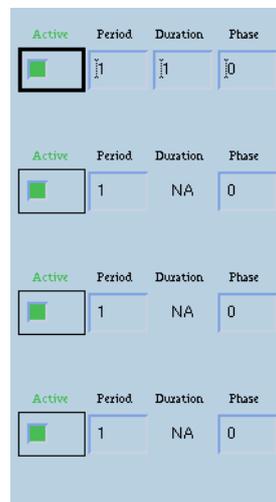
Configuration section allows you to configure the scheduling activities for each block reported in the *Blocks section*.

You can:

- activate each block, so the block will be simulated with its scheduling parameters;
- deactivate each block, so the block will not be simulated, and the outputs of the block will be set to zero during simulations.

You can define the scheduling parameters by using the *Period*, *Phase* and *Duration* items (the unit of measurement is the simulation step).

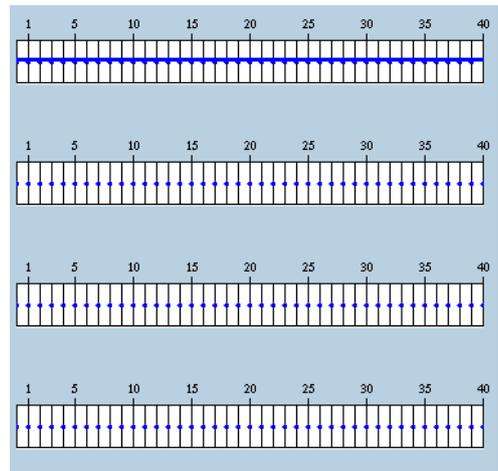
You can define the scheduling parameters by using the *Period*, *Phase* and *Duration* items (the unit of measurement is the simulation step).



Note

It is not possible to set the duration of the converters because they are considered instantaneous (their duration is considered very much lower than the simulation step). Their duration is indicated 'NA' which means 'not applicable'.

Temporal diagram section reports a time diagram where you can visualize the temporal behaviour of your block function. You can refresh the *temporal diagram section* by using the **Update Scheduling** button.



Buttons

The top section of the SIMBUILDER window contains menus and related buttons able to drive you in building your project.



Icon	Name	Description
	Copy	Copy the selected blocks (Ctrl+C keyboard buttons)
	Cut	Cut the selected blocks (Ctrl+X keyboard buttons)
	Paste	Paste the blocks before copied or cut (Ctrl+V keyboard buttons)
	Print	Print the layout on a printer or on a file

	Save	Save the project
	Compile	Compile the project and generate the SIM, only for the <i>User</i> project
	Hide	Close the current layout window
	Hide All	Close the SIMBUILDER (all the layout windows)
	EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com

Popup menu

You can build your project by developing and managing both blocks and layouts by means of suitable and complete popup menu.

Popup menu contains items present in the SIMBUILDER window menu; they are collected in sections depending on the operation they perform.

Block popup menu

You can enter the popup of a block by right clicking on it.

The popup are composed by the following sections:

Section	Description
Block Programming	Contain functions for developing and implementing your block
Block Setting	Contain functions for defining parameters and structure of your block
Plot Configuration	Allow to set SIM and POST plot options
Block Modification	Contain functions to copy, paste, rotate etc...
Help	Run the EICASLAB User Manual

Note

The sections and their items strictly depend on the block you select. Different blocks can have different items in the popup menu.

Block programming

Item	Description
Open Layout	Open the layout window for graphical blocks
Open File Manager	Open the File Manager window for blocks programmed in C mode
Open C Function	Open the C file related to <i>C Macros</i>
Programming Mode	Select the programming mode

Block setting

Item	Description
Block Setting	Contain functions for defining parameters and structure of your block
Export to 'user' Library	Just for <i>C macro</i> : export <i>C Macro</i> in a <i>user library</i>

BLOCK SETTING SUB-POPUP MENU

Item	Description
Name	Change the name of the block
Dimension	Change the Input/Output dimension of the block
Input/Output	Change the Input/Output setting of the block
Structure	Just for <i>C macro</i> : change the structure of the block, including input/output, name, etc...
Data	Change the parameters value of the block

Plot configuration

Item	Description
SIM Plot Block	Set variables for SIM Plot
POST Plot Block	Set variables for POST Plot

Block modification

Item	Description
Copy	Copy the selected block (Ctrl+C keyboard buttons)
Cut	Cut the selected block (Ctrl+X keyboard buttons)
Delete	Delete the selected block (Cancel keyboard button)
Paste	Paste the selected block (Ctrl+V keyboard buttons)
Rotate	Rotate the selected block (Ctrl+R keyboard buttons)

Rotate sub-popup menu

Item	Description
Put right -> left	Rotate the block in order to set inputs on the left and outputs on the right
Put left -> right	Rotate the block in order to set inputs on the right and outputs on the left
Put up -> down	Rotate the block in order to set inputs up and outputs down
Put down -> up	Rotate the block in order to set inputs down and outputs up
Rotate 90	Rotate the block clockwise with steps of 90 degrees

Help

Item	Description
Help	Run the EICASLAB DEMO User Manual

Layout popup menu

You can enter the popup of a layout by right clicking on it.

Item	Description
Paste	Paste the selected block (Ctrl+V keyboard buttons)
Refresh	Refresh the layout (F5 keyboard button)
Clear All	Delete all the blocks in the layout
Select All	Select all the blocks in the layout
Deselect All	Deselect all the blocks in the layout
Scale	Allow to change the dimension of the layout. The scale factor is between 40% and 200%

Navigating the project

You can navigate your project by using the SIMBUILDER menu (the *Layout* menu allows to move among layouts) or by using the mouse.

For EICASLAB it is suggested to use a three buttons mouse; here the description of the purposes of each button is reported.

Left button

One click to select blocks in the layouts, and to move them.

Double click to enter:

- the block layout (high-level, low-level and subsystem layouts);
- file manager (for blocks programmed in *C mode*);
- “*Block Setting: Data*” window for elementary blocks.

Central button

One click to enter:

- the block layout (high-level, low-level and subsystem layouts);
- file manager (for blocks programmed in *C mode*);
- “*Block Setting: Data*” window for elementary blocks.

Right button

One click for the popup menu of the selected block.

User project only

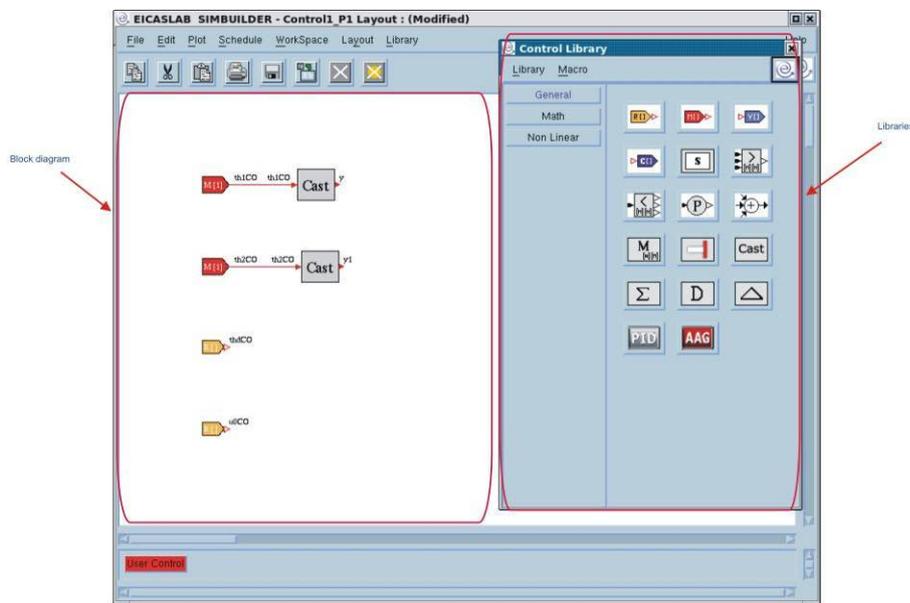
Graphical mode

Graphical mode works on libraries that are the core of the layout concept on which EICASLAB is based.

Libraries

Libraries are the core of the layout concept on which EICASLAB is based: each layout is equipped with a particular and oriented *library window*.

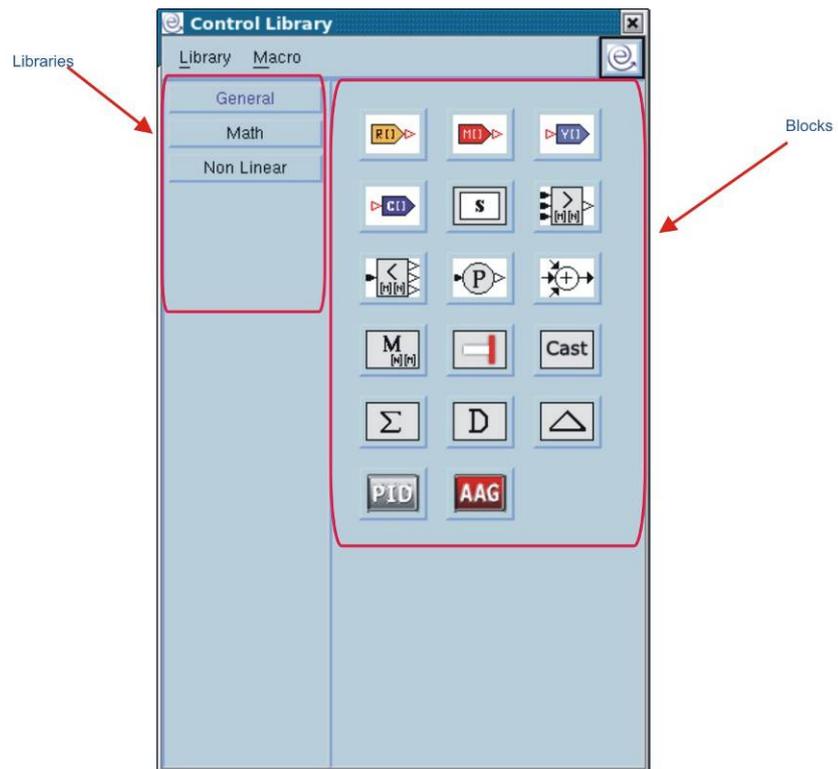
You can build your graphical representation by dragging & dropping the blocks available in the libraries.



Library window

The *library windows* have three sections:

- the top section of the *library window*, which contains menus:
 - the **Library** → **Hide** menu hides the *library window*, for viewing it again you can use the **Library** menu of the layout,
 - for other menus see paragraph “Macros” (page 44)
- the left section, which indicates the name of the available libraries of the layout related to the *library window*,
- the right section, which contains the *icons* representing the blocks contained in the selected library,



there are two kind of libraries:

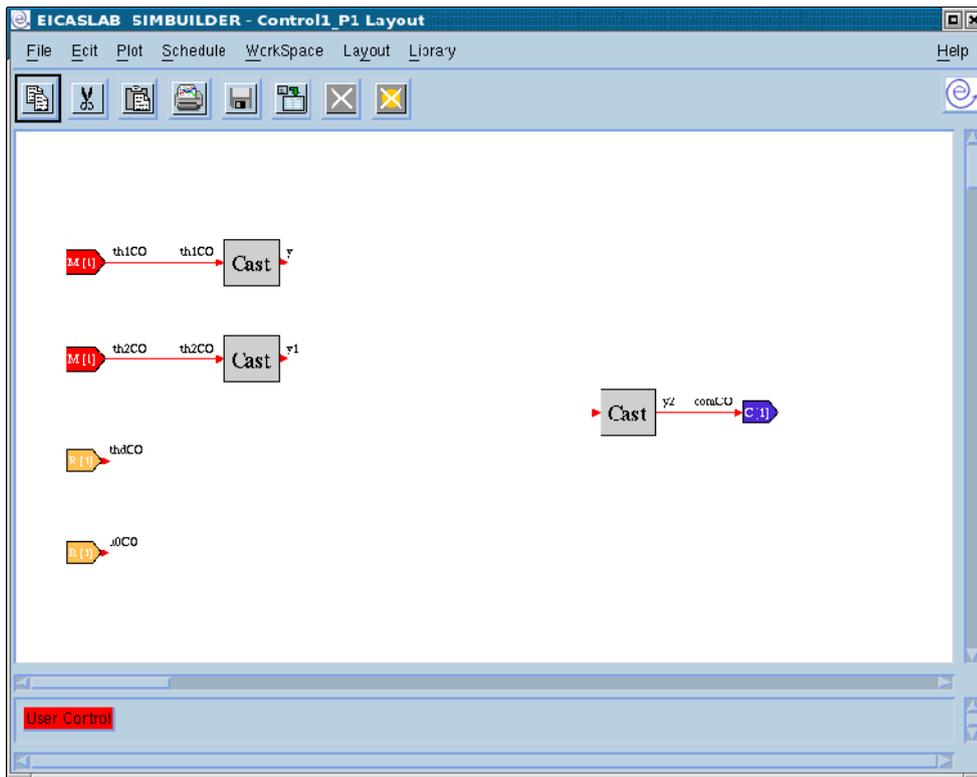
- *default libraries*: containing the blocks available as default for each specific layout,
- *user libraries*: containing the user blocks, named *macros* (see paragraph “User Library”, page 44).

You can select a specific library by clicking on the corresponding button available in the left section of the *library window*. After the selection, the button label of the selected library changes its colour from black to blue and the blocks associated with the library appear in the right part of the *library window*. In the above figure the *General default library* is selected

Default Libraries

You can build your Control (only for the *User* project) in a graphical representation by clicking & pasting the blocks available in the library.

A complete description of input/output variables related to the Control in the *User* project is reported in the Appendix “I/O Control variables for User project”.



EICASLAB organizes its libraries into families according to the block purposes: General, Math, Non-Linear, Local, User, System.

Control blocks

General

Block	Short description
Control Measure Input	Provide input measure signal (not enabled in EICASLAB DEMO)
Control Reference Input	Provide input reference signal (not enabled in EICASLAB DEMO)
Control Command Output	Provide output command signal (not enabled in EICASLAB DEMO)
Control Data Output	Provide output data signal (not enabled in EICASLAB DEMO)
Discrete Integrator	Perform discrete-time integration of a signal
Gain	Multiply scalar inputs by a specified scalar value
Sum	Output the sum of inputs
Matrix	Multiply vector inputs by a specified matrix value
Subsystem	Provide the Subsystem Layout
Mux	Combine several input scalar signals into a vector and output a vector signal
Demux	Extract and outputs the elements of a vector input signal
Delay	Provide an outputs signal one step delayed respect to the input signal
AAG	Implement an AAG algorithm (not enabled in EICASLAB DEMO)
Cast	Implement a data type cast conversion
Delta	Execute the difference between the input at the current time and its value at the previous time
PID	Implement a PID control algorithm
Terminator	Terminate an unconnected output signal

Math

Block	Short description
Absolute	Output the absolute value of the input
Product	Generate the element-wise product of block scalar inputs
Division	Generate the element-wise quotient of block scalar inputs

Non-Linear

Block	Short description
Min Sat	Limit the lower value of a signal
Max Sat	Limit the upper value of a signal
Double Sat	Limit the range of a signal

Macros

In EICASLAB you can create your own *user libraries*, in which you can insert your personal blocks, called *macros*. A *macro* is a block you can customise according to your needs and save for future use.

EICASLAB provides two kinds of *macros*:

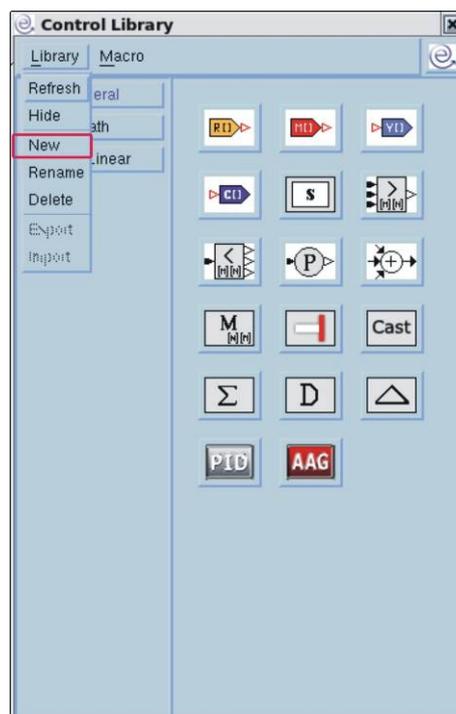
- Graphical Macro: you can build your block in *Graphical mode*, working on the *Graphical Macro* layout.
- C Macro: you can develop your block by using file templates available in ANSI C language.

Macros, in the EICASLAB DEMO, are available just for the control of the user demo project.

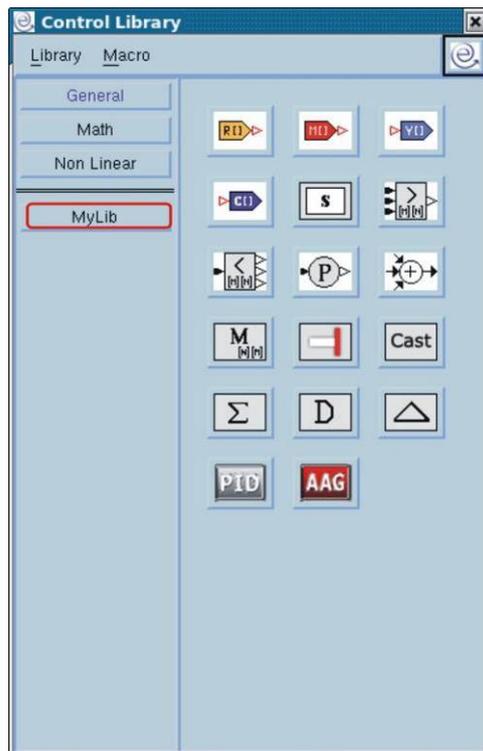
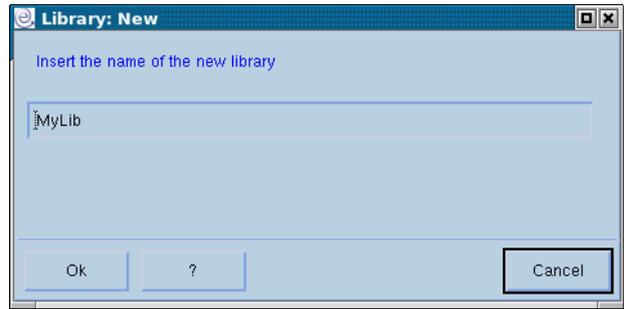
Every *macro* belongs to a *user library*.

User Library

Beyond the previously described *default libraries*, you can create your own libraries, by clicking on the menu *Library* → *New*:

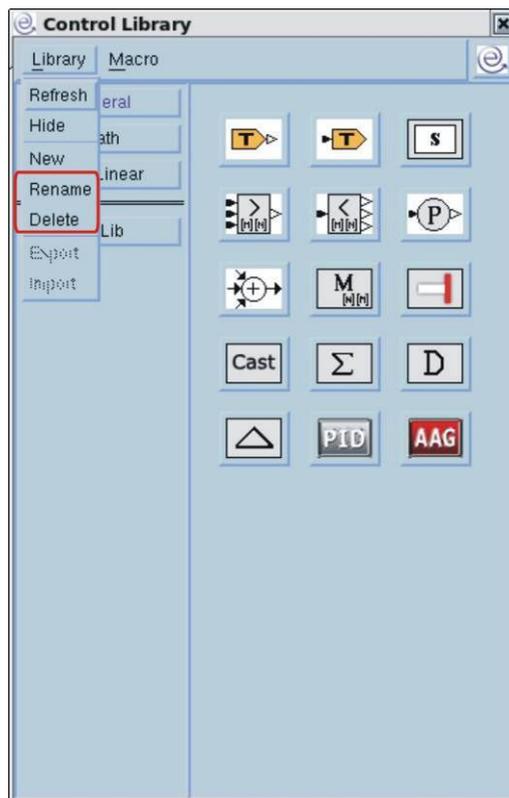


You then see the “*Library: New*” window in which a default name (*MyLib*) of the new *user library* is proposed, that you can change.



When you press the **Ok** button of the “*Library: New*” window, EICASLAB creates the new *user library* and a new button appears in the left section of the library *window*.

It is possible to rename or delete a *user library* by selecting it in the left section of the *library window* and by clicking on the menus **Library** → **Rename** or **Library** → **Delete**

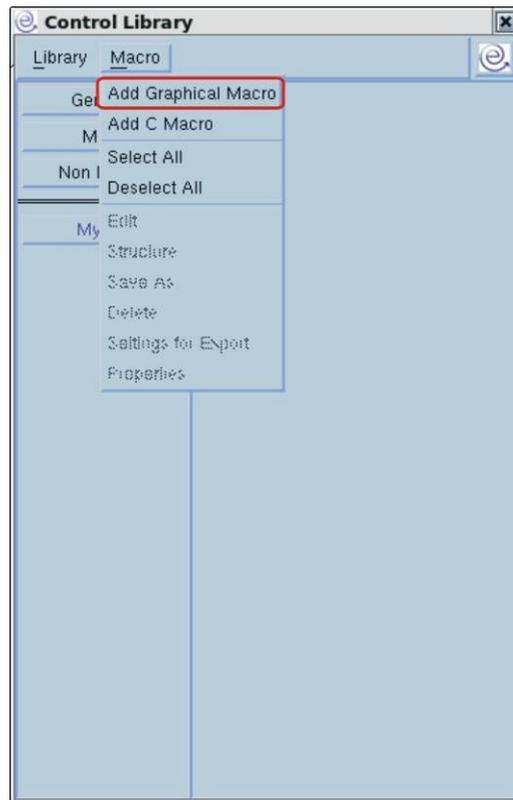


Graphical Macro

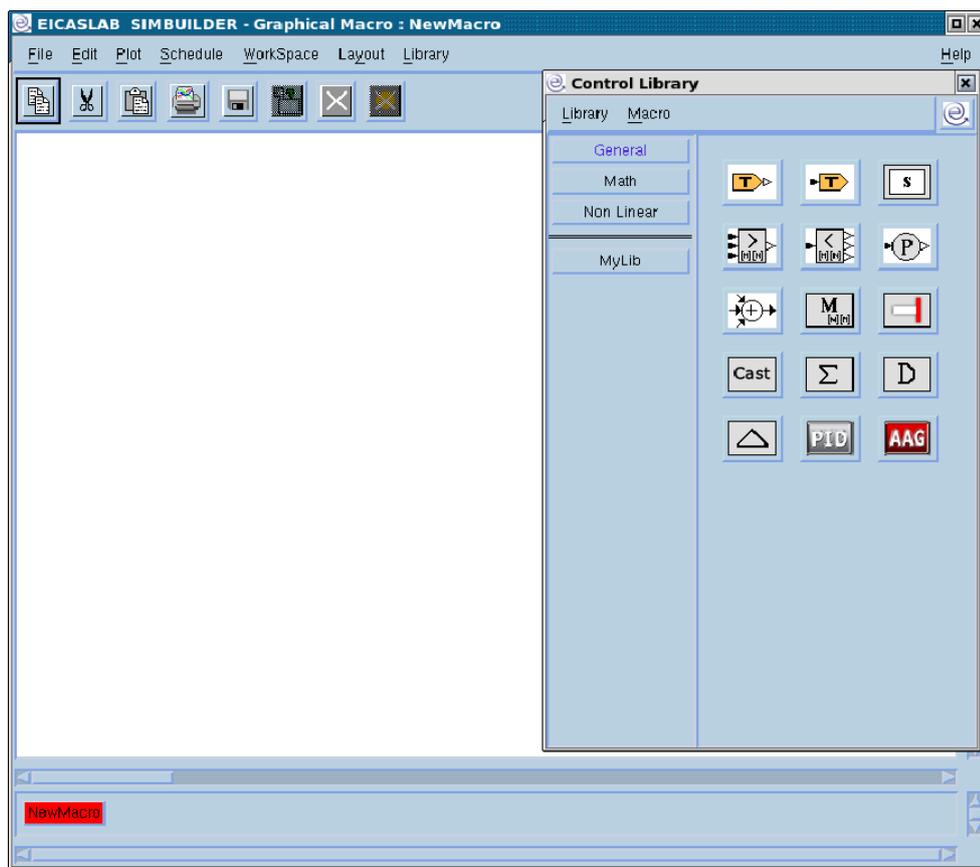
Creation of a Graphical macro

If you want to create a *Graphical Macro* in the Control Layout, after having inserted in your project a graphical Control block, you must:

1. open the Control Layout (by double clicking on the Control block or by selecting the **Open layout** item of its popup menu),
2. create in the *library window* of the Control Layout the *user library* that will contain your *Graphical Macro*,
3. select such a library and click on the menu **Macro** → **Add Graphical Macro** of the *library window*.



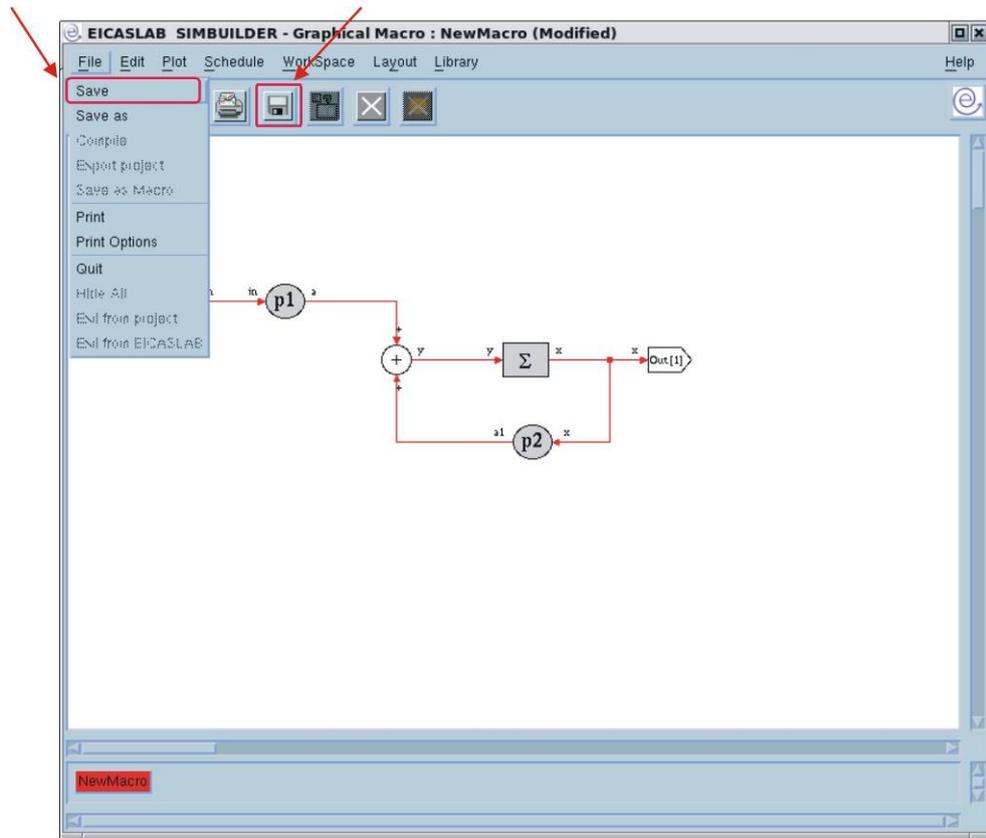
A new layout appears, named “*EICASLAB SIMBUILDER – Graphical Macro: NewMacro*”, having the same *library window* of the Control. You can then build your *macro* by editing such a layout.



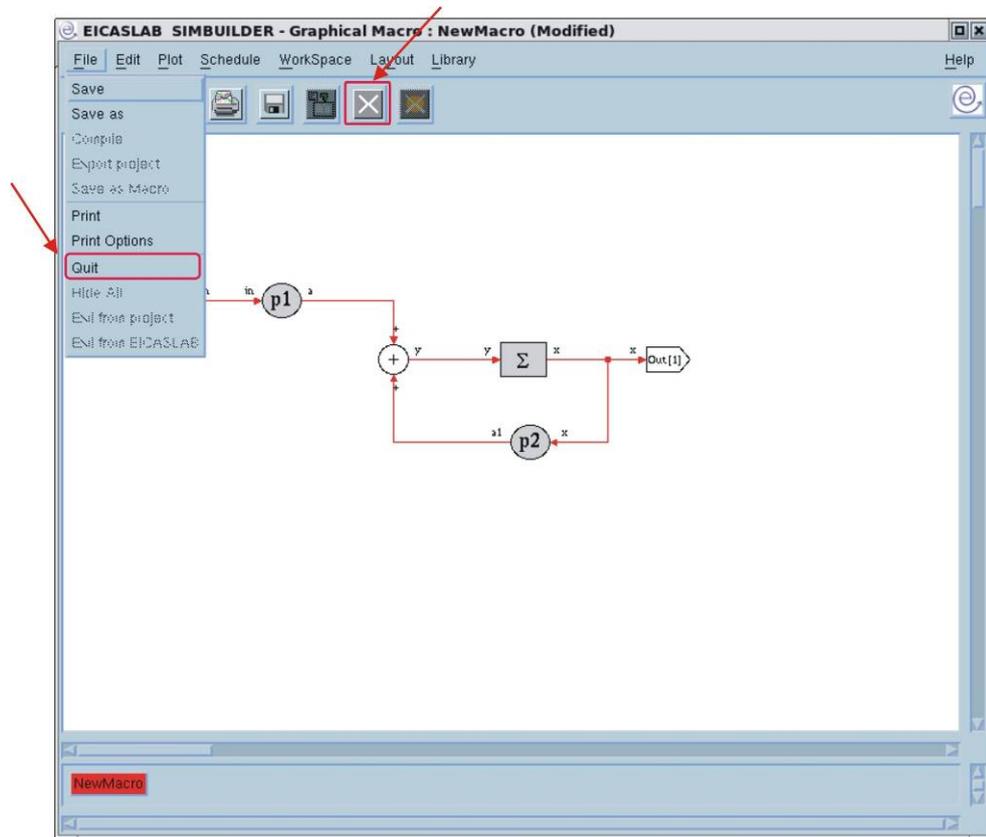
Note

The “EICASLAB SIMBUILDER – Graphical Macro: NewMacro” hides the Control Layout: you cannot edit the Control before you end the editing of your *macro* by closing the macro layout .

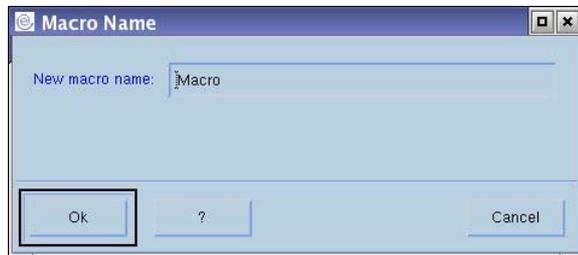
While you edit the *macro*, you can save it by means of the menu **File** → **Save** or by clicking on the dedicated push available among the *window* buttons.



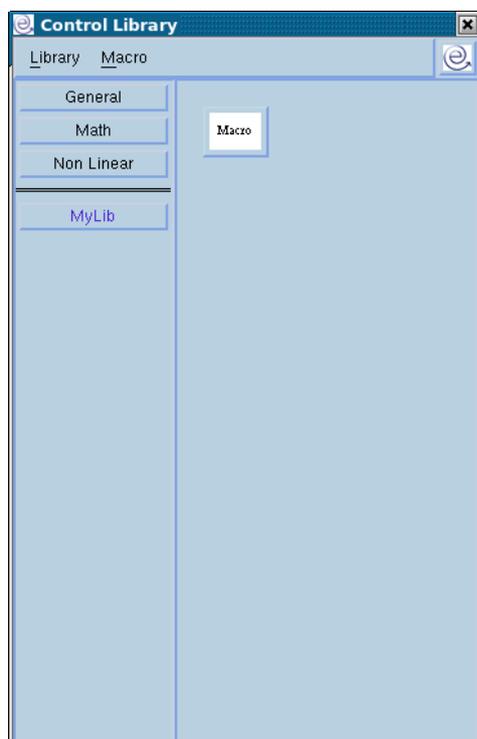
When you finish building your *macro*, you can close its layout by means of the menu **File** → **Quit**, or by clicking on the dedicated push available among the *window* buttons.



The first time you save or close your *macro* the name of the new *macro* is requested (the default name is *Macro*):



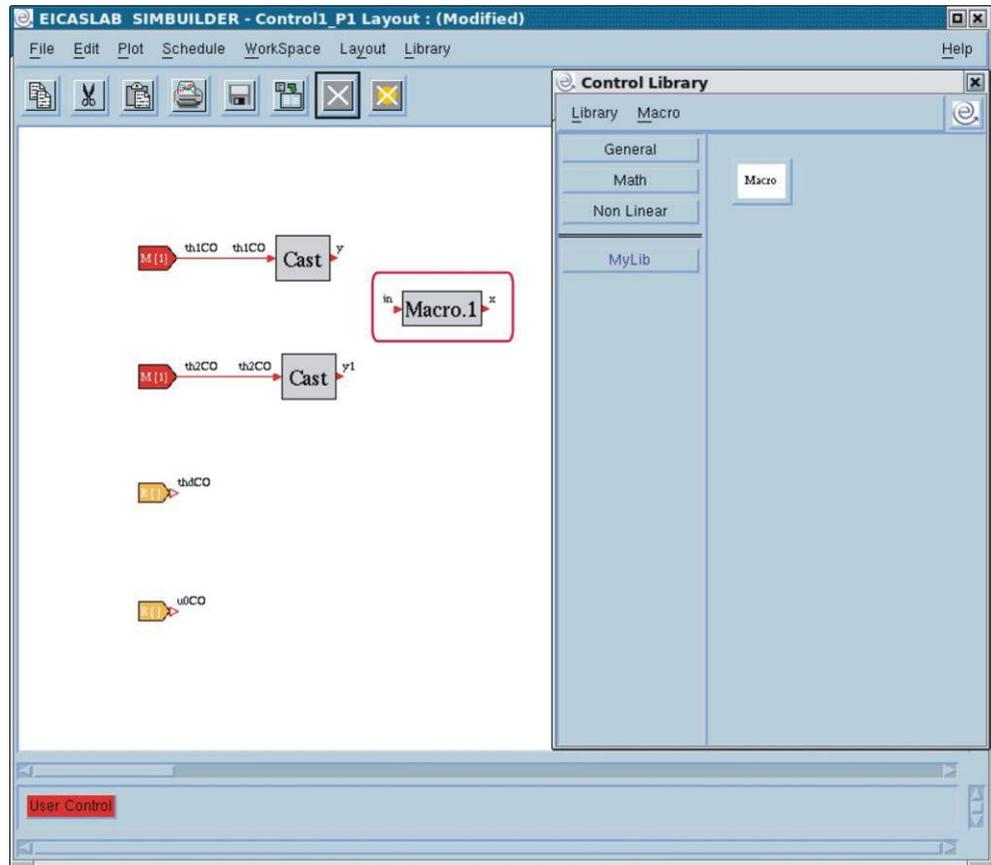
Then your *macro* will be available in the selected *user library* and an icon representing it will appear in the right section of the *library window*:



Instances of Graphical Macros

Once you have created your *macros* in your *user libraries*, you can use them – exactly as you use default blocks - by inserting macros in the layout you are editing through drag & drop.

Considering the example of previous paragraph, you can insert in the Control Layout the *macro* named '*Macro*':



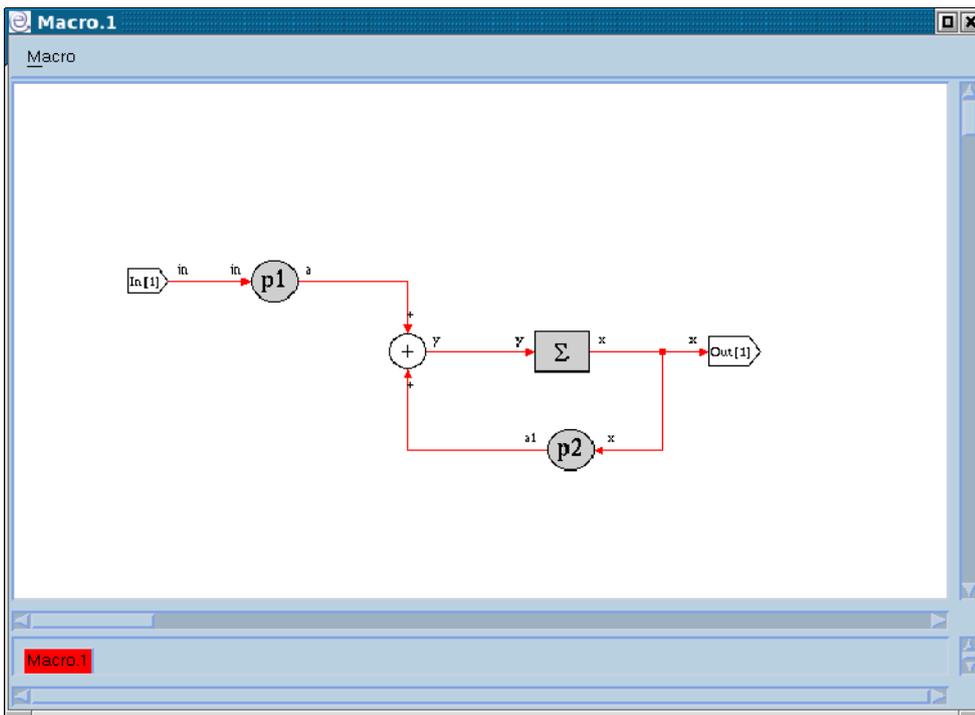
An *instance* of your *macro* appears in the layout represented by a block named *Macro.1*.

Note

The name of the *instances* of the *macros* is:

<macro name>. <ID number of the instance>.

You can customise the parameter values of any *instance* by double clicking on the *instance*, or by selecting the **Open Layout** item in the *popup menu* of the *instance*: a *Graphical Macro* layout representing the *macro* gets opened:



Note

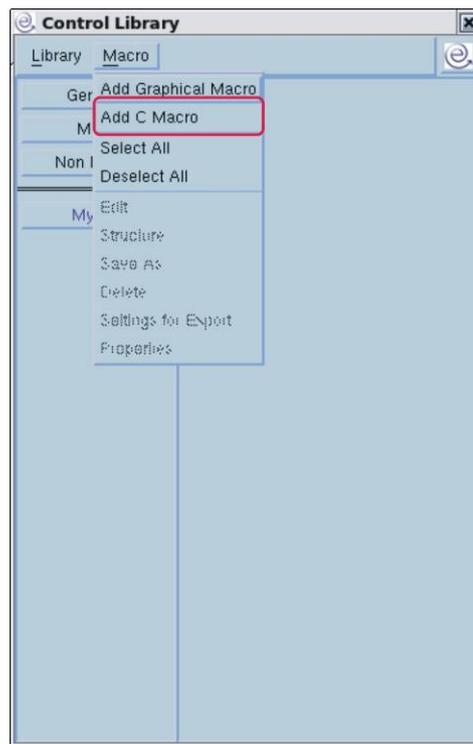
In the Macro layout you cannot change the structure of the *macro* but you can modify the value of the parameters.

C Macro

Creation of a C Macro

If you want to create a *C Macro* in the Control Layout, after having inserted in your project a graphical Control block, you must:

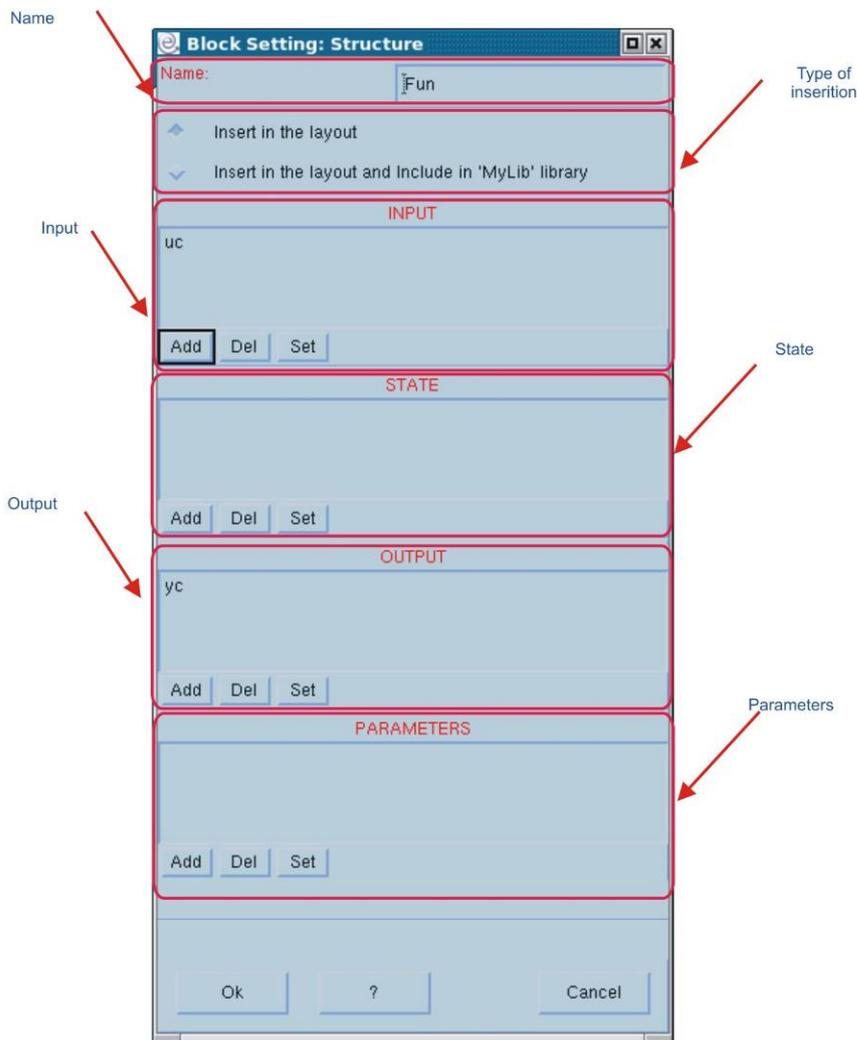
1. open the Control Layout (by double clicking on the Control block or by selecting the **Open layout** item of its popup menu),
2. create in the *library window* of the Control Layout the *user library* that will contain your *C Macro*,
1. select such a library and click on the menu **Macro** → **Add C Macro** of the *library window*.



The *"Block Setting: Structure"* window gets opened.

The structure window of the C Macros

The *"Block Setting: Structure"* window is represented in the following figure:



The window is subdivided in the sections below described:

Field 'Name'

It shows the current name of the *macro*, which can be modified.

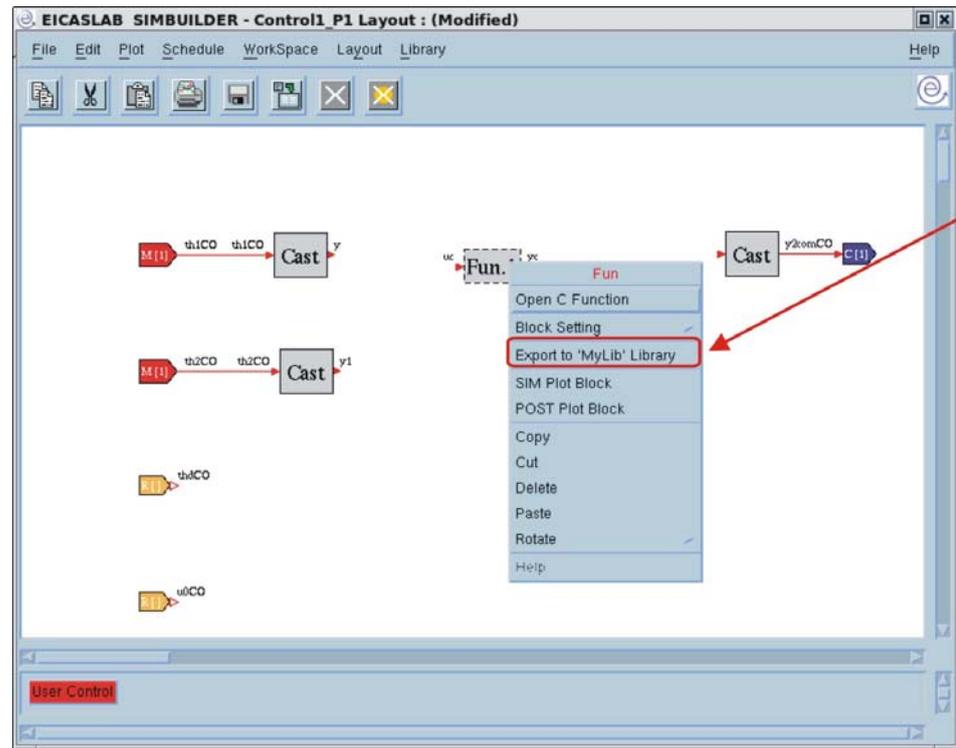
Section 'Type of insertion'

This section is only present when you are creating the *C Macro*.

- If you select the *Insert in the layout* option, EICASLAB creates a *C macro* in the layout you are editing.
- If you select the *Insert in the layout and include in <MyLib> library* option, EICASLAB creates a *C macro* in the layout you are editing and inserts the *C Macro* in the selected *user library*.

When the *C Macro* is just inserted in a layout and not inserted in a *user library*, you can always include it in a *user library* by selecting the ***Export to <MyLib>¹ Library*** item in its popup menu:

¹ It indicates the name of the *user library* in which you can insert your *C Macro*: it corresponds to the *user library* that was selected when you created the *C Macro*.



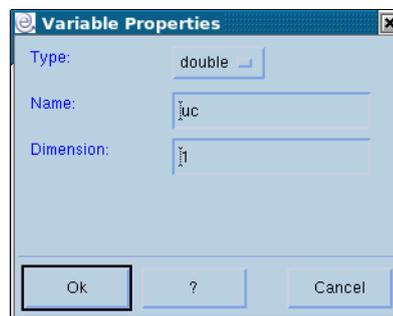
Sections: INPUT, STATE, OUTPUT, PARAMETERS

In the "Block Setting: Structure" window you can define Input, State, Output and Parameter variables of the *C Macro*.

You can:

- add new variables (through the *Add* button);
- delete a selected variable (through the *Del* button);
- modify existing variables (through the *Set* button or double clicking on the variable).

The *Add* and *Set* buttons open the "Variable Properties" window:



in which it is possible to:

- select the type of the variable (*int*, *float* or *double*),
- fix the name of the variable,

- fix the dimension of the variable: inputs, outputs and states can be scalars or arrays of dimension 1, parameters can also be arrays of dimension 2.

Note

If you do not define any state variable, the *macro* is *static*, otherwise it is *dynamic*; this choice performs a different scheduling of the *C Macro* source code in simulation.

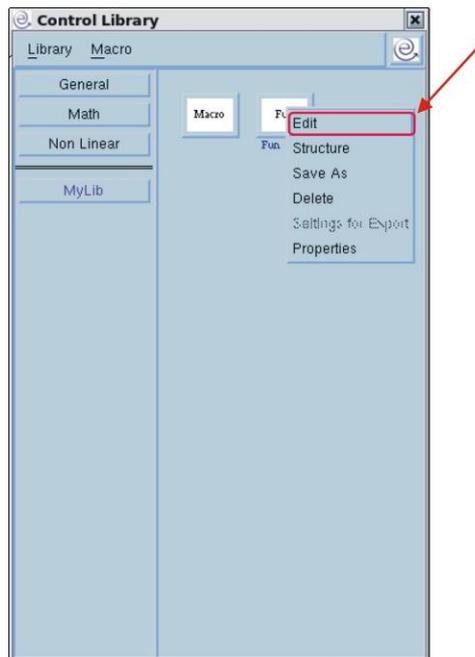
C file of the C Macros

Every *C macro* is associated to a file template in ANSI C Language that appears automatically when you run the first time the “*Block Setting: Structure*” window;

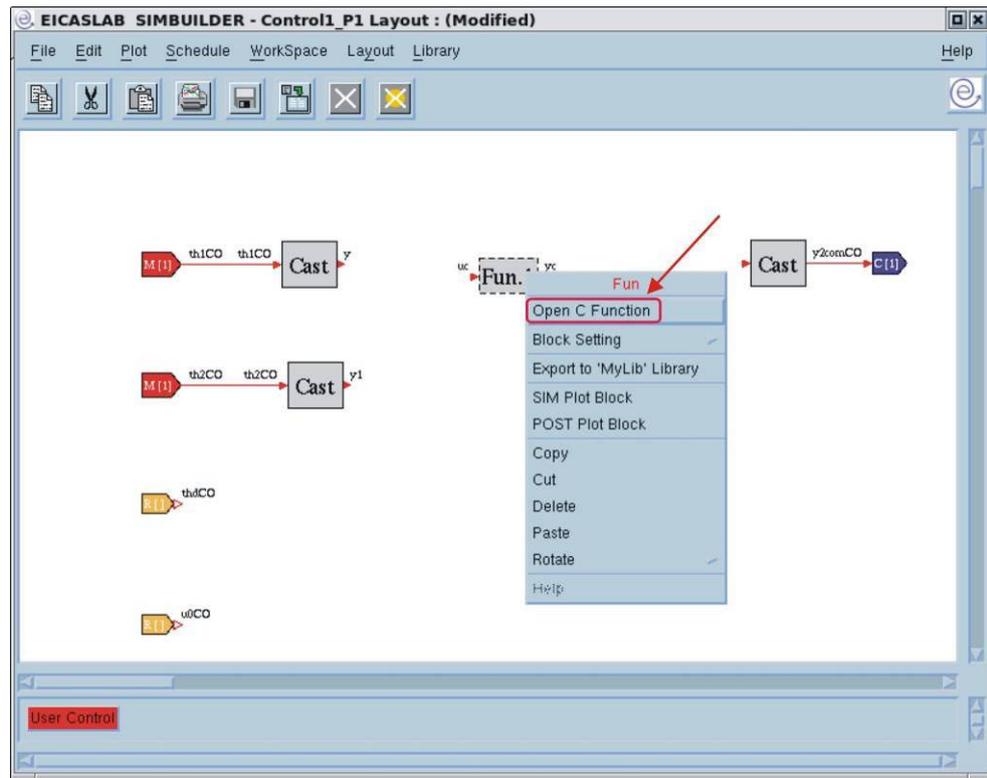
C MACROS EDITING

The C file associated to a *C Macro* can be edited:

when the *C Macro* is not inserted in a *user library*, by clicking on the ***Edit*** item of the popup menu of the *C Macro*:



when the *C Macro* is just inserted in a layout and not available in a *user library* and, select the **Open C Function** item of its popup menu:



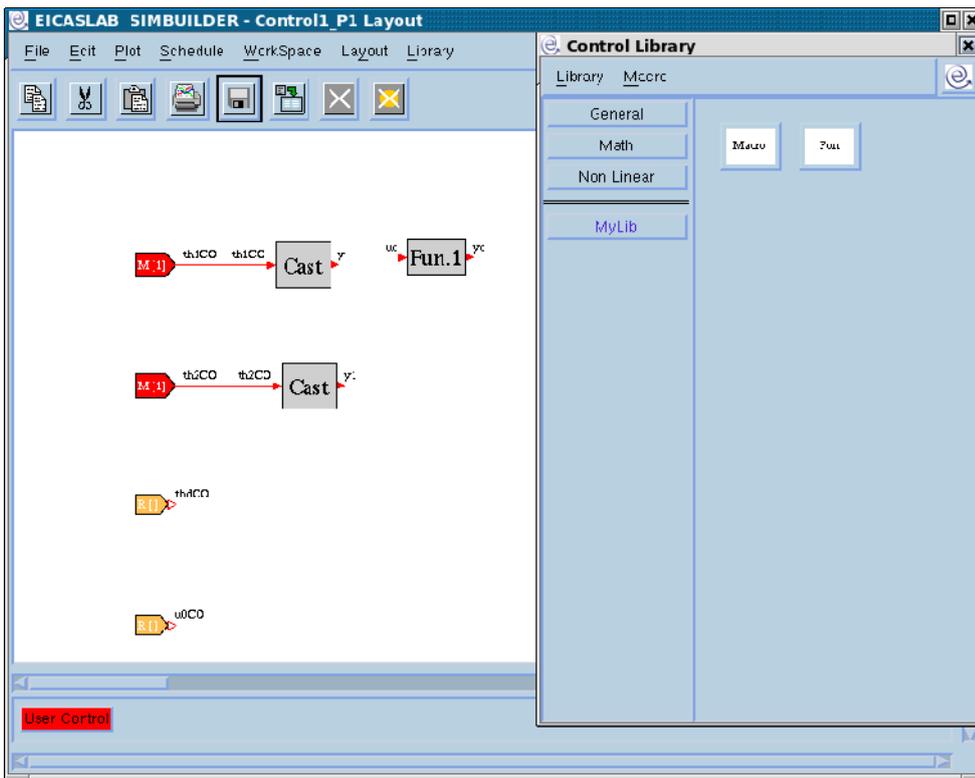
The C file contains different functions according to the static or dynamic characteristic of the *macro*. The functions available are the following:

Function	Description
<MacroName>_Ini	Function executed before starting the simulation. It can be used for initialisation purposes
<MacroName>_Exe	Function used to calculate the state evolution of dynamic <i>macros</i> or the output of static <i>macros</i>
<MacroName>_Out	Function used to calculate the output of dynamic <i>macros</i>
<MacroName>_Fin	Function executed at the end of the simulation.

Instances of C Macros

Once you have created your *macros* in your *user libraries*, you can use them – exactly as the defaults blocks - by inserting them in the layout you are editing through drag & drop.

Considering the example of the previous paragraph, you can insert in the Control Layout the *macro* 'Fun':



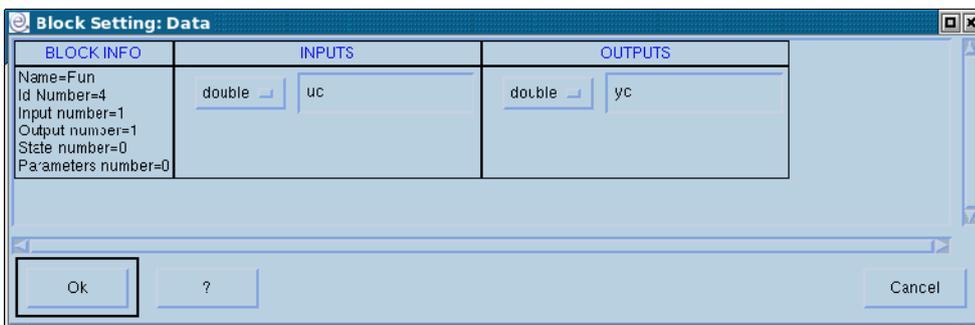
You then have an *instance* of your *macro* (the block that has the name *Fun.1*) in the layout.

Note

The name of the *instances* of the *macros* is:

<macro name>.<ID number of the instance>.

You can customise the parameter values of any *instance* by double (or central) clicking on the *instance*, or by selecting the **Block Setting** → **Data** item in the popup menu of the *instance*: a “Block Setting: Data” window gets opened:

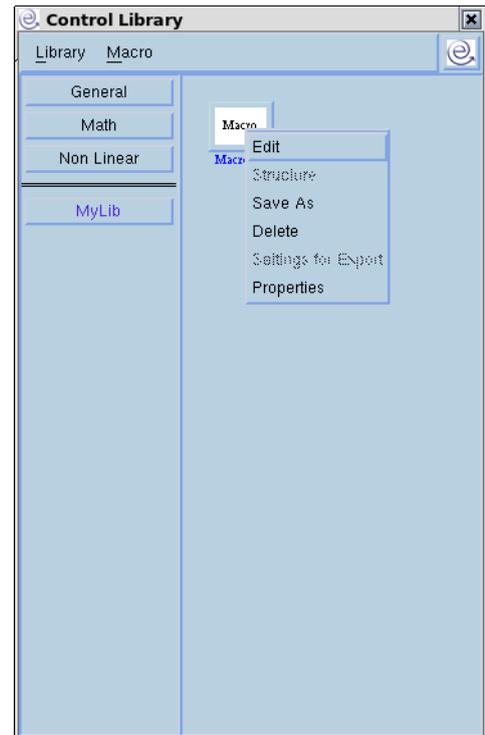


in which you can:

- view (but not modify) the name of the inputs,
- modify the parameter values,
- modify the value of the initial states, if there are states,
- view (but not modify) the name of the outputs.

Macro popup menu

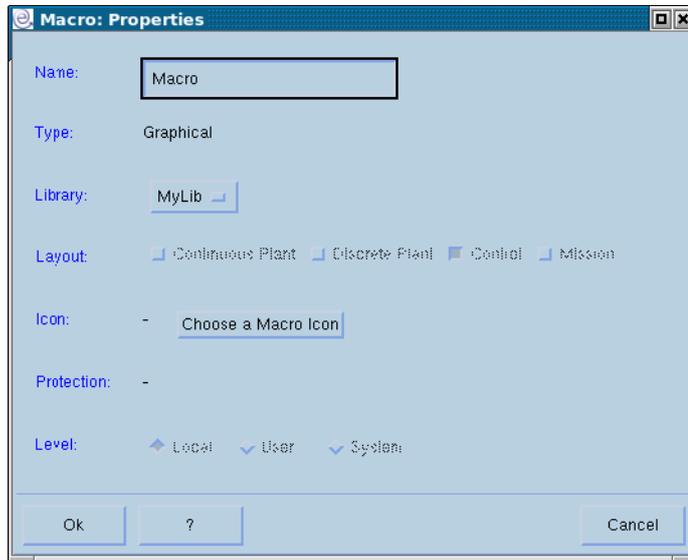
To view the *macro* popup menu right click on its icon in the *library window*:



Item	Description
Edit	Allow to modify the <i>macro</i> . It opens the C source file for <i>C Macros</i> , and the <i>Graphical Macro Layout</i> for <i>Graphical Macros</i> .
Structure	Open the “ <i>Block Setting: Structure</i> ” window (available only for <i>C Macros</i>), to modify the structure of the <i>macro</i> .
Save as	Create a copy of the <i>macro</i> with a new name.
Delete	Delete the <i>macro</i> .
Settings for Export	Not available in the EICASLAB DEMO
Properties	See paragraph “ <i>Properties window</i> ” (page 59)

Properties window

If you select one *macro* and click on the ***Properties*** item of its popup menu, the following window gets opened:



where each single field is below described:

Field 'Name'

It shows the current name of the *macro*, which can be modified.

Field 'Type'

It indicates whether the *macro* is graphical or C.

Field 'Library'

It indicates the name of the *user library* to which the *macro* belongs; it is possible to change *user library* through the option menu button.

Field 'Layout'

This field shows the current layouts in which the *macro* can be used: in the EICASLAB commercial version it is possible to extend the use of the *macro* to other layouts, this operation is not available in the EICASLAB DEMO.

Field 'Icon'

Each *macro* is represented by an icon in its *user library*; by default the icon contains the name of the *macro* (max 5 characters are visible).

Note

It is possible to provide a personal icon for the *macro* having the following format:

xpm file with dimensions 40 x 29 pixels.

By clicking on the **Choose a macro icon** button of the “*Macro: Properties*” window, a browser gets opened for selecting your own icon. At any time it is possible to turn back to the default icon or to change icon.

Field ‘Protection’

This menu indicates whether the *macro* is protected: in EICASLAB DEMO the *macros* cannot be protected.

Field ‘Level’

Not available in EICASLAB DEMO.

In EICASLAB commercial version you have at disposal three levels of *macros*:

Local level *macros*: *macros* that can be used and modified only in the project in which they have been built,

User level *macros*: *macros* that can be used and modified only by the user who built them,

System level *macros*: *macros* that can be used and modified by every user that share the same EICASLAB.

Macro upgrade

As explained before, you can modify your *macros* in every moment according to your needs.

There are two types of modifications:

1. the modifications that do not change the *macro interface*, that is the number and/or the type of the input/output variables of the *macro*,
2. the modifications that change the *macro interface*.

A *macro* can be used in two ways:

- in the project: when an *instance* of the *macro* is inserted in a layout of the project,
- in the user library: when an *instance* of the *macro* is inserted in the layout of another *macro*.

When you modify a *macro* - depending on the type of modifications and on how the *macro* is used - EICASLAB should ask you if (and how) your changes should be extended to:

- all the related *instances* placed in your project, and/or,
- all the *Graphical Macros* of the *user library* that include the modified *macro*.

Two cases are considered:

1. the *macro interface* have not been modified:

if the changed *macro* is used in the project or in the *user library*, before saving it, a warning appears indicating that the project and/or the *user library* will be consequently changed. As possibilities you can:

- a. save the *macro*, accepting the changes in the project and/or in the *user library*,
- b. save the *macro*, but with a new name,

- c. do not save any change.
- 2. the *macro interface* have been modified:
 - a. if the *macro* is used in a *user library* you must save it with a new name,
 - b. if the *macro* is not used in the library but it is used in the project, the connections of its *instances* will be deleted.

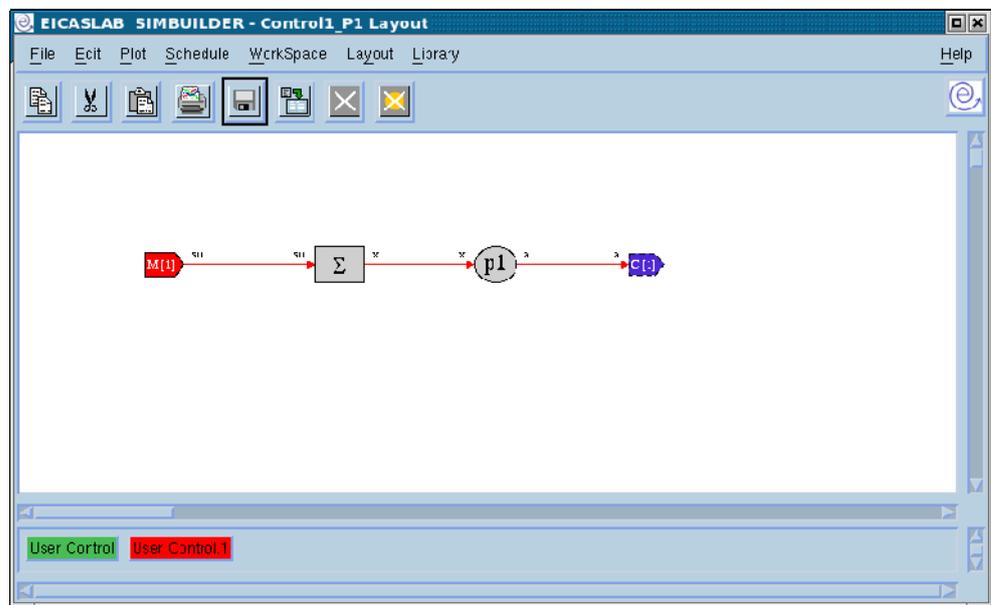
Subsystems

When you create your project in *Graphical mode*, you can simplify the representation of your system by collecting parts of your block diagram in many big blocks called Subsystems.

For doing this you can insert in the layout the following block of the *General* library:



Double clicking on the subsystem opens the *Subsystem* Layout, where you can build part of the project. You can use all the blocks available in the libraries; you can also create other subsystems in order to build a hierarchical block diagram.

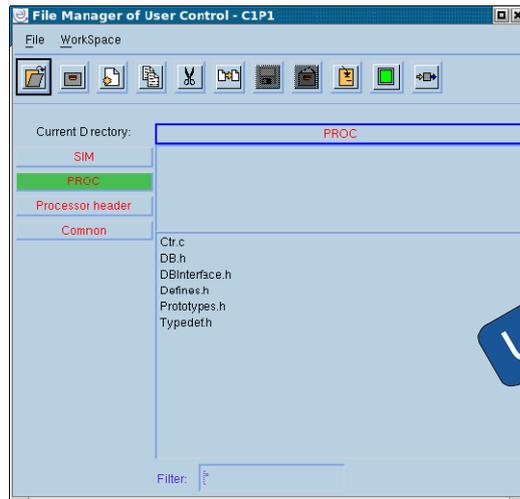


Note

Subsystems just allow you to change the graphical representation of your system. They do not affect the project scheduling.

C mode

You can build your Control (only for the *User* project) using C code. This possibility makes you free to include your pre-existent code and to avoid the limitations due to a graphical representation.



When you choose the *C mode* you have at your disposal a complete structure of files and directories subdivided in categories.

All the blocks you can program in C have generally two categories of files: one devoted to block modelling and another one containing files common to all the blocks in the project.

Control is structured in a different way: the concept is to distinguish between files only useful for simulation purposes (SIM category) and files for control algorithm design (PROC category) which can constitute your Application Software.

A complete description of input/output variables available in the Control in the *User* project is reported in the Appendix "I/O Control variables for User project".

Control files

SIM

File	Short description
CtrDes.c	Contain functions to implement the control system design such as the pole placement. These functions are called just once at the beginning of the simulation.
Prototypes.h	Contain the prototypes.
ReadPar.c	Contain functions to read parameters stored in the Ctr.par file in Param directory. These functions are called just once at the beginning of the simulation.
RWState.c	Contain functions to read initial conditions (stored in the Ctr.inistate file, IniState directory) and to write final conditions (to be stored in the Ctr.finstate file, FinState directory).
Ctr.par	Present the Control parameters (Par directory)
Ctr.inistate	Present the initial conditions of the Control states (IniState directory)
Ctr.finstate	Present the final conditions of the Control states (FinState directory)

PROC

File	Short description
DB.h	Contain the definitions and the declarations of the global variables (global to all the files of the working block)
DBInterface.h	Contain the definitions and the declarations of the input and output variables of the working block
Defines.h	Contain the defines
Ctr.c	Contain functions to implement the control algorithm. These functions are called at any simulation step.
Prototypes.h	Contain the prototypes
Typedef.h	Contain the definition of new types

Processor header

File	Short description
DBP.h	Processor header

Common

File	Short description
Common.h	Contain the definitions and the declarations of the global variables, global to all the files of your project; (not enabled in EICASLAB DEMO).

Functions

C files provided by EICASLAB are structured in functions able to help you in building your systems. Each function is well commented and provides all the necessary information: descriptions of inputs, outputs and purpose of the function are reported.

The name of each function is determined by the following rules:

$$\text{BlockId_FunctionTag}()$$

BlockId depends on the block you are programming, for Control C#P\$ (Control number # of the Processor \$) is used².

FunctionTag depends on the purpose of the function itself:

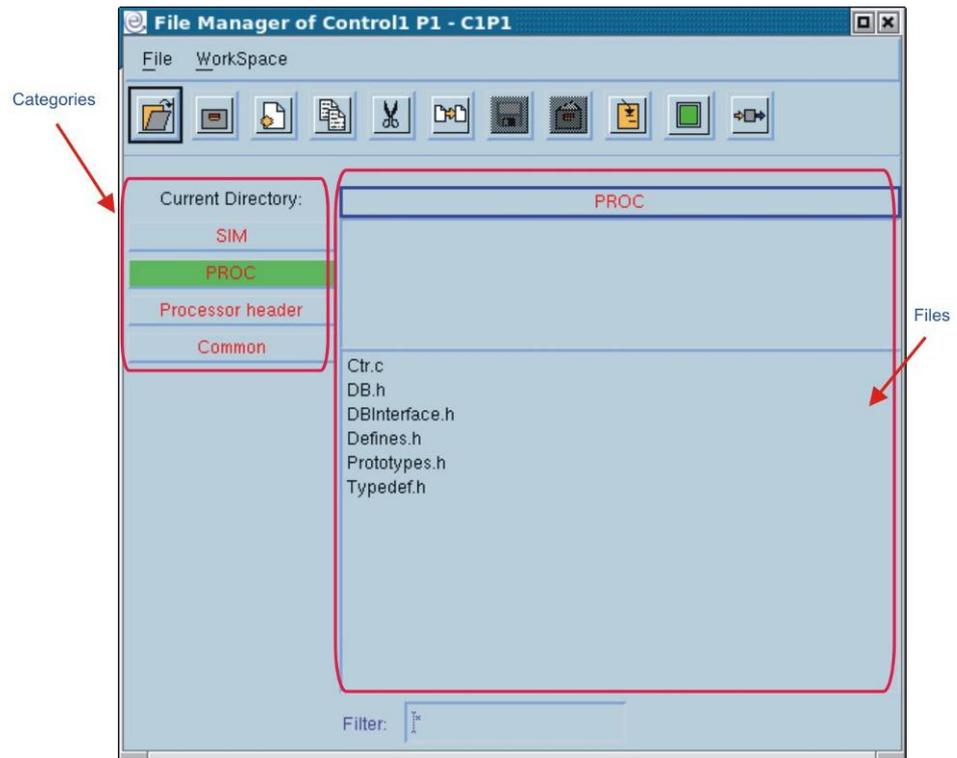
FunctionTag	Short description
ReadPar	Read block parameters before simulation
ReadState	Read initial value of the block states before simulation
WriteState	Write final value of the block states after simulation
Ini	Implement operations to be done at the beginning of the simulation (for instance variables initialisations etc...)
Exe	Execute the core algorithm of the block
Out	Calculate outputs (only for dynamic blocks)
Fin	Implement operations to be done at the end of simulation

² In the DEMO version you have just one control, then the BlockId is *C1P1*.

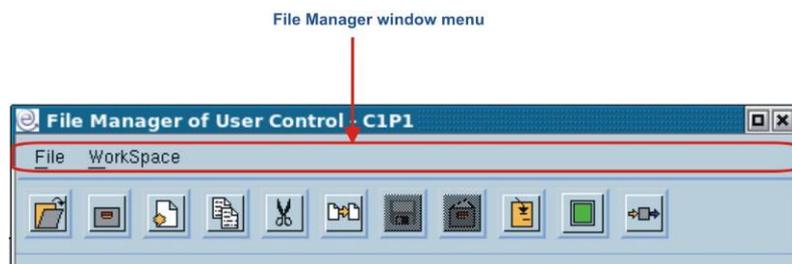
File Manager

You can manage the C files by means of a complete and easy to use File Manager window.

On the left frame it is possible to see the categories in which the files are subdivided. The right frame shows, on the top, the folders composing every category, and on the bottom, the files present in each folder (a filtering option is available).



The top section of the File Manager window contains menus and related buttons able to drive you in building your block.



File menu

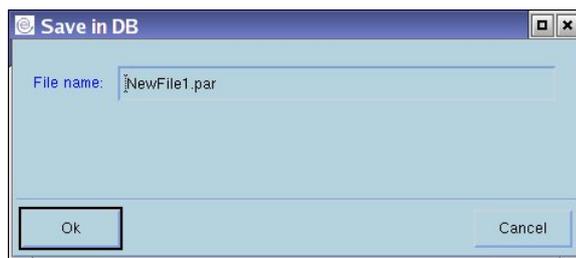
File menu contains items devoted to the C files management.



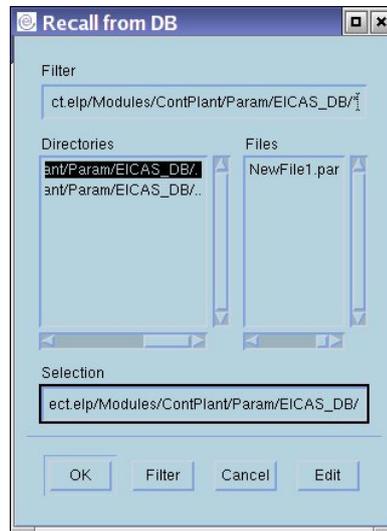
Item	Description
Open	Open the selected file
New Directory	Create a new directory
New File	Create a new file
Copy	Copy the selected file or directory
Delete	Delete the selected file or directory
Rename	Rename the selected file
Save in DB	Allow to store a parameter file (files <code>.par</code> , <code>.inistate</code> or <code>.finstate</code>)
Recall from DB	Allow to reload a parameter file stored by means of "Save in DB"
Grep	Allow a research of string in the selected category
Open Terminal	Open a terminal
Input/Output	Allow to modify input and output variables of the block
Quit	Close the file manager window

Save and Recall from DB

File manager allows you to store and reload a particular set of parameter files. You can store all the files contained in the Param, Inistate and FinState directories.



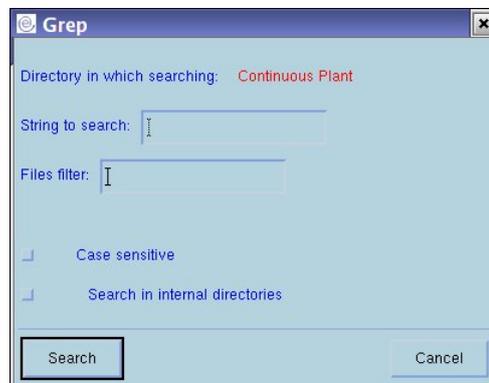
Then you can reload the stored files in order to obtain a previous block configuration.



Grep

File manager allows you to search for a particular string in the file or in the directory you select.

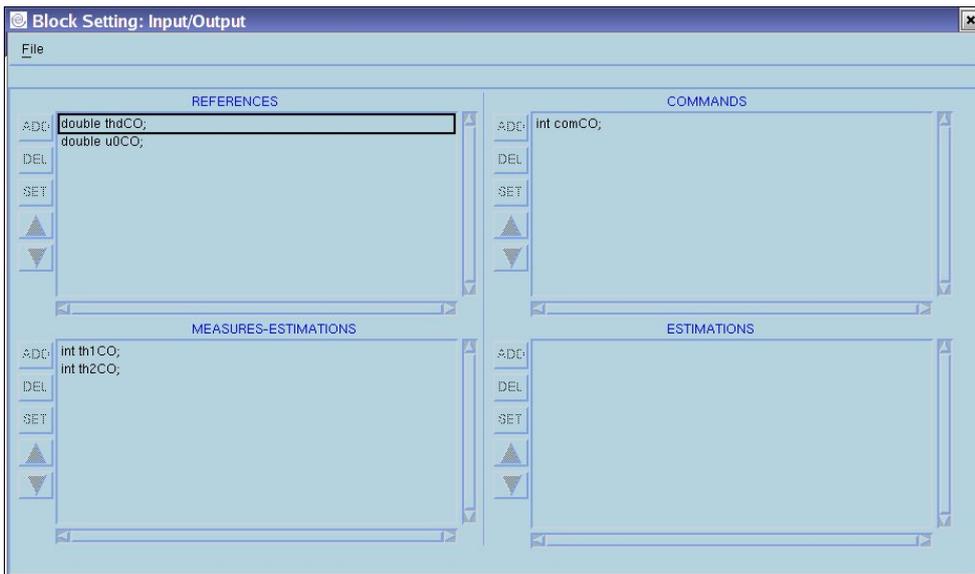
It is possible to perform a recursive research and to use filter options.



Input/Output

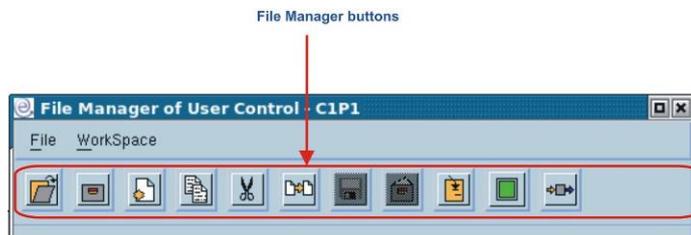
From the file manager window you can modify input and output variables of your block: this function is not available in EICASLAB DEMO because it is not possible to change the control interface.

Input and output variables are subdivided in groups according to the purpose they assume in the block you are programming. So, for example, you can find COMMANDS as output of Control.



Buttons

The top section of the File Manager window contains menus and related buttons able to drive you in building your block.



Icon	Name	Description
	Open	Open the selected file
	New Directory	Create a new directory
	New File	Create a new file
	Copy	Copy the selected file or directory
	Delete	Delete the selected file or directory
	Rename	Rename the selected file

	Save in DB	Allow to store a parameter file (files .par .inistate or .finstate)
	Recall from DB	Allow to reload a parameter file stored by means of "Save in DB"
	Grep	Allow a research of string in the selected category
	Open terminal	Open a terminal
	Input/Output	Allow to modify input and output variables of the block

Compiling project

You can create the executable file for your simulation by choosing the **Compile** button on the MASTER window. The compilation will be instantly on progress on the “*Compile*” window.

Here you can see all the messages displayed by the Linux compiler during its work, including warnings and errors due to C problems. A “done” message informs you when the compilation ends successfully.



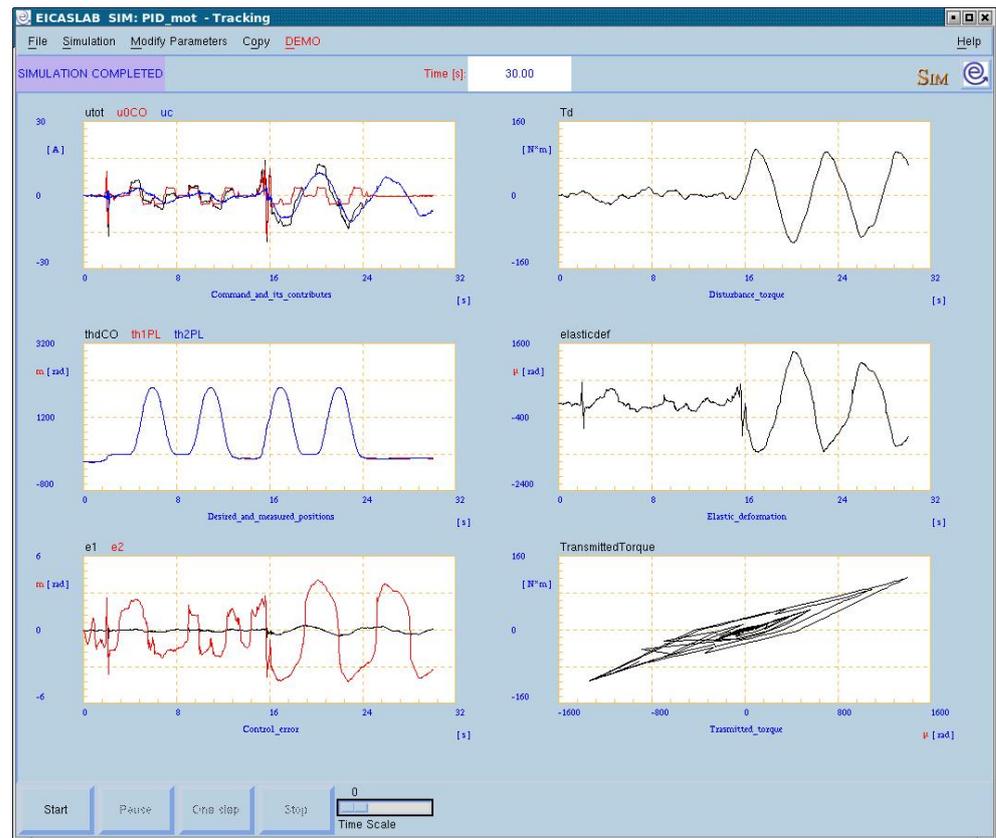
In the lower section of the “*Compile*” window some additional buttons are available, in order to allow you some additional features.

Button	Description
Stop the compilation	Interrupt the compilation
Compile again	Restart the compilation
Warnings	Open the “ <i>Warnings</i> ” window where EICASLAB presents the problems related to the system design
Exit	Close the “ <i>Compile</i> ” window

SIM

SIM is the EICASLAB tool that performs the simulation of the system you design by means of SIMBUILDER.

It appears as a window with six plotting areas where, during the simulation, you can display your variables.

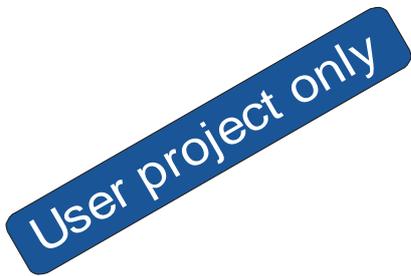


Preparing User project for simulation

In order to start the simulation of DEMO projects belonging to *PID*, *Classical Model Based* and *EICAS Model Based* families, you just need to select the **Run SIM** button on the MASTER window (see paragraph “Running projects”, page 21).

For the *User* project you need to follow some additional steps:

- 1 configuring variables to be displayed in simulation (operation to be carried out at SIMBUILDER level);
- 2 compiling the project, generating SIM;
- 3 running the simulation;
- 4 or analysing it by means of the debugger.



Configuring variables to be displayed

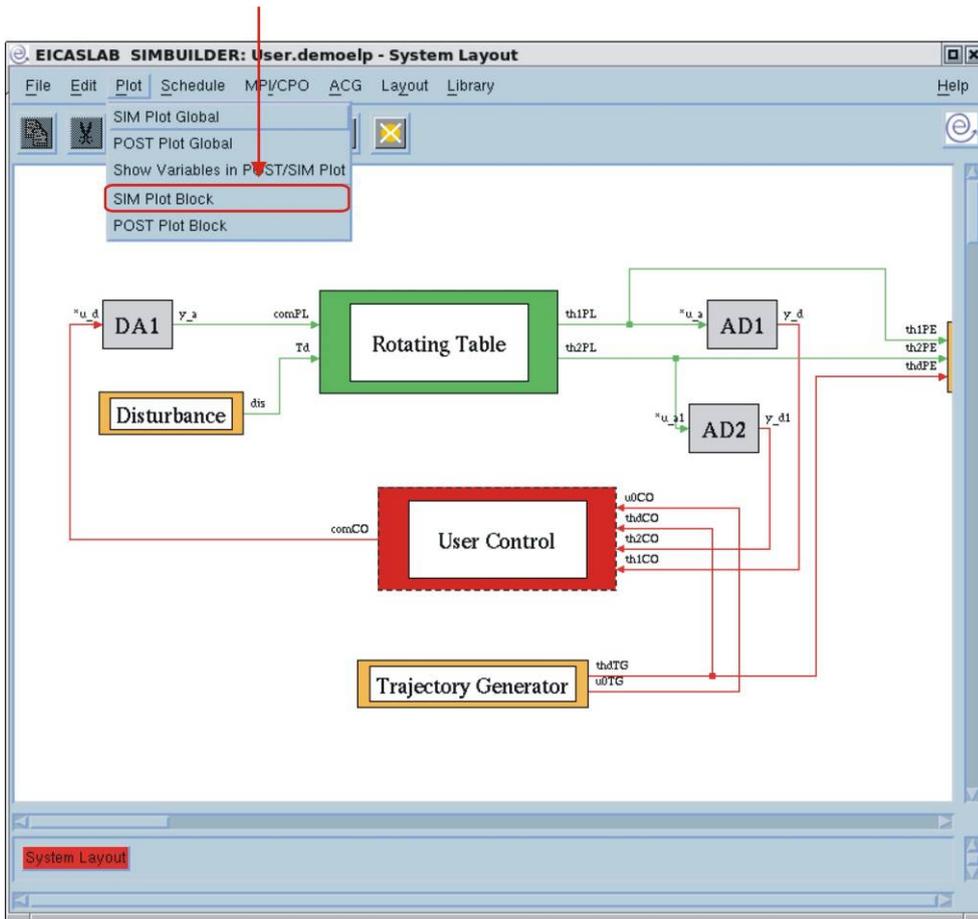
The SIM window contains six plotting areas, in each one you can display up to four variables. The variables must be selected in SIMBUILDER.

SIM Plot Block is the SIMBUILDER option you need for selecting variables for plotting: there are two ways to use this option:

- select the **SIM Plot Block** item in the popup menu of the block containing the variables you want to plot,



- select the ***Plot*** → ***SIM Plot Block*** menu of the top section of the layout of the block containing the variables you want to plot, having selected the block; for example, if you want to select variables of the user control block for plotting (which must be a C block, otherwise the user control is graphical, and you have to select its internal blocks):

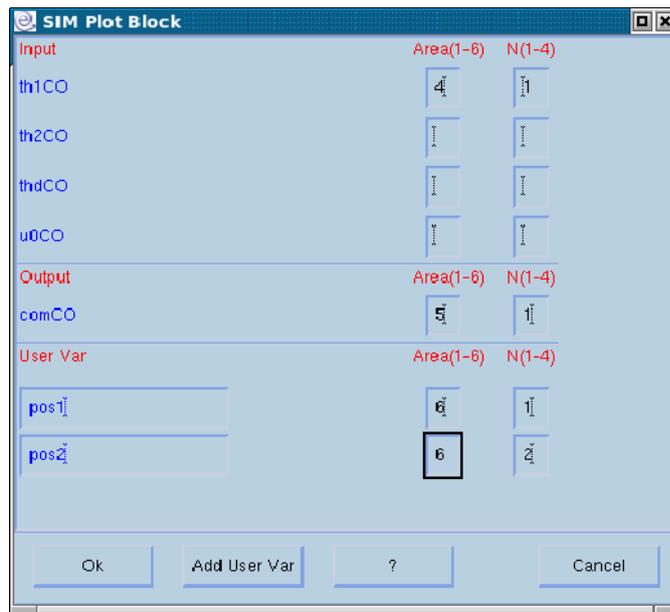


The “SIM Plot Block” window reports all the variables characterising the block: inputs, outputs, and, for dynamic blocks, states.

For the variable you need, you have to set the number of the area (from 1 to 6) in which the variable will be displayed, and its progressive number (from 1 to 4) in that area.

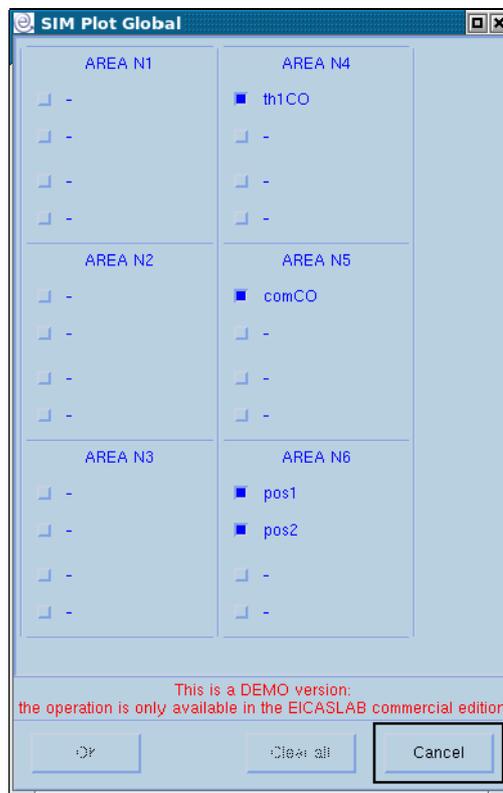
In the *User* project only areas 4, 5 and 6 are available for plotting user control variables.

For blocks programmed in *C mode*, the “SIM Plot Block” window allows you also to add new internal variables to be displayed (*Add User Var* button).



You can remove from plot any user variable clearing its name or its number of area or its progressive number.

SIM Plot Global item of the *Plot* menu of SIMBUILDER opens a window where a résumé about the variables selected for simulation purposes is presented.



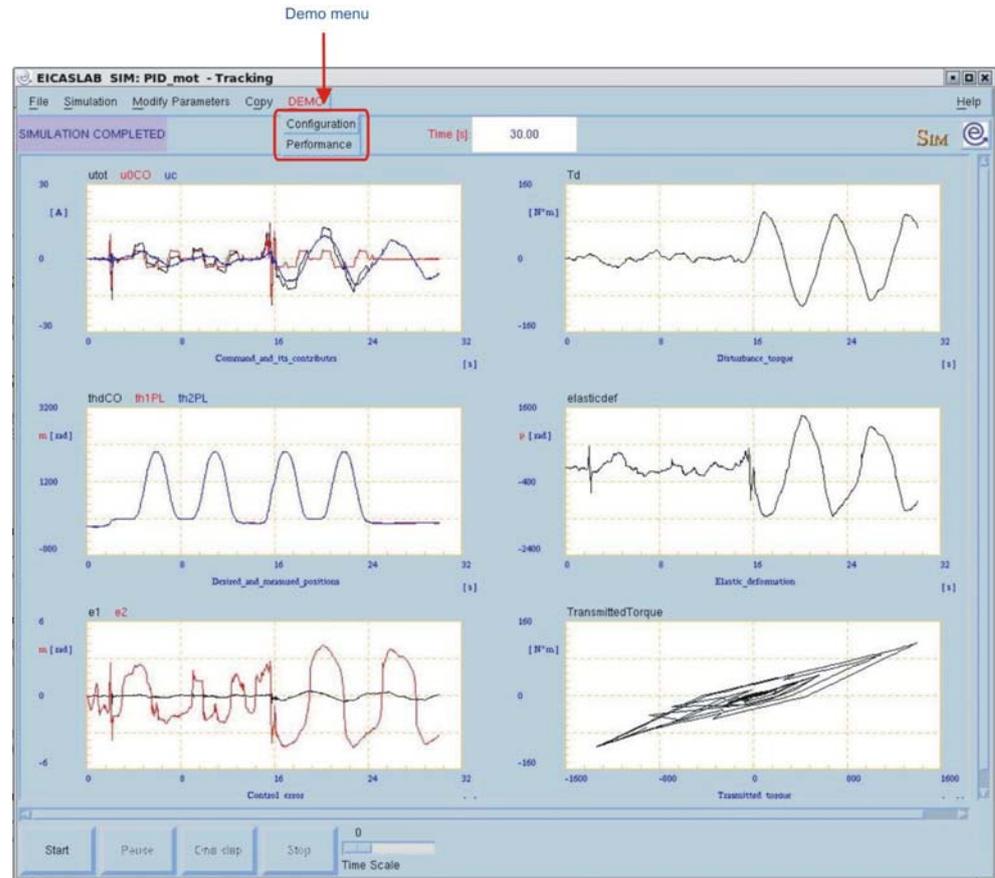
Here you can deselect the variable you do not want to display.

CLEAR ALL button deselects all the variables (available just in EICASLAB commercial version).

Running the simulation

You can run SIM by clicking on the **Run SIM** button on the MASTER window.

SIM appears as a graphical window where, in foreground, six plotting areas are displayed. The lower section is designed to manage the simulation (it includes **Start**, **Stop** buttons etc...). The menu section presents all the items for configuring the simulation and for changing project parameter values.



Plotting areas

Each area can display up to four variables. The colours used for the plot are:

Variable number	Colour
1	Black
2	Red
3	Blue
4	Green

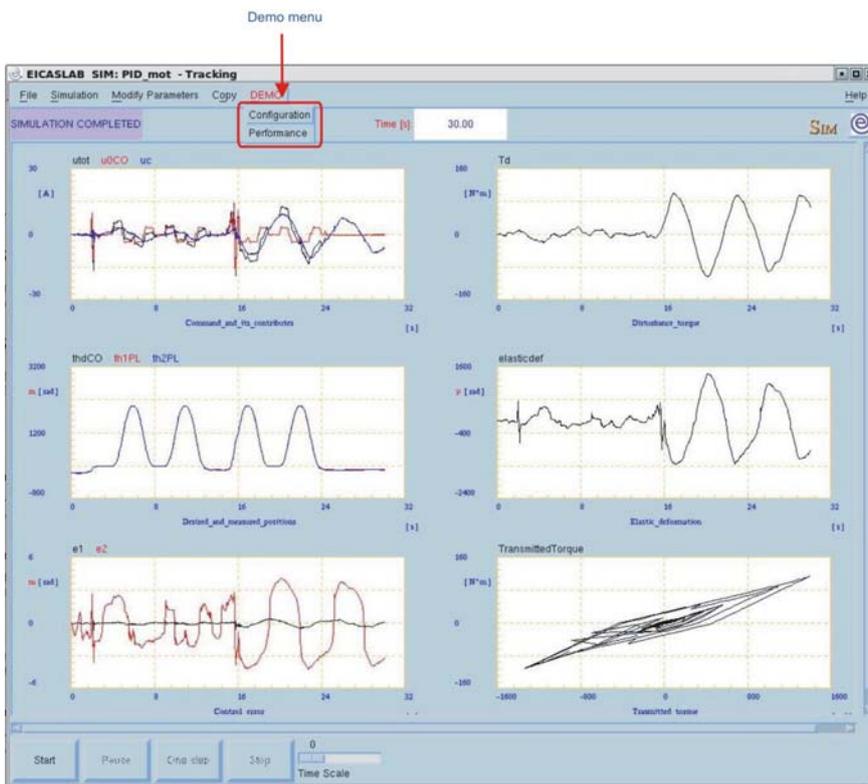


Abscissas and ordinates values are automatically defined by EICASLAB during the simulation. You can modify the name of the graph and the unit of measurement of both abscissa and ordinate by double clicking on their labels.

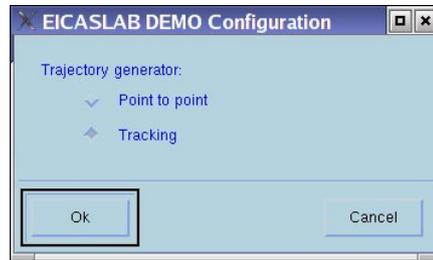


Demo functionalities

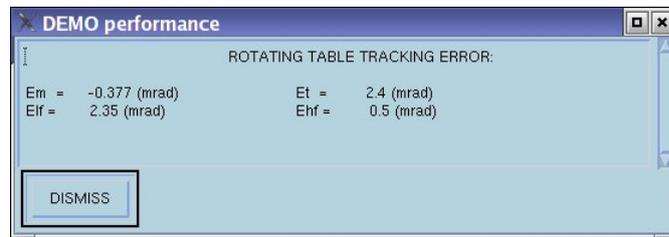
The **DEMO** menu provides some useful features able to let you easily configure and analyse demo simulations.



The **Configuration** item allows to choose the operating mode of the trial: point to point or tracking (see paragraph “Operating modes”, page 18, for more details).

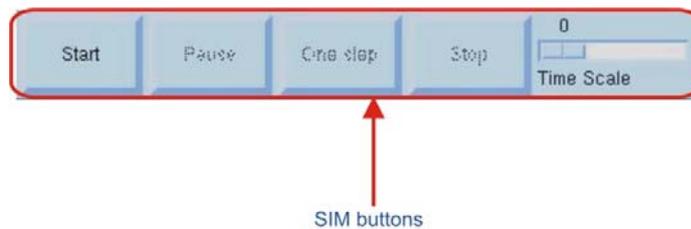


The **Performance** item, available only for locked projects (belonging to *PID*, *Classical Model Based* and *EICAS Model Based* families) summarises the performance indexes of the trial according to what reported in the technical note *EICASLABDEMOtestcases.pdf*.



Managing the simulation

The lower section of the SIM window presents a set of commands you can use for managing your simulation.



Button	Description
Start	Run the simulation
Pause	Stop temporarily the simulation. If you press Start the simulation will continue, if you press Pause one simulation step is executed
One Step	Run one step simulation at a time
Stop	Stop the simulation. If you press Start the simulation will restart from the beginning

The **Time-Scale** slider is useful to change the duration of the simulation and allows “like real-time” simulations: it is possible to obtain simulations whose duration is scaled by a constant factor with respect to a real-time simulation.

When **Time-Scale** is set to 0, the simulation speed is maximum, so if you have a calculator powerful enough you can obtain real-time simulations. On the other hand if **Time-Scale** is set to an x value, the simulation speed is scaled by an x factor with respect to a real-time simulation.

Debugging the User project

EICASLAB allows you to run the simulation of the *User* project in debugging mode.

You can run the debugger by clicking on the *Debug SIM* item of the SIM menu in the MASTER window,



or by clicking on the button:



EICASLAB automatically opens the debugger you set in the "Preferences" window.

```

DDD: /home/MyUser/MyProject.elp/Modules/Load/Load.c
File Edit View Program Commands Status Source Data Help
0: main
Lookup Find Break Watch Print Display Plot Show Rotate Set Undo

#include <Standard.h>
#include <GDefines.h>
#include <GTypeDef.h>
#include <GLibraries.h>
#include "../ConvAD1/Prototypes.h"
#include "../ConvDA1/Prototypes.h"
#include "../SimProc1/Proc/Ctr1/Prototypes.h"
#include "../SimProc1/Sim/Ctr1/Prototypes.h"

void PAssDBLoc(FLOATING **EICASLAB_u, int **EICASLAB_uint, FLOATING
**EICASLAB_y, int **EICASLAB_yint);

void PFunctionAssign_PL(void);
void MSAssDBLoc1(FLOATING **EICASLAB_u, int** EICASLAB_uint, FLOATING
**EICASLAB_y, int** EICASLAB_yint);
void MSAssDBLoc2(FLOATING **EICASLAB_u, int** EICASLAB_uint, FLOATING
**EICASLAB_y, int** EICASLAB_yint);
void ConvAD1AssDBLoc(double **EICASLAB_i, int **EICASLAB_o, double
**EICASLAB_m, double **EICASLAB_lq);
void ConvDA1AssDBLoc(int **EICASLAB_i, double **EICASLAB_o, double **EICASLAB_lq);
void C1P1_ReadPar(FILE *fp);
void C1P1_ReadState(FILE *fs);
void C1P1_WriteState(FILE *fs);
void C1P1_Des();
void DB1P1();
void DBo1P1();
void C1P1_post(FILE *fp, double t);
void C1P1_video(double t);

void Load()

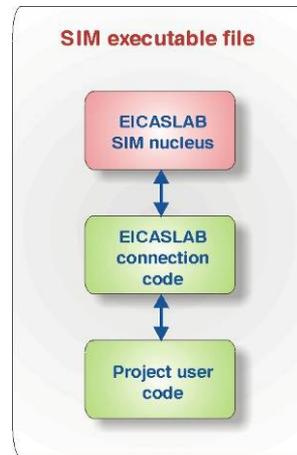
Copyright © 1999-2001 Universität Passau, Germany.
Copyright © 2001 Universität des Saarlandes, Germany.
Copyright © 2003 Free Software Foundation, Inc.
Using host libthread_db library "/lib/tls/libthread_db.so.1".
(gdb)
Welcome to DDD 3.3.8 "Cinnamon" (i386-redhat-linux-gnu)

```

The SIM executable file is created by compiling three main parts of code:

- 1 the *EICASLAB SIM nucleus*, that implements the SIM window and the integration algorithm;
- 2 the *Project user code*, that corresponds to the C code you developed when programming your project in *C mode*, or using *C macros*;
- 3 the *EICASLAB connection code*, that is the C code provided by EICASLAB for connecting the *EICASLAB SIM nucleus* with the *Project user code*.

With the debugger facility you can check and inspect the *Project user code* and the *EICASLAB connection code* of your project.



SIM window menu

The upper section of the SIM window contains the menus which help you in configuring your simulation and your project parameters.



In the following only items available in EICASLAB DEMO will be considered.

File menu

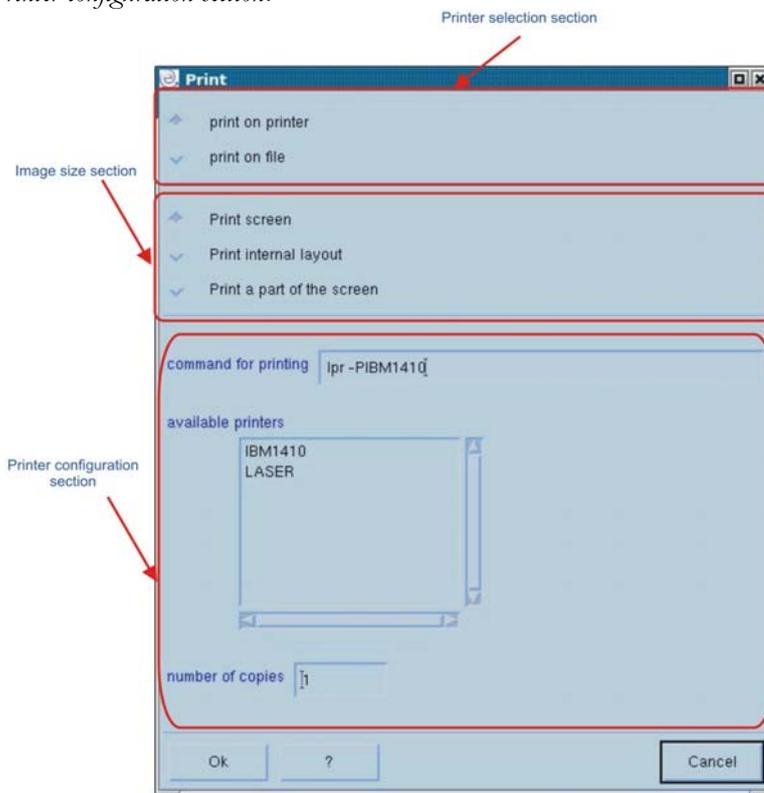
Item	Description
Print	Print the layout on a printer or on a file
Print Options	Pre-selection of print options
Quit	Close SIM

Print

Print item allows you to print your SIM window (or a part of it) on a printer or on a file. This facility is based on the `import` Linux command.

The "Print" window is subdivided in three main sections:

- *Printer selection section;*
- *Image size section;*
- *Printer configuration section.*



Printer selection section allows you to select whether you want to print on a printer or on a file.



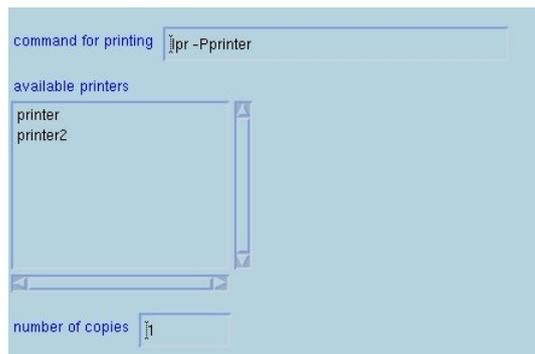
Image size section allows you to select what you want to print:

- *Print screen* option will print what you can see in the screen
- *Print a part of the screen* option will print only a selection of the screen. The selection can be done by clicking and moving the mouse around the part of the layout you want to print.

Printer configuration section reports all the options related to the printer you selected.

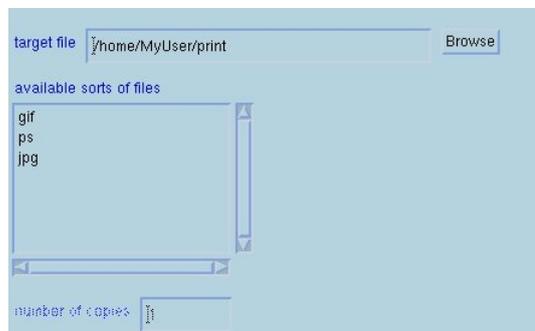
If you choose to print on a printer, you can select:

- the Linux command to use
- the printer
- the number of copies



If you choose to print on a file, you can select:

- the name of the file
- the type (gif, ps or jpg)



The result of your print is an exact copy of what you see on the screen.

The file where you printed your graphs (if you choose to print on file) is reported in the `bin` directory of your project.

Simulation menu

Item	Description
Data	Allow to configure the duration of the simulation and simulation step
Plot	Allow to enable or disable graphical plots
Random seed	Allow to set the initial value of the seed for the random function
SIM standard output	Open a window that plots the file <i>SIM.out</i> which contains the standard output
SIM Restart	Quit and restart SIM

Data

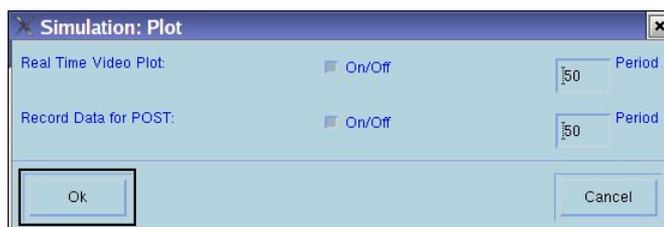
Data menu allows you to define initial time, final time and simulation step.



Item	Description
Initial Time	Starting time for simulation, expressed in seconds (initial conditions are related to this instant)
Final Time	Final time for simulation, expressed in seconds
Simulation Step	Simulation step for the simulation algorithm. It represents also the starting value for the integration algorithm

Plot

Plot item allows you to configure plotting options.

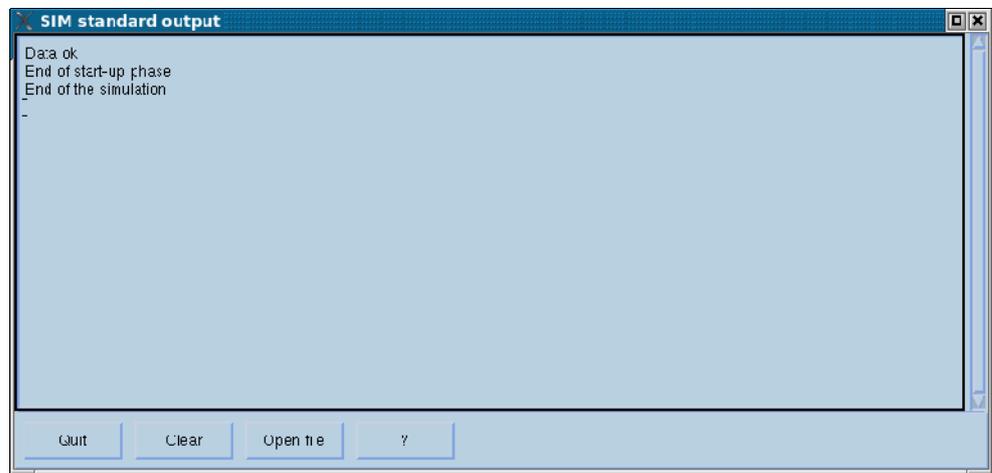


Item	Description
Real Time Video Plot	Allow you to enable or disable the plot during your simulation
Record Data for POST	Allow you to enable or disable storing data to be analysed with POST

For both items, if enabled, you can set the *Period* value: if the number selected is x , one sample of your variables will be plotted (or stored) every x simulation steps.

SIM standard output

The “*SIM standard output*” window plots the file *SIM.out* which contains the standard output:



Button	Description
Quit	Close the window
Clear	Clear what is plotted in the window (the file <i>SIM.out</i> , is not cleared)
Open file	Open the standard output file <i>SIM.out</i>

Modify parameters menu

Not enabled in EICASLAB DEMO.

Copy menu

Not enabled in EICASLAB DEMO.

DEMO menu

Item	Description
Configuration	Configure the operating mode of the trial: point to point or tracking.
Performance	Report the performance obtained in the trial.

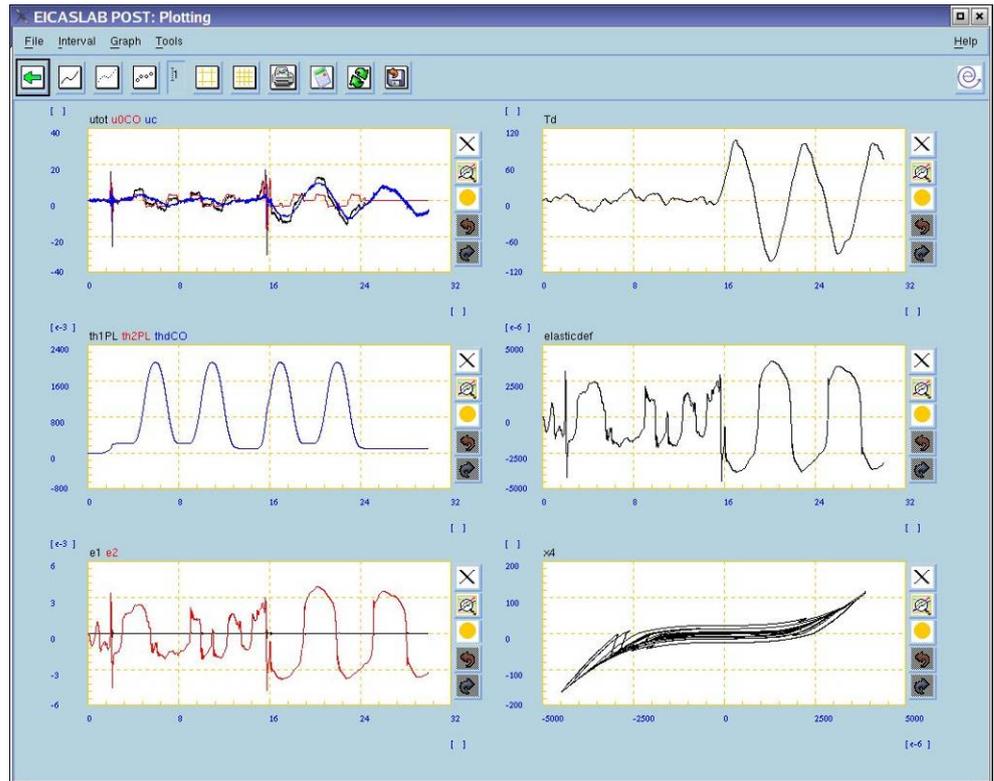
(only for the locked projects).

Help menu

Item	Description
User Manual	Runs the EICASLAB DEMO User Manual
EICASLAB on the Web	Runs the EICASLAB home page www.eicaslab.com
About EICASLAB	Displays some information about version and licence of EICASLAB DEMO

POST

POST is the EICASLAB tool for the off-line analysis of the simulation trial results. The signals stored during the simulation can be processed through a large variety of mathematical functions.



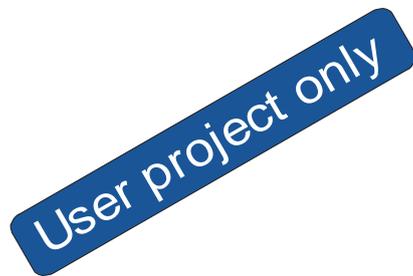
Preparing project for post-processing

POST works on data generated automatically during the simulation (by means of SIM). They are related to the variables you selected when you built your project in SIMBUILDER.

In order to process data of DEMO projects belonging to *PID*, *Classical Model Based* and *EICAS Model Based* families, you just need to select the **Run POST** button on the MASTER window (see paragraph “Running post-processing”, page 96).

For the *User* project you need to follow some additional steps:

- 1 configuring variables to be post-processed (operation to be carried out at SIMBUILDER level);
- 2 compiling the project, generating SIM;
- 3 running the simulation;
- 4 running POST.



Configuring variables to be post-processed

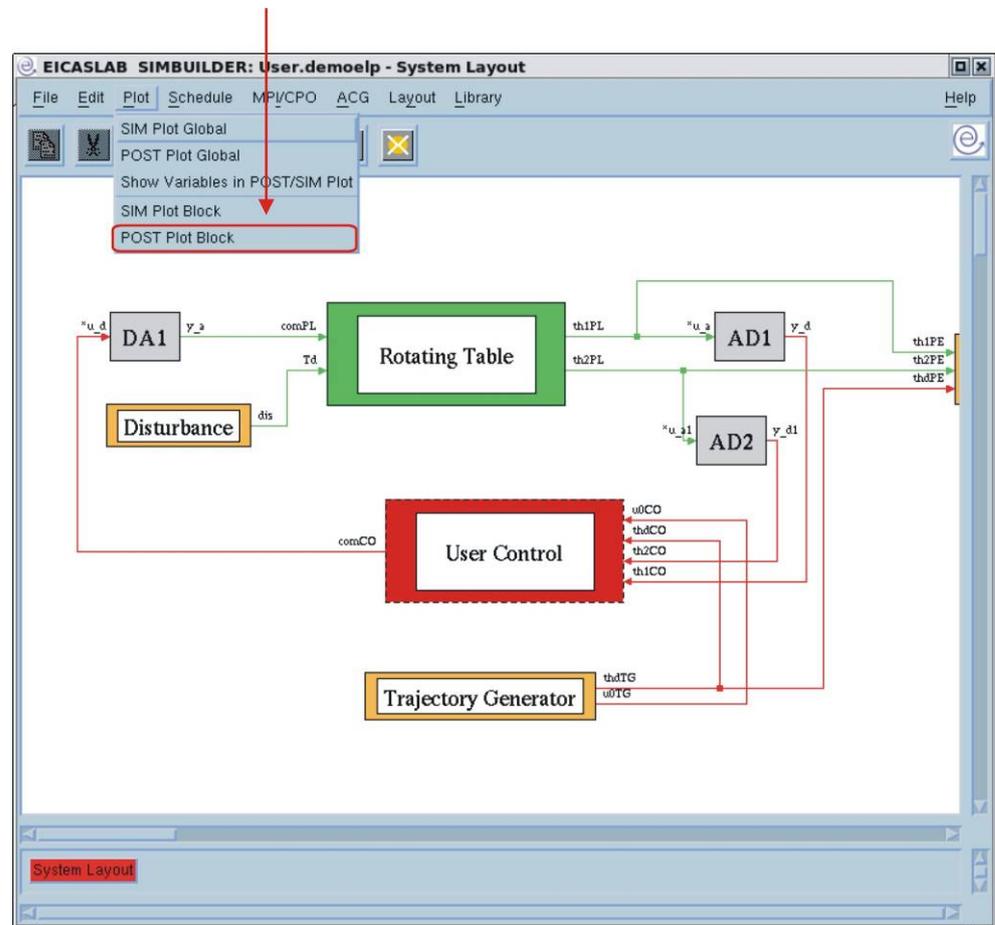
POST Plot Block is the SIMBUILDER option you need for selecting variables for post-processing: there are two ways to use this option:

- select the **POST Plot Block** item in the popup menu of the block containing the variables you want to display,



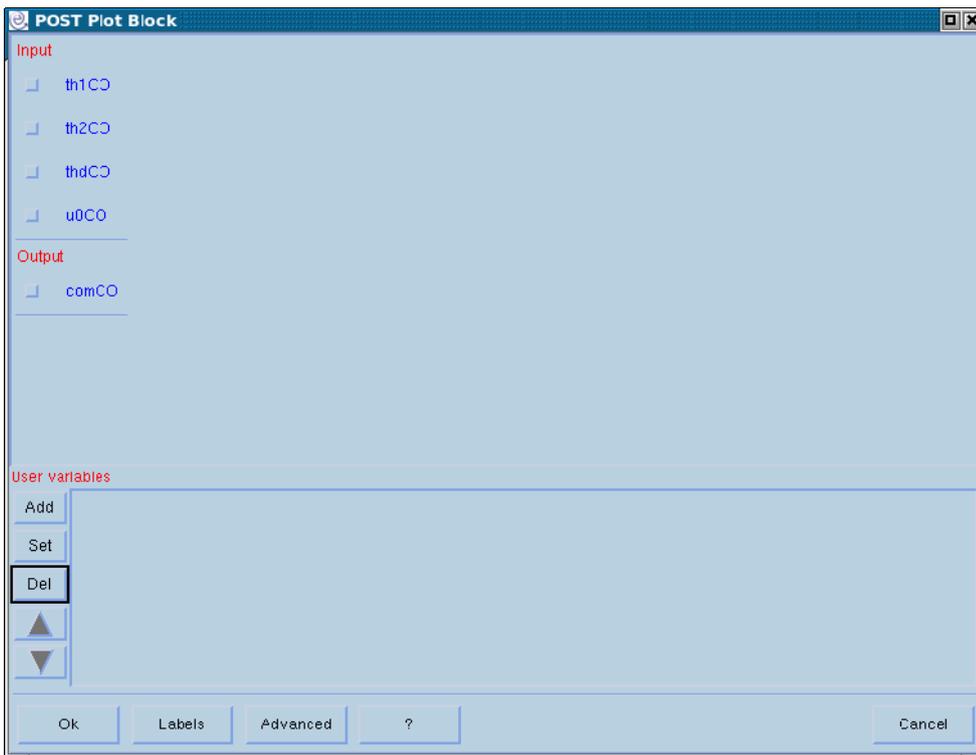
- select the **Plot** → **POST Plot Block** menu of the top section of the layout of the block containing the variables you want to display, having selected the block; for example, if you want to select variables

of the user control block for post-processing (which must be a C block, otherwise the user control is graphical, and you have to select its internal blocks):



“POST Plot Block” window presents all the variables characterising the block: inputs, outputs, and, for dynamic blocks, states.

You have just to push the button related to the variable you need: it will be automatically stored during the simulation.

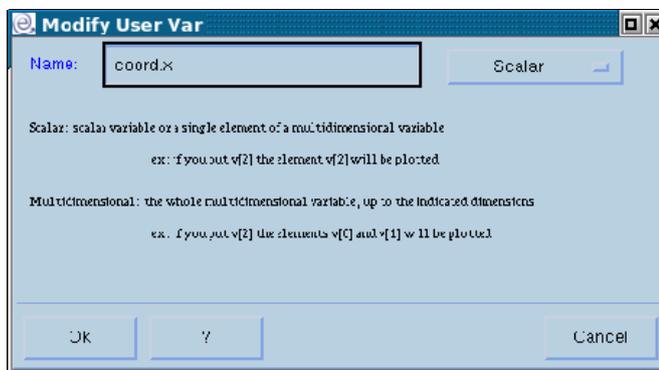


For blocks programmed in *C mode*, it is necessary to be able to plot in POST some user variables: the “*POST Plot Block*” window allows you to:

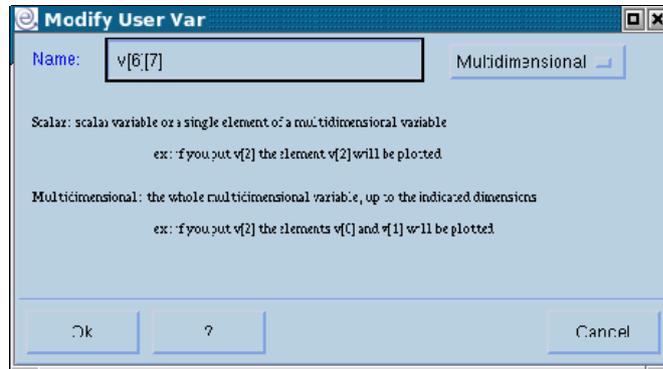
- add, in POST file, new variables, **Add** button;
- delete a selected variable, **Del** button;
- modify existing variables, **Set** button or double clicking on the variable;
- move up or down the variables in the group list (this changes the order in which the variables are graphically reported);

The *Add* and *Set* operations open a window in which you can define a new user variable or modify an already defined user variable:

you can define a scalar variable, or a single element of a multidimensional variable (which can be also a structured variable):



you can also define a whole multidimensional variable: you must then insert the dimensions of the variable that you want to plot: all the elements of the variable, up to the indicated dimensions, will be plotted:



The *Labels* button allows to modify the variable name that will be reported in the POST files; the *Advanced* button allows to write user code for the plotting; these operation are available only in the EICASLAB commercial version.

Record Data for POST

In SIM, the *Plot* item of the *Simulation* menu allows you to further configure your POST files.



From *Record Data for POST* item you can enable or disable storing POST files during the simulation.

Period item defines the decimal ratio of the POST files storage. If you set x value, one point of your variables will be stored at every x simulation step.

Running post-processing

You can run POST by clicking on the *Run POST* item of the *POST* menu in the MASTER window,



or by clicking on the button:



POST is composed by two main windows:

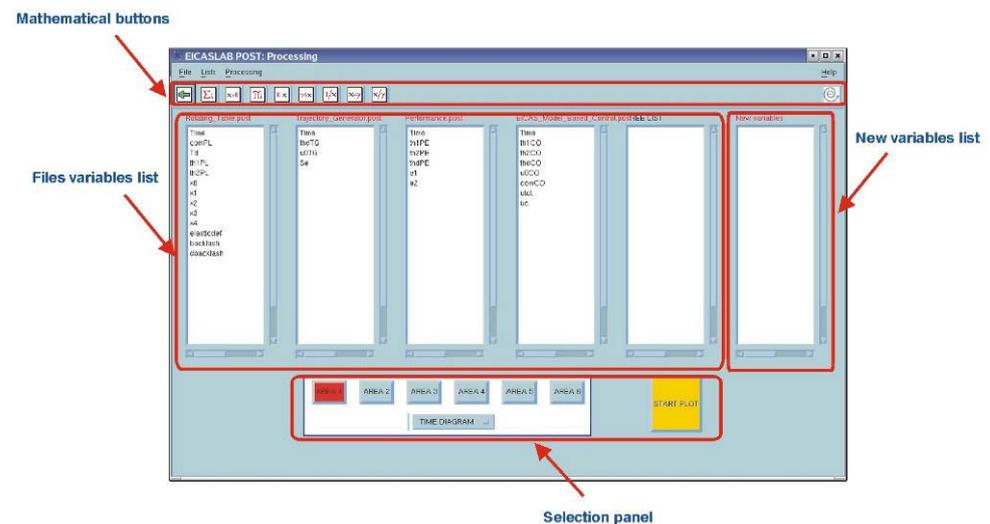
- processing window: able to perform complex mathematical computations and to select the graphs to be displayed;
- plotting window: where you can analyse the behaviour of your variables and perform statistical analysis.

Processing window

The Processing window is in charge of performing mathematical elaborations of your variables and defining the plotting areas.

This window can be subdivided into four main sections:

- *Files variables list*, that presents the contents of the POST loaded files;
- *New variables list*, containing the variables generated after mathematical elaborations;
- *Selection panel*, where you can select the plotting area where displaying your variables, and the type of graph;
- *Mathematical buttons*, useful for implementing the most common mathematical computations.



A complete description of variables available in the post files available in the Processing window is reported in the Appendix “POST variables”.

Selection panel

The selection panel is the part of the Processing window where you can define the graphs to be displayed in the Plotting window.



It is composed of six **AREA** buttons able to select one of the six areas available in the Plotting window, a diagram selector (**TIME DIAGRAM/PHASE DIAGRAM**) and a confirmation button (**START PLOT**).

Button	Description
AREA #	Allow to select variables to be displayed in the plotting area number #
TIME DIAGRAM / PHASE DIAGRAM	Select the type of diagram you want to display in the area you selected
START PLOT	Apply your configuration and run the Plotting window

If you select a time diagram, EICASLAB displays a graph where you have to choose only the ordinate variable, the abscissa is automatically the time.

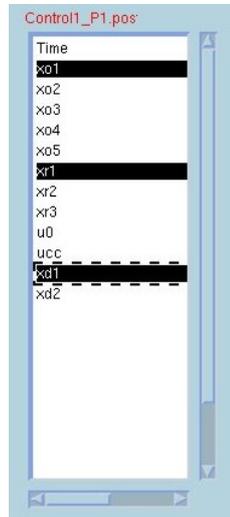
If you select a phase diagram, EICASLAB provides an X-Y graph where you have to choose both ordinate and abscissa.

The available **AREA** buttons are six because the Plotting window can display up to six areas

Variable selection and graph customisation

Each variable on the Processing lists (*Files variables lists* and *New variables list*) can be selected for plot customisation.

You can select a variable by clicking on it; the selected variable will be highlighted in black. You can deselect a variable by clicking again on it.



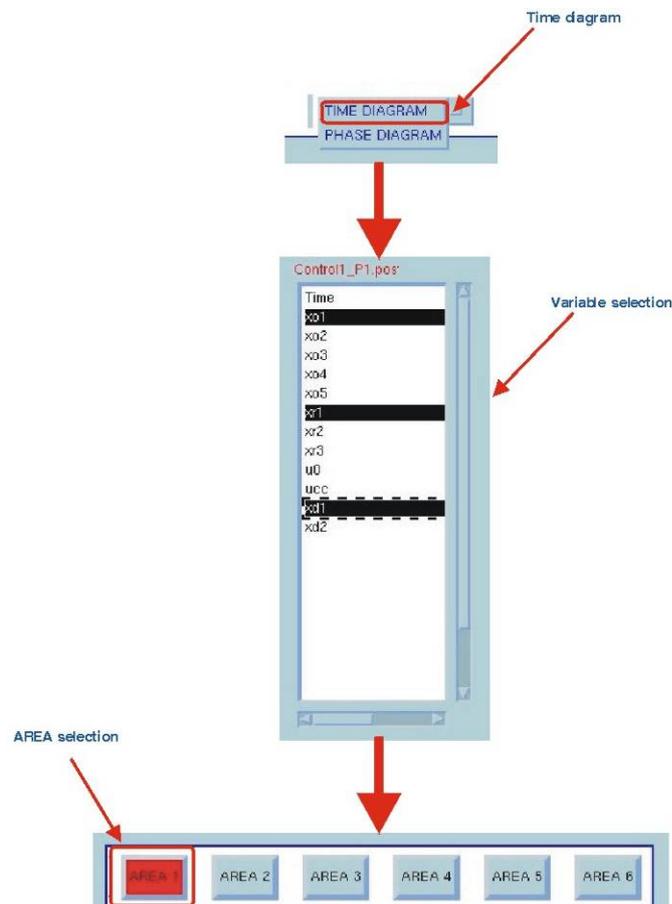
The customisation of every plotting area depends on the diagram you want to display.

Time diagram customisation

If you want to customise a plotting area displaying a time diagram, you need to:

- set the **TIME DIAGRAM** in the *Selection panel*,
- select up to 10 variables among the *Files variables lists* and the *New variables list* (in each area up to 10 time diagrams can be displayed);
- press the **AREA #** button where you want to visualise the graphs (the **AREA #** button will become coloured in red).

In this way the plotting area you select will display up to 10 graphs with the same abscissa (time) but with different ordinate (each selected variable is considered as an ordinate).

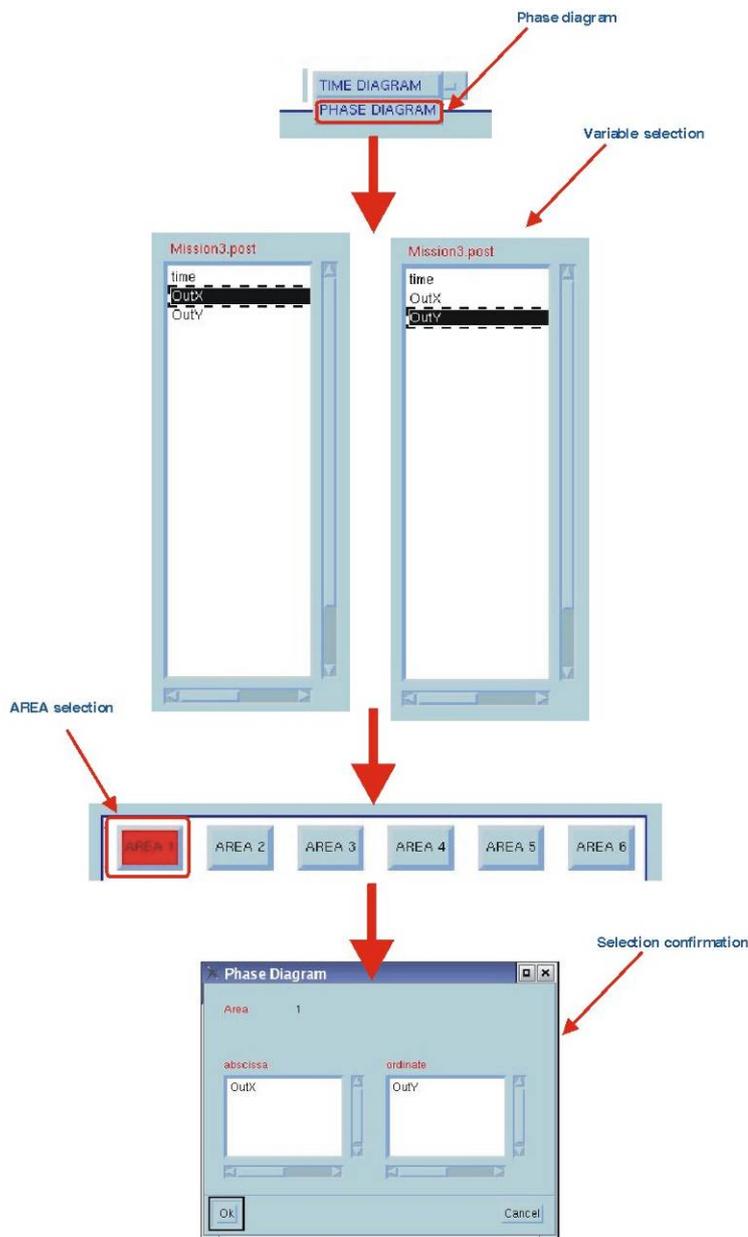


Phase diagram customisation

If you want to customise a plotting area displaying a phase diagram, you need to:

- set the **PHASE DIAGRAM** in the *Selection panel*;
- select, for each graph you want to visualise (up to 10 graphs) two variables among the *Files variables lists* and the *New variables list*; the first variable is considered as an abscissa, the second one as an ordinate. For example if you want to select three phase diagrams you need to select three couples of abscissa/ordinate variables;
- press the **AREA #** button where you want to visualise the graphs (the **AREA #** button will become coloured in red).

In this way the plotting area you select will display up to 10 graphs with different couples of abscissa/ordinate variables (the variables you selected).



Mathematical buttons

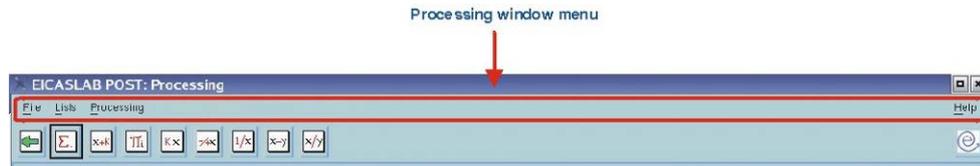
This section lists mathematical buttons for implementing the most common elaborations.

Icon	Name	Description
	Quit	Close the Processing window and come back the Configuration window
	Addition	Create a new variable composed of the sum of the variables you selected
	Addition of a scalar	Create a new variable composed of the one you selected plus an offset
	Product	Create a new variable composed of the product of the two variables you selected
	Product for scalar	Create a new variable composed of the one you selected multiplied by a scalar
	Change of sign	Create a new variable composed of the one you selected multiplied by -1
	Inverse	Create a new variable composed of the inverse of the one you selected
	Difference	Create a new variable composed of the difference between the two variables you selected
	Division	Create a new variable composed of the division between the two variables you selected

You can also apply these operations by means of the *Processing window* menu.

Processing window menu

The upper section of the Processing window contains the menus which help you in managing your selection and in performing post-processing of your variables.



In the following only items available in EICASLAB DEMO will be considered.

File menu

Item	Description
Save configuration	Allow you to save a configuration of the plotting areas
Load configuration ...	Allow you to load a stored configuration of the plotting areas
Save new variables on file	Save the variables of the <i>New variables list</i> in a POST file
Delete new variables	Delete all the variables of the <i>New variables list</i>
Quit	Close the Processing window and comes back the Configuration window
Exit	Close the Processing window and comes back the Configuration window, but maintain all the variable in the <i>new variables</i> list and the configuration of the plotting areas
Close POST	Close POST

Lists menu

Item	Description
Search for name ...	Allow you to search for a variable in a POST file (you have just to write the name of the variable)
Search for number ...	Allow you to search for a variable in a POST file (you have to write the progressive number of the variable)
Cancel selection	Deselect all the selected variables

Processing menu

Item	Description
Addition	Create a new variable composed of the sum of the variables you selected
Addition of a scalar	Create a new variable composed of the one you selected plus an offset
Product	Create a new variable composed of the product of

	the two variables you selected
Product for scalar	Create a new variable composed of the one you selected multiplied by a scalar
Change the sign	Create a new variable composed of the one you selected multiplied by -1
Inverse	Create a new variable composed of the inverse of the one you selected
Difference	Create a new variable composed of the difference between the two variables you selected
Division	Create a new variable composed of the division between the two variables you selected
Sin	Create a new variable applying the sine function to the variable you selected
Cos	Create a new variable applying the cosine function to the variable you selected
Tan	Create a new variable applying the tangent function to the variable you selected
Arcsin	Create a new variable applying the arcsine function to the variable you selected
Arccos	Create a new variable applying the arccosine function to the variable you selected
Arctan	Create a new variable applying the arctangent function to the variable you selected
Exp	Create a new variable applying the exponential function to the variable you selected
Raising to a power: x^n	Create a new variable applying x^n function to the variable you selected
Power of a scalar n : n^x	Create a new variable applying n^x function to the variable you selected
Power: x^y	Create a new variable applying x^y function to the variable you selected
Natural logarithm	Create a new variable applying the natural logarithm function to the variable you selected
Base 10 logarithm	Create a new variable applying the logarithm to the basis 10 to the variable you selected

Help menu

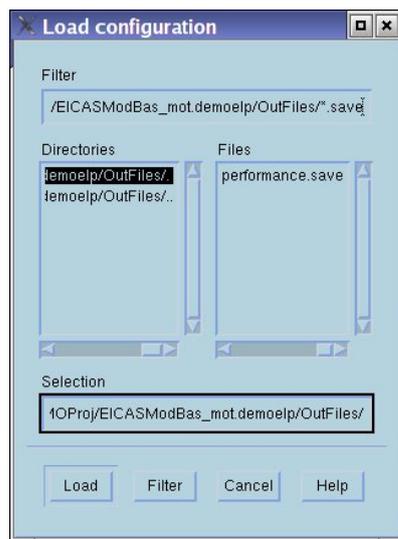
Item	Description
User Manual	Run the EICASLAB DEMO User Manual
EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com
About EICASLAB	Display some information about version and licence of EICASLAB DEMO

Configuration for locked projects

For locked projects (belonging to *PID*, *Classical Model Based* and *EICAS Model Based* families) EICASLAB DEMO provides a configuration file able to set all the plotting areas according to what displayed during the simulation. You has to use:

- **File -> Load Configuration** item

and select the `performance.save` file.



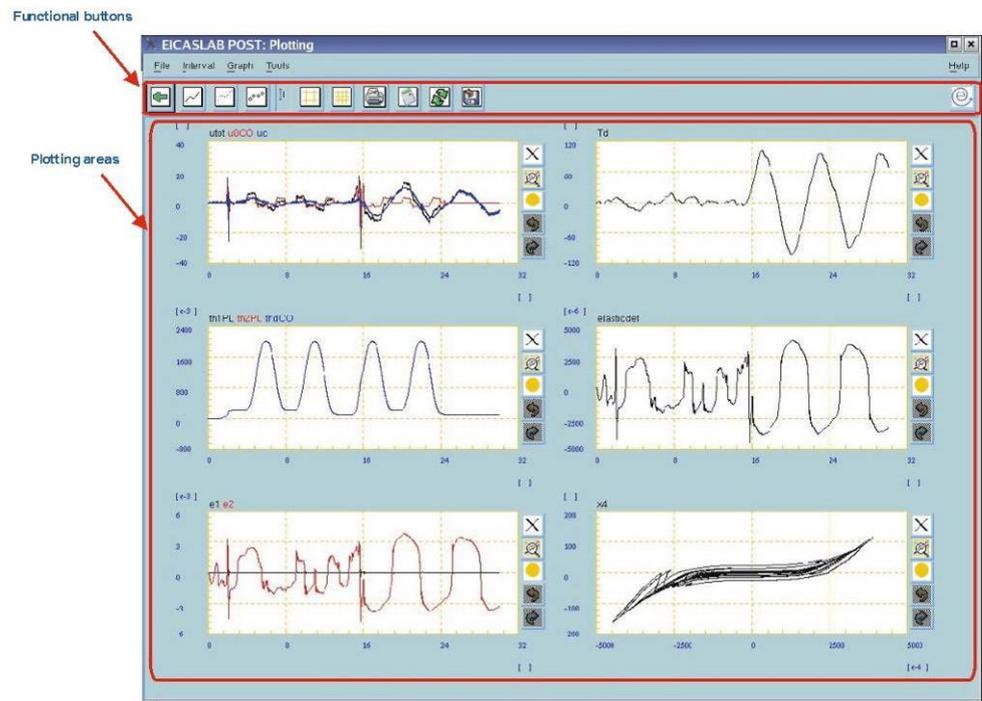
Plotting window

The Plotting window is the window where you can visualise and analyse the graphs related to the variables elaborated in the Processing window.

The Plotting window can be subdivided in two main sections:

- *Plotting areas*, where EICASLAB displays your graphs;
- *Functions buttons*, useful for analysing the graphs.

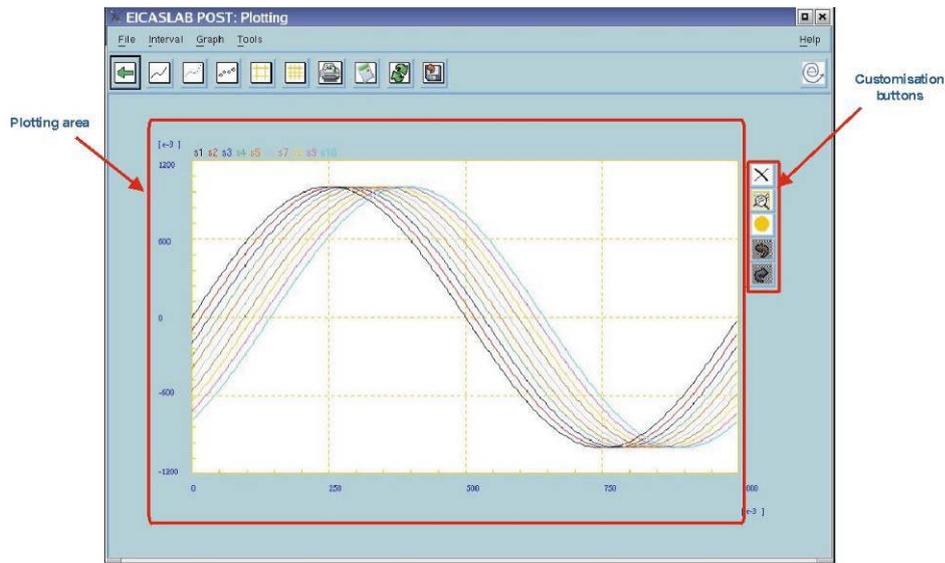
Additional options are available by using the *Plotting window* menu.



Plotting areas

Plotting areas is the section of the Plotting window where the graphs you selected in the Processing window are displayed. Up to 6 six plotting areas can be displayed; each one can contain up to ten graphs. The colours used for the graphs are:

Graph number	Colour
1	Black
2	Red
3	Blue
4	Green
5	Crimson
6	Violet
7	Brown
8	Yellow
9	Pink
10	Cyan



You can customise each area by using the customisation buttons available at the right of each area.

Icon	Name	Description
	Hide area	Hide the area you selected
	Display area	Hide all the other areas, in this way the Plotting window displays only the area you selected
	View all areas	Display all the original area, selecting in the Processing window
	Undo	Erase the last change done and revert the graph to an older state
	Redo	Reverse the undo changes

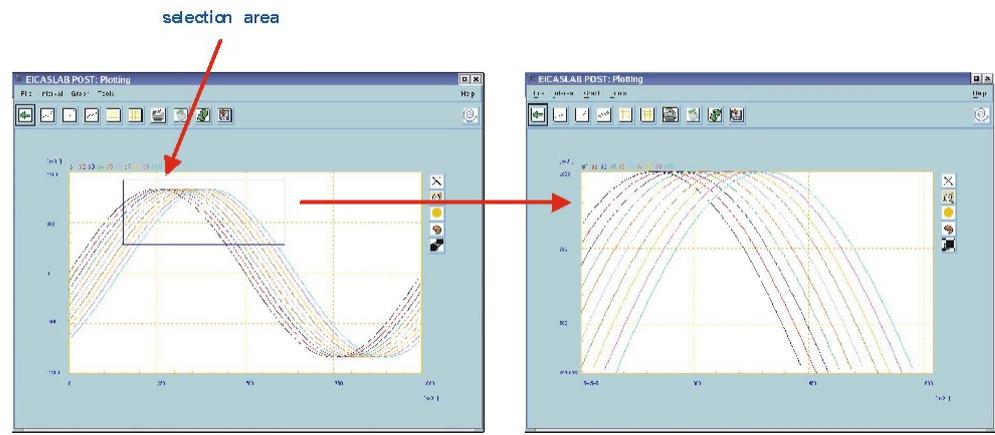
Managing areas

You can customise your areas by using the *Plotting window* menu or by working with the mouse.

For EICASLAB it is suggested to use a three buttons mouse; here the description of the purposes of each button is presented.

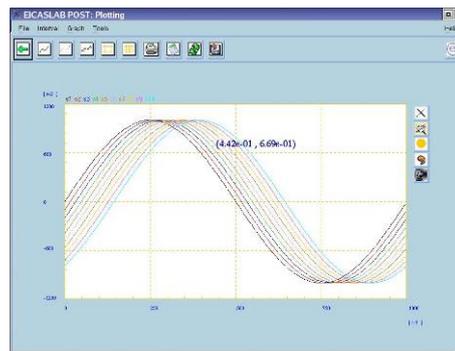
Left button

You can zoom on your graphs by clicking and moving the mouse. A blue rectangular will show the zone you selected for the zoom. The scale of each axis will be automatically recalculated.



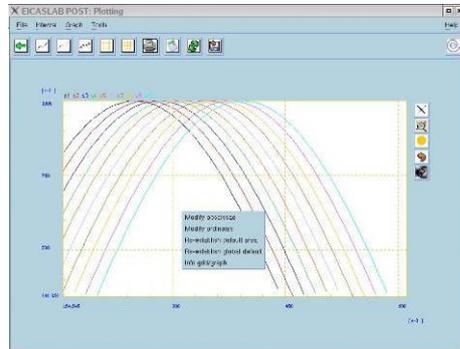
Central button

One click on a point in the area to view its X-Y coordinates.



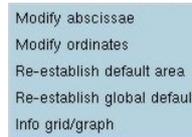
Right button

One click inside the area will show its popup menu.



Popup menu

You can enter the popup of a plotting area by right clicking on it.

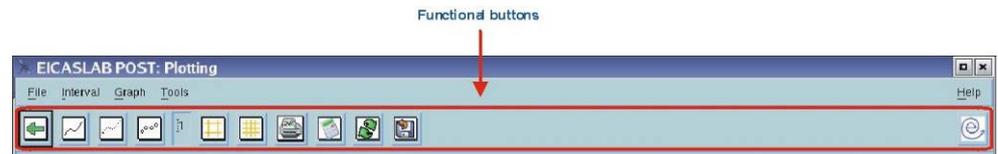


Item	Description
Modify abscissa	Allow to modify the range value of abscissas of the graphs in the selected area
Modify ordinate	Allow to modify the range value of ordinates of the graphs in the selected area
Re-establish default area	Re-establish the default range values of both ordinate and abscissa axes of the selected area
Re-establish global default	Re-establish the default range values of both ordinate and abscissa axes of all the areas
Info grid/graph	Provide information about the maximum and minimum values of the graphs in your area and the dimension of the grid

You can also apply these operations by means of the *Plotting window* menu.

Functional buttons

This section presents functional buttons to perform the most common graph customisation.



Icon	Name	Description
	Quit	Close the Plotting window and come back the Processing window
	Continuous graph	Display all the graphs in all the areas as lines
	Pointed graph	Display all the graphs in all the areas as points
	Pointed marked graph	Display all the graphs in all the areas as ball marks
	Line width	Set the width of the lines, point and ball marks. The range is from 1 to 9
	Default grid	Set the default grid for all the areas
	Dense grid	Set the dense grid for all the areas
	Print	Print the Plotting area in a printer or in a file
	Calculator	Run Linux calculator
	Refresh	Refresh the Plotting window
	Revert	Refresh the Plotting window by reading again the POST files. It is useful when you are working on POST while the simulation is still in progress
	EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com

You can also apply these operations by means of the *Plotting window* menu.

Plotting window menu

The upper section of the Plotting window contains the menus which help you in customising and analysing your graphs.



In the following only items available in EICASLAB DEMO will be considered.

File menu

Item	Description
Revert	Refresh the Plotting window by reading again the POST files. It is useful when you are working on POST while the simulation is still in progress
Refresh	Refresh the Plotting window
Prepare for printing	Allow to configure your graphs for technical reports
Print	Print the layout on a printer or on a file
Quit	Close the Plotting window and comes back the Processing window
Close POST	Close POST

Interval menu

Item	Description
Modify abscissa	Allow to modify the range value of abscissa of the graphs in the selected area
Modify ordinate	Allow to modify the range value of ordinate of the graphs in the selected area
Default values	Re-establish the default range values of both ordinate and abscissa axes of all the areas
Logarithmic scales	Allow to set logarithmic scales for the area you select

Graph menu

Item	Description
Continuous graph	Display all the graphs in all the areas as lines
Pointed graph	Display all the graphs in all the areas as points
Pointed marked graph	Display all the graphs in all the areas as ball marks
Type of grid	Set the default or the dense grid for all the areas

Info grid/graph	Provide information about the maximum and minimum values of the graphs in your area and the dimension of the grid
-----------------	---

Tools menu

Item	Description
Statistics	Allow you to perform static analysis of your graphs (not enabled in EICASLAB DEMO).
Calculator	Run Linux calculator

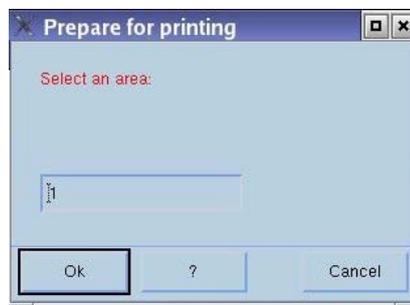
Help menu

Item	Description
User manual	Run the EICASLAB DEMO User Manual
EICASLAB on the Web	Run the EICASLAB home page www.eicaslab.com
About EICASLAB	Display some information about version and licence of EICASLAB DEMO

Prepare for printing

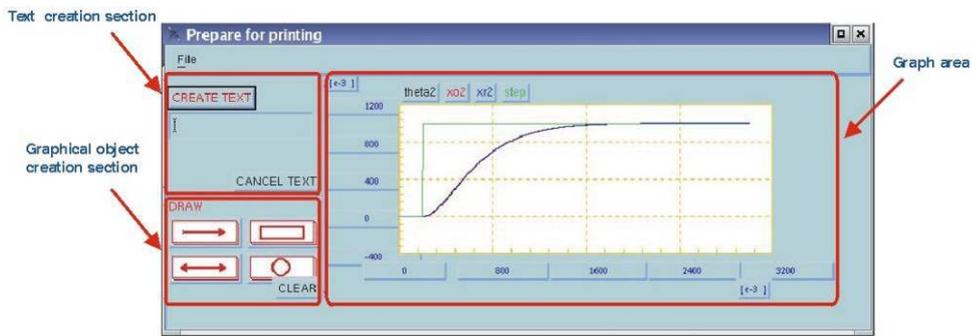
EICASLAB allows you to prepare graphs for technical reports. *Prepare for printing* item of the *File* menu allows to modify labels in your areas, insert new labels as titles, ordinate and abscissa axes, show significant points in your graphs by means of arrows and other graphical objects etc...

You have to select the area you want to customise.

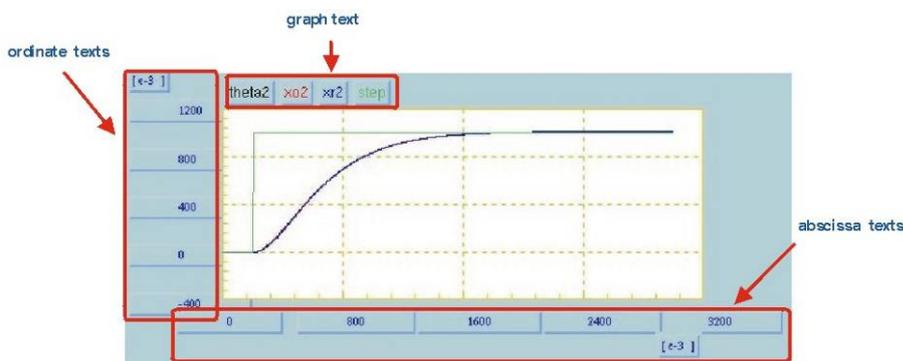


Then you can work on the Prepare for printing window. The window can be subdivided in three main zones:

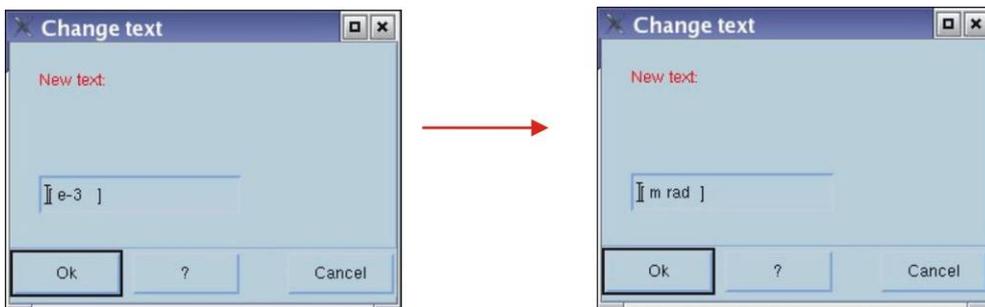
- *Graph area;*
- *Text creation section;*
- *Graphical object creation section.*



Graph area reports the area you want to customise. Here you can modify and insert labels in predefined places all around the graph.



You need only to click in the place where you want to change or insert the label, and write the new label in the text window.



You can:

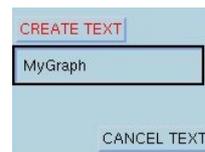
- Change the variables names
- Modify the ordinate and abscissa values
- Insert units of measurement



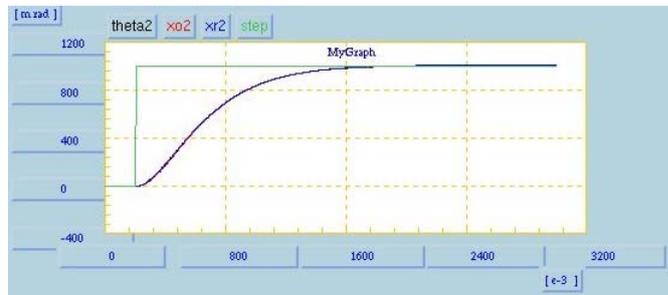
Text creation section allows you to create new labels to insert inside the area.



You need to write your text in the text section and press the **CREATE TEXT** button.

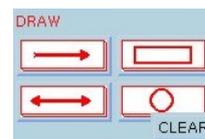


Then new label will appear when you move the mouse inside the area. Clicking inside the area (in the *Graph area*) will fix the text.



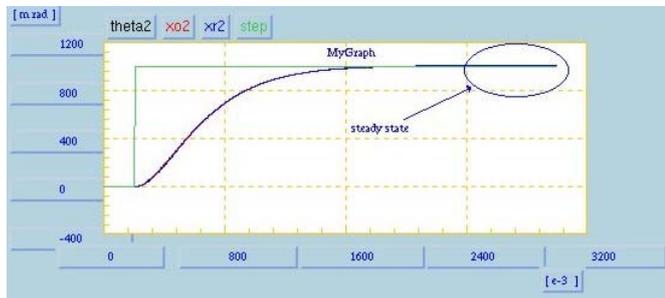
CANCEL TEXT deletes all the texts created in the area.

Graphical object creation section allows you to insert a graphical object in your area.

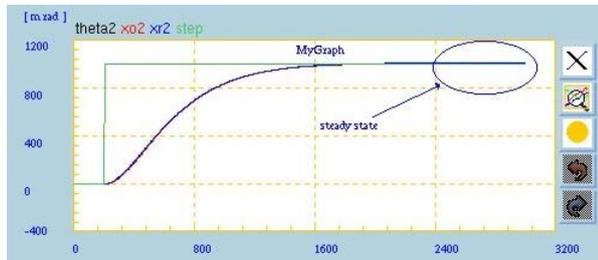


You can choose among: single arrows double arrow, rectangle, circle.

You need to click on the button related to the object you prefer. Then clicking and moving the mouse inside the area (in the *Graph area*) will fix your object.



The **CLEAR** button deletes all the graphical objects created in the area. When you finish to customise your area, you have to press **Exit** button in the Prepare for printing window, and your modifications will be applied to the graphs in the Plotting window.



Create your control algorithm and try it now!
 Enjoy your work with EICASLAB DEMO!



Appendix

I/O Control variables for User project

A complete description of the I/O variables belonging to the Control block of the *User* project is reported in the following tables.

Input

Variable	Unit	Description
th1CO	-	Quantized motor angular position; it is reported after mechanical gear
th2CO	-	Quantized rotating table angular position
thdCO	rad	Desired angular position
u0CO	A	Open loop command

Output

Variable	Unit	Description
comCO	-	Quantized total command

The variables th1CO and th2CO are respectively the quantized motor angular position and the quantized rotating table angular position. In order to be compared with the desired angular position thdCO, they have to be multiplied for the quantization level defined in the A/D converter.

The variable comCO is the command computed by the control algorithm divided by the quantization level defined in the D/A converter.

POST variables

A complete description of the variables available in the POST Processing window is reported in the following tables.

Rotating_Table.post

Variable	Unit	Description
Time	s	Time
comPL	A	Command
Td	N*m	Perturbing torque
th1PL	rad	Motor angular position; it is reported after the mechanical gear
th2PL	rad	Rotating table angular position
x0	rad/s	State related to motor angular rate
x1	Rad	State related to motor angular position
x2	Rad/s	State related to rotating table angular rate
x3	Rad	State related to rotating table angular position
x4	N*m	State related to hysteresis
totaldef	rad	Total elastic deformation of the mechanical gear: difference between the motor angular position (reported after the mechanical gear) and the rotating table angular position
elasticdef	rad	Effective elastic deformation of the mechanical gear
backlash	rad	Difference between totaldef and elasticdef

Trajectory_Generator.post

Variable	Unit	Description
time	s	Time
thdTG	rad	Desired angular position
u0TG	A	Open loop command
Se	rad	Commanded position sequence

Performance.post

Variable	Unit	Description
time	s	Time
th1PE	rad	Motor angular position; it is reported after mechanical gear
th2PE	rad	Rotating table angular position
thdPE	rad	Desired angular position
e1	rad	Error between desired and measured motor angular position; it is reported after the mechanical gear
e2	rad	error between desired and measured rotating

		table angular position
--	--	------------------------

PID_Control.post

Variable	Unit	Description
time	s	Time
th1CO	-	Quantized motor angular position; it is reported after mechanical gear
th2CO	-	Quantized rotating table angular position
thdCO	rad	Desired angular position
u0CO	A	Open loop command
comCO	-	Quantized total command
utot	A	Total command ($uc + u0CO$)
uc	A	Command generated by the PID control algorithm
yp	A/rad	Proportional contribute of the PID control algorithm
yi	A/s/rad	Integrative contribute of the PID control algorithm
yd	A*s/rad	Derivative contribute of the PID control algorithm
ex	rad	Error between desired and measured angular position. It is equal to $e1$ if the control algorithm uses the motor angular position measurement. It is equal to $e2$ if the control algorithm uses the rotating table angular position measurement.

Classical_Model-Based_Control.post

Variable	Unit	Description
time	s	Time
th1CO	-	Quantized motor angular position; it is reported after mechanical gear
th2CO	-	Quantized rotating table angular position
thdCO	rad	Desired angular position
u0CO	A	Open loop command
comCO	-	Quantized total command
utot	A	Total command ($uc + u0CO$)
uc	A	Command generated by the <i>Classical Model Based</i> control algorithm
ex	rad	Error between desired and measured angular position. It is equal to $e1$ if the control algorithm uses the motor angular position measurement. It is equal to $e2$ if the control algorithm uses the rotating table angular position measurement.
xo0	rad	State of the Luemberger observer. It estimates

		ex
xo1	Rad/s	State of the Luemberger observer. It estimates the derivative of ex

EICAS_Model-Based_Control.post

Variable	Unit	Description
time	s	Time
th1CO	-	Quantized motor angular position; it is reported after mechanical gear
th2CO	-	Quantized rotating table angular position
thdCO	rad	Desired angular position
u0CO	A	Open loop command
comCO	-	Quantized total command
utot	A	Total command ($uc + u0CO$)
uc	A	Command generated by the <i>EICAS Model Based</i> control algorithm