EICASLAB™ DEMO

The professional software suite
for automatic control design and forecasting

EICASLAB Demo RT-emb
User manual

Version 1.0

www.eicaslab.com

excellence and passion
in automatic control design

EICAS Automazione S.p.A.

How to contact EICAS Automazione S.p.A.

www.eicas.it	web site
info@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 RT-emb 1.0 User manual

Printing History: May 2016, first printing for EICASLAB Demo RT-emb 1.0 User manual

Preface

This document is the user manual of the EICASLAB Demo RT-emb (real-time demo for embedded system), indicated in the following as "the demo".

This manual contains specific information on the demo and a F.A.Q. section. The details about the full version of EICASLAB , its features and functions are available apart in its main documentation (EICASLAB User Manual) included also in the demo installation directory.

The application case and the models are the same ones of the other EICASLAB Demo about the rotating table application case. Please refer to the rotating table technical note available online to get more information about its model and control design considerations.

If you have more questions or need further assistance, you can contact the EICASLAB support team by email support@eicaslab.com.

Contents

Preface 1

Contents 2

Overview 4

Key Features *4*

Operative modes and tools *4*

Requirements 6

Requirements for the PC *6*

Requirements for the embedded system *7*

Operating system configuration *7*

Installation 10

Download *10*

Install *10*

Run the MASTER tool *11*

The DEMO menu *11*

Network configuration *12*

Open the project *13*

Modelling and Like Real-time Simulation 14

SIMBUILDER tool *14*

SIM tool *16*

POST tool *18*

Rapid Control Prototyping 22

RCP On Field sub-mode *23*

Slow Motion sub-mode *28*

Slow Motion tool *29*

Target operative mode.. 31

Automatic Code Generation for target.. 31

Hardware-in-the-loop sub-mode.. 35

HIL Manager tool.. 36

Final Validation Test sub-mode.. 37

TARGET Manager tool.. 39

The user defined control.. 42

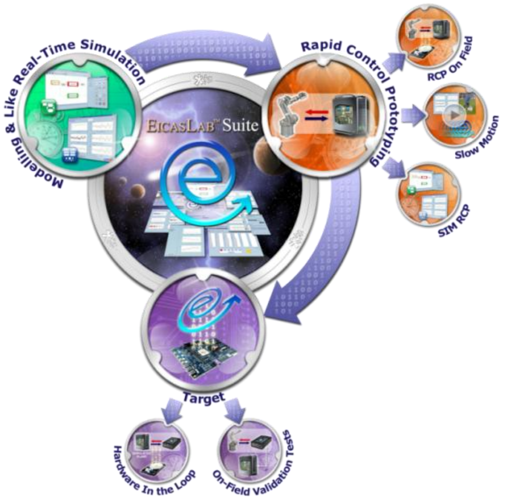
Frequently Asked Questions.. 44

I cannot find the target executable after the ACG for target. Where is it?.............. 44

Why I cannot run the RCP/HIL/TARGET Manager?... 44

Why I read an increasing number of errors and constant curves in the GUI?........ 44

Overview


Modelling & Like Real-Time Simulation
Rapid Control Prototyping
RCP On Field
Slow Motion
SIM RCP
EICASLAB Suite
Target
Hardware In the Loop
On-Field Validation Tests

The EICASLAB Demo RT-emb has been conceived to allow you to test all the operative modes of EICASLAB , namely "Modelling and Like Real-time Simulation", "Rapid Control Prototyping" and "Target", with **special focus on Target operative mode** (HIL and FVT sub-modes) with an external hardware target.

Two versions of the demo are available in order to support popular embedded targets: one is specific for the BeagleBone Black (BBB) and the other is for a generic board with Linux operating system, for example the Raspberry Pi.

Key Features

The demo highlights the following key features of EICASLAB . For more information about them, please visit the Key Feature pages on the EICASLAB website:

- one suite, one project for all design phases;
- real-time scheduling;
- automatic code generation;
- rapid control prototyping;
- slow motion;
- hardware-in-the-loop.

Operative modes and tools

The following table summarises the EICASLAB tools used in each operative mode and operative sub-mode in the demo. More information are available on the website pages How can EICASLAB help you and Tools overview.

Operative mode	**Operative sub-mode**	**Tools**
Modelling and Like Real-time Simulation	-	MASTER, SIMBUILDER, SIM, POST
Rapid Control Prototyping	RCP On Field	MASTER, SIMBUILDER, RCP Manager, POST
	Slow Motion	MASTER, SIMBUILDER, Slow

		Motion, POST
Target	Hardware-in-the-loop	MASTER, SIMBUILDER, HIL Manager, POST
	Final Validation Test	MASTER, SIMBUILDER, TARGET Manager, POST

The last three sections of the user manual provide details on functions and settings specific of the demo, which are a demonstrative subset of the EICASLAB software suite features.

Requirements

The execution of the demo in all the operative modes require hardware and software components listed in the table. Some entries are marked as part of the EICASLAB RCP Platform, i.e. a PC with a Linux Real Time Operating System (RTOS) and EICASLAB software suite. More information about it are available in the full EICASLAB user manual or on the website.

Type	Requirement	Part of the EICASLAB RCP Platform
HW	PC	Yes
	Embedded board	No
SW	RTOS	Yes
	EICASLAB suite	Yes
	EICASLAB projects	No

In order to experience good real-time performance on a standard PC and a Linux embedded board, the user has to check the following hardware and software requirements and configure properly the execution environment, as suggested in the following paragraphs.

Note

The real-time execution of the demo might have poor performance and provide a poor experience because of not suitable hardware or software configuration of the PC.

In that case, we suggest you to have a look at the F.A.Q. section of the document (section 5) or have a look at the video webinar available on EICASLAB website to watch the standard expected behaviour of the demo. If you have specific questions, send us a description of the problem at the e-mail address support@eicaslab.com.

Requirements for the PC

The demo requires a laptop or desktop PC with the following minimum features.

Hardware

- Dual core CPU
- 2 GB of RAM
- 1 GB of disk space

- Network interface[1]

Software

- GNU/Linux operating system (tested on Fedora 22, but can be any desktop Linux distribution, see the next paragraphs)
- PREEMPT_RT kernel configuration (recommended, see the next paragraphs)
- The software installer containing the three projects of the demo.

Requirements for the embedded system

The demo requires a Linux embedded board with the following features.

Hardware

- 512 MB of RAM
- 100 MB of storage space
- Network interface

Software

- GNU/Linux operating system
- SSH server enabled
- RT PREEMPT kernel configuration (mandatory, see the next paragraphs)

The demo was tested on the BeagleBone Black Rev. C with Ethernet-over-USB interface and the Raspberry Pi 3 Model B with Ethernet 10/100 interface.

Operating system configuration

The correct execution of EICASLAB software suite on Linux operating system requires a set of libraries that are usually pre-installed in most Linux distributions (Debian, Ubuntu, Red Hat, Fedora, etc.).

The demo requires a network connection to the external hardware (for HIL, FVT) and a virtual loopback network interface "lo" (for RCP): they are enabled by default on most systems and the user can check it with ifconfig command run in a terminal (i.e. konsole, gnome-terminal) or Network Manager system tool.

Note

Remember to check and optionally update the IP addresses on all the interfaces as explained in the "Network configuration" of the "Installation" chapter of this manual.

[1] Depending on the board interfaces. For example, the BeagleBone Black support Ethernet-over-USB so a USB2.0 port on PC side is enough. Other boards may have WiFi, Ethernet or other interfaces.

GNU/Linux distribution

If a Linux operating system is not installed on the PC, there are some options to run the demo (please check compatibility and warranty issues for the machine, before installing):

- install Linux on the disk drive in dual boot configuration with the operating system currently installed;
- install Linux on a removable drive, if the system supports the boot from an external device.

Fully RT PREEMPT kernel configuration

Real-time execution means the scheduling of some programs and threads must be completed in deterministic way, with limited latency and other timing requirements which are not compatible with the scheduling settings of most general purpose systems.

This EICASLAB demo is designed to work in Linux environment, but since Linux is a general purpose operating system, we recommend to install a RT PREEMPT Linux kernel to get better real-time execution performance. For more details about the RT PREEMPT patch of the Linux kernel, please read the rt.wiki.kernel.org website.

Some Linux distributions provide a set of pre-compiled ready-to-use Linux kernels to the users by means of their software package repositories: using the right packages the users can get quickly and without effort a low latency version of the kernel with real-time configuration. There is a set of alternative Linux kernel versions in each distribution, some of them include real-time versions (usually they are labelled with -rt). Next paragraphs consider different cases.

On the BeagleBone Black

The Debian 7.9 image for BeagleBone Black includes in its repository some RT PREEMPT versions of kernel. We recommend the version labelled 4.1.20-ti-rt-r58. A possible script to install it is the following (to be executed as root):

```
apt-get update
apt-get install linux-image-4.1.20-ti-rt-r58
reboot
```

On the Raspberry Pi 2 and 3

In the demo section on EICASLAB website, it is available a RT PREEMPT version of the Linux kernel for 32 bit Raspbian operating system for the popular Raspberry Pi 2 and 3 model B boards. That system is configured starting from a Raspbian image but the kernel is Linux raspberrypi 4.1.21-rt22-v7+ #1 SMP PREEMPT RT and the root user password is set to “raspberry”, the same as the standard “pi” user.

The following instructions to be executed on the Linux PC allow to copy the pre-configured RTOS to a microSD card (minimum size 16 GB), supposing it is the `/dev/sde` in the file system (please check it with `"dmesg|tail"` and `"fdisk -l"` commands to avoid losing data from your drives).

```
su - root
wget http://media.eicaslab.com/files/rpi_rt_sd16GB_image.gz
gzip -dc rpi_rt_sd16GB_image.gz | dd bs=4M of=/dev/sde
```

On the PC's and other embedded systems

If the software packet manager is apt, you can use the following commands to check the availability of Linux kernel versions.

```
apt-get update
apt-cache search linux-image
```

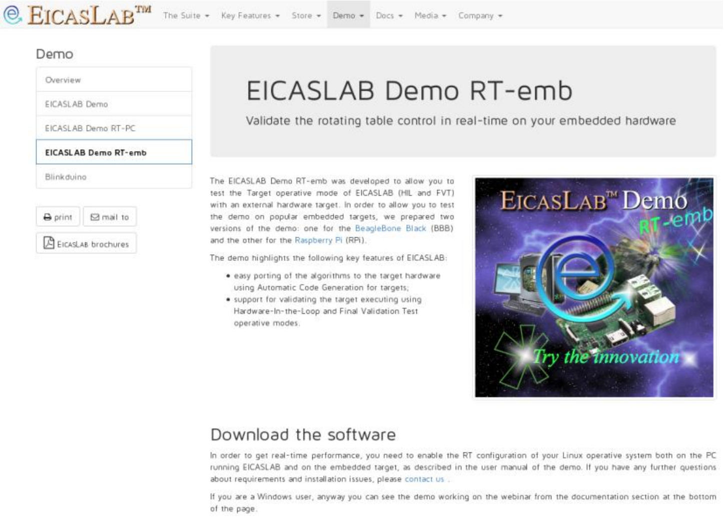
On the other hand, the general solution is to download the kernel source and the related rt-preempt patch from www.kernel.org, and build the patched kernel as explained step-by-step by the how-to page.

Installation

After checking the system requirements, the user must download and install the software, then the starting point of the demo execution is running the MASTER tool of EICASLAB as described in the following paragraphs.

Download

The download link for the demo installer is on the EICASLAB website.



EICASLAB Demo RT-emb

Validate the rotating table control in real-time on your embedded hardware

The EICASLAB Demo RT-emb was developed to allow you to test the Target operative mode of EICASLAB (HIL and FVT) with an external hardware target. In order to allow you to test the demo on popular embedded targets, we prepared two versions of the demo: one for the BeagleBone Black (BBB) and the other for the Raspberry Pi (RPi).

The demo highlights the following key features of EICASLAB:

- easy porting of the algorithms to the target hardware using Automatic Code Generation for targets;
- support for validating the target executing using Hardware-In-the-Loop and Final Validation Test operative modes.

Download the software

In order to get real-time performance, you need to enable the RT configuration of your Linux operative system both on the PC running EICASLAB and on the embedded target, as described in the user manual of the demo. If you have any further questions about requirements and installation issues, please contact us .

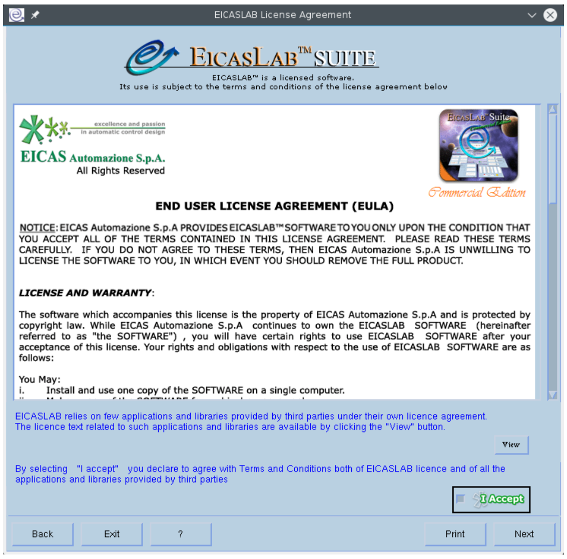
If you are a Windows user, anyway you can see the demo working on the webinar from the documentation section at the bottom of the page.

Moreover, in the page of the demo, the user can download all the documentation associated to the demo.

Install

To install the demo, first extract the demo installer archive and then run the *Install* executable.[2] The user has to agree to the license terms and optionally edit the install settings as shown by the installer program windows (installation directory and other preferences).

[2] In order to run the Install executable, open a terminal emulator in the EICASLAB installation folder containing the Install executable and type ./Install followed by <Enter> key.



EICASLAB License Agreement
EICASLAB™ SUITE

EICASLAB™ is a licensed software.
Its use is subject to the terms and conditions of the license agreement below

excellence and passion in automatic control design
EICAS Automazione S.p.A.

All Rights Reserved

EICASLAB Suite
Commercial Edition

END USER LICENSE AGREEMENT (EULA)

NOTICE: EICAS Automazione S.p.A PROVIDES EICASLAB™ SOFTWARE TO YOU ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS LICENSE AGREEMENT. PLEASE READ THESE TERMS CAREFULLY. IF YOU DO NOT AGREE TO THESE TERMS, THEN EICAS Automazione S.p.A IS UNWILLING TO LICENSE THE SOFTWARE TO YOU, IN WHICH EVENT YOU SHOULD REMOVE THE FULL PRODUCT.

LICENSE AND WARRANTY:

The software which accompanies this license is the property of EICAS Automazione S.p.A and is protected by copyright law. While EICAS Automazione S.p.A continues to own the EICASLAB SOFTWARE (hereinafter referred to as "the SOFTWARE") , you will have certain rights to use EICASLAB SOFTWARE after your acceptance of this license. Your rights and obligations with respect to the use of EICASLAB SOFTWARE are as follows:

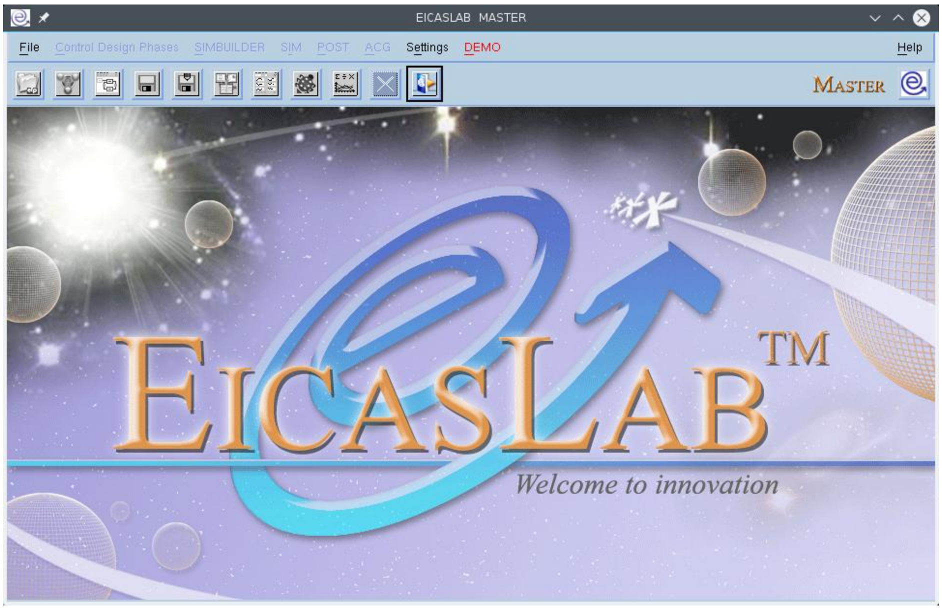
You May:
i. Install and use one copy of the SOFTWARE on a single computer.

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" button.

By selecting "I accept" you declare to agree with Terms and Conditions both of EICASLAB licence and of all the applications and libraries provided by third parties

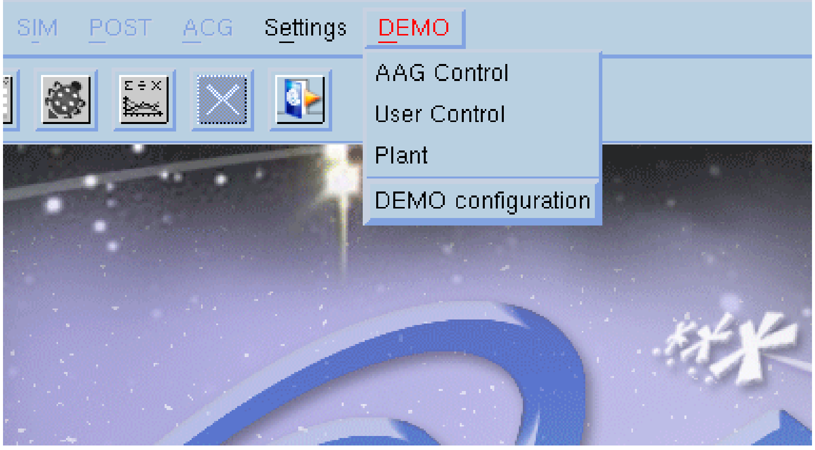
Run the MASTER tool

The entry point to work with EICASLAB software suite is the MASTER tool, which manages the whole projects and the other tools of the suite: therefore the first step to run the demo is to launch it from the desktop link or from the installation directory.



The DEMO menu

In the top menu of the MASTER tool window, select the menu DEMO and you can see the three projects included in the demo and the DEMO configuration menu.



- ***AAG Control*** project provides a control logic for the rotating table by Automatic Algorithm Generation. It is recommended to start with this project to easily become familiar with EICASLAB demo.
- ***User Control*** project allows the user to design his own control algorithm for the rotating table. We suggest you to use it after a full experience with the AAG version, as explained in the dedicated chapter "The user defined control".
- ***Plant*** project is a real-time emulator of the rotating table plant that is used by this demo instead of a real plant during RCP and FVT trials.
- ***DEMO Configuration*** item is available to edit the networking parameters of the demo as explained in the following paragraph.

Network configuration

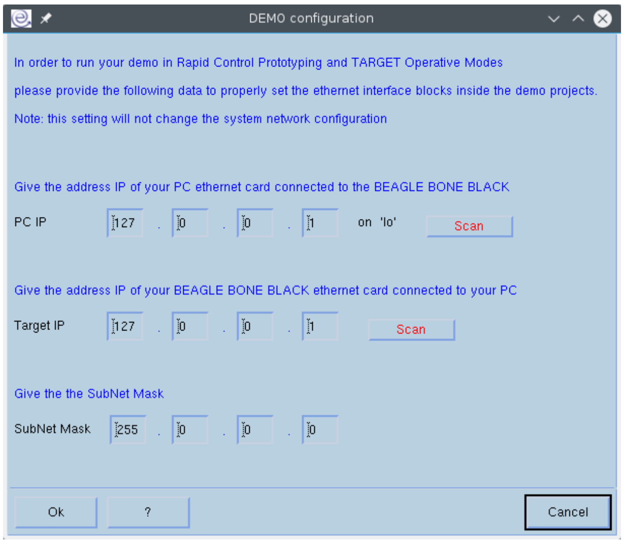
The use of the demo in Target operative mode requires some networking activities between the EICASLAB RCP Platform and the target embedded system, therefore the interface parameters must be adapted to suit your LAN configuration. The Beagle Bone Black board also allows you connecting to the board using a USB cable and then the standard Ethernet-over-USB network interface is enabled. So you can choose among the standard Ethernet interface or Ethernet-over-USB interface.

Note

To check the Ethernet-over-USB connection, you can use the "ifconfig" command in the command line: if the list of the interfaces includes a "usbX" or "enpYYYuZ" interface, then the driver is installed and you can try to ping the board with the command "ping 192.168.7.2".

In some cases, the OS recognizes the interface but does not enable it. In order to do that, you can use the command "ifconfig usbX 192.168.7.1 netmask 255.255.255.0 up".

If your board is in a LAN, you have to discover its IP address connecting to it in a suitable way and then you must use the DEMO Configuration function of the DEMO menu of the MASTER tool.



In order to run your demo in Rapid Control Prototyping and TARGET Operative Modes
please provide the following data to properly set the ethernet interface blocks inside the demo projects.
Note: this setting will not change the system network configuration

Give the address IP of your PC ethernet card connected to the BEAGLE BONE BLACK

PC IP 127 . 0 . 0 . 1 on 'lo' Scan

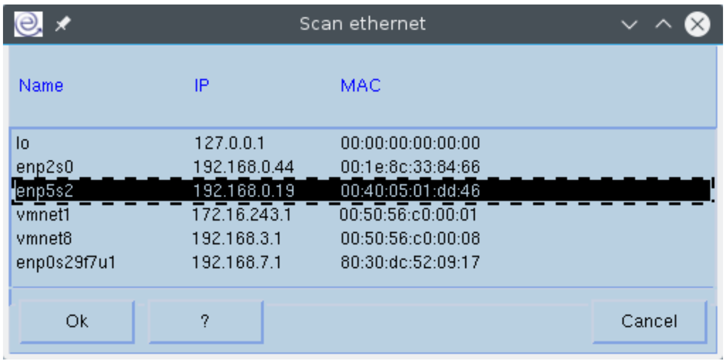
Give the address IP of your BEAGLE BONE BLACK ethernet card connected to your PC

Target IP 127 . 0 . 0 . 1 Scan

Give the the SubNet Mask

SubNet Mask 255 . 0 . 0 . 0

When you open the DEMO Configuration window[3], click on the first Scan button (the one related to the PC IP) to view the Ethernet interfaces available on the PC side and select the one sharing the same LAN of the board. By selecting it the corresponding Subnet Mask field of the window is automatically updated.



Name	IP	MAC
lo	127.0.0.1	00:00:00:00:00:00
enp2s0	192.168.0.44	00:1e:8c:33:84:66
enp5s2	192.168.0.19	00:40:05:01:dd:46
vmnet1	172.16.243.1	00:50:56:c0:00:01
vmnet8	192.168.3.1	00:50:56:c0:00:08
enp0s29f7u1	192.168.7.1	80:30:dc:52:09:17

Then click on the second Scan button (the one related to the Target IP) to discover the IP address of your board which has to match the subnet mask field.

Finally press the Ok button of the DEMO Configuration window and the demo will be automatically updated to suit your network configuration.

Open the project

Once the network settings are correctly set, click on the AAG Control project item from the DEMO menu to start to explore the demo.

Next sections help the user to cross step-by-step the whole sequence of the operative modes and the tools of EICASLAB available within this demo. Going through the workflow, the user can try the main features of the software suite, giving some insight to the full range of facilities provided by each of them.

[3] The first time you run the EICASLAB demo the DEMO Configurator window automatically gets opened.

Modelling and Like Real-time Simulation

Modelling and Like Real-Time Simulation mode allows to design and test the control algorithm and simulate the Rotating Table response to its commands.

The Modelling and Like Real-time Simulation operative mode enables the control designer to validate its algorithms in such a simulated environment using the following tools of the software suite: SIMBUILDER, SIM and POST.

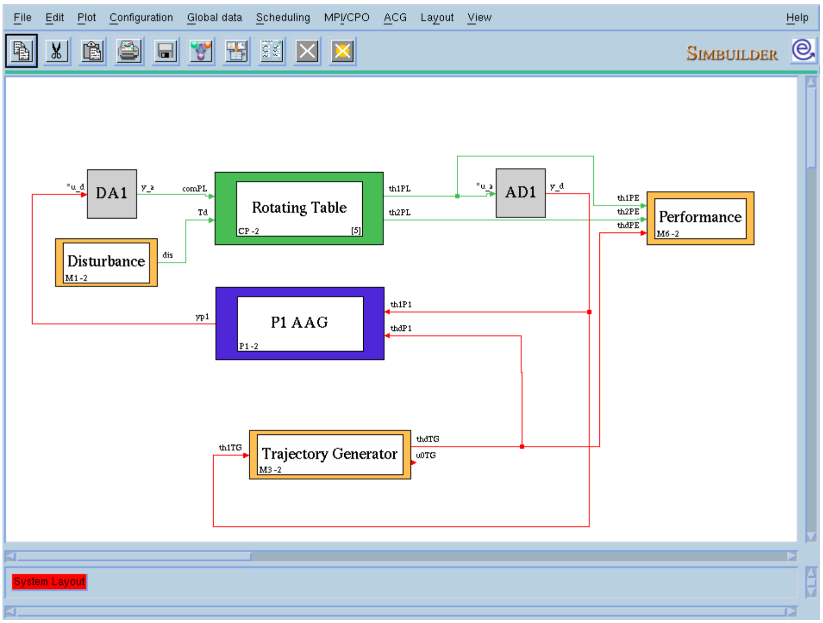
The following diagram summarises the roles of each element of the scenario in this first operative mode. You will find similar tables in each operative mode or sub-mode in the next chapters. The rotating table emulator is not the real hardware device, but it takes the place of the "real plant", thus it is included in the HW row of the table.

HW	EICASLAB RCP Platform	Embedded board target	Rotating Table emulator
SW	run control logic simulate the plant	unused	unused

SIMBUILDER tool

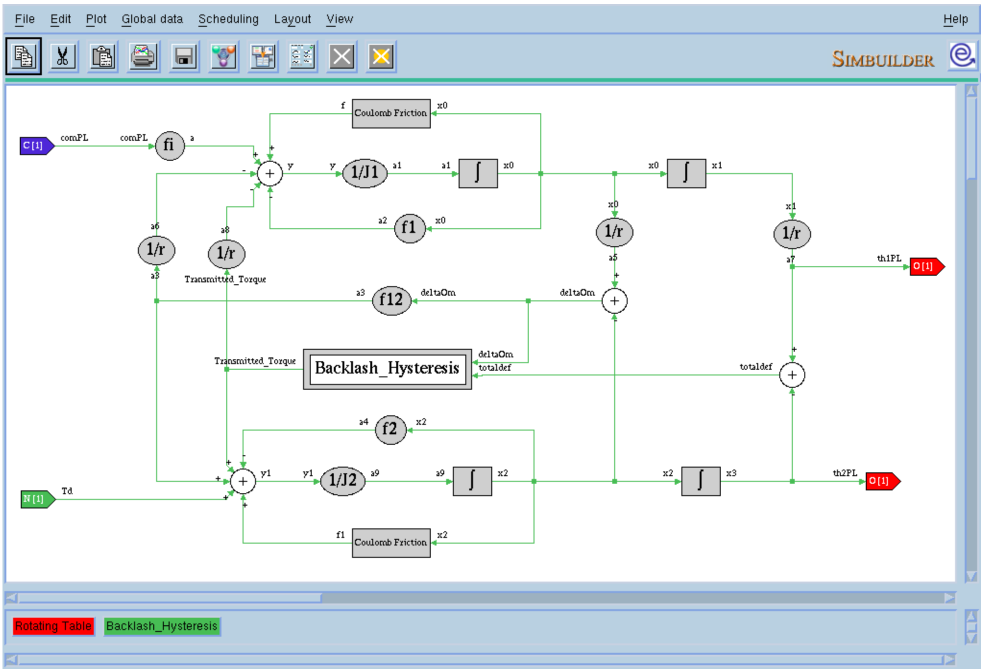
By default, the starting operative mode of the demo is Modelling and Like Real-time Simulation. The user can explore the layout double clicking on its elements.

Into the system layout of the demo there are examples of subsystems belonging to the three areas composing a typical control system:

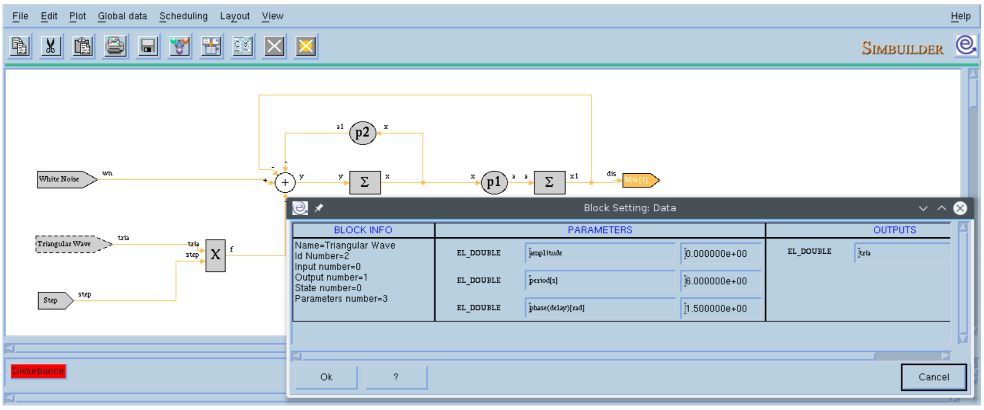


- the Rotating Table green block (Plant Area) is the fine model of the plant to be controlled in this demonstrative case
- the P1 AAG blue block (Control Area) is the control processor, containing the control algorithms under development
- the Trajectory Generator, Disturbance and Performance yellow blocks (Mission Area) are designated respectively for computing desired trajectory, modelling disturbances acting on the plant and computing performance evaluation;
- Analog-to-Digital (AD) and Digital-to-Analog (DA) grey interface blocks model the conversion from continuous signals to digital signals and vice versa. The plant works on continuous signals while the control manages digital signals.

The system layout is a hierarchical graphical representation: the Rotating Table plant block is described using several interconnected elements from the SIMBUILDER graphical block library.

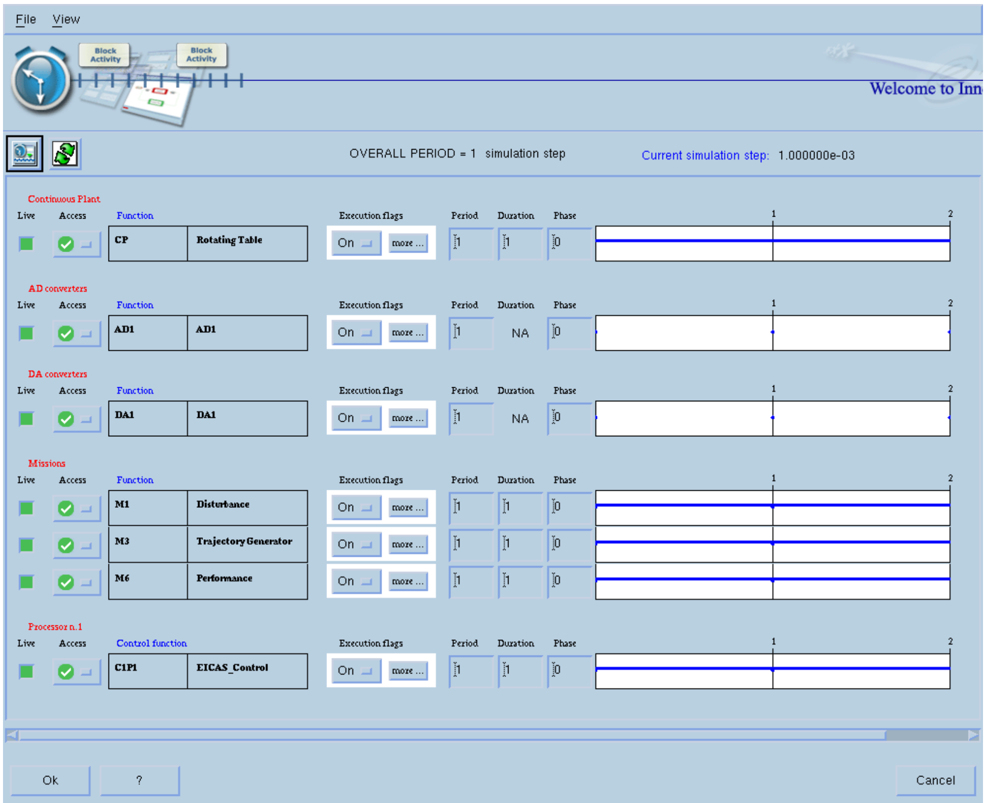


Each element of the graphical library has a set of parameters whose meaning is explained in the full EICASLAB user manual. In the demo it is possible to open the parameter window of each block.



BLOCK INFO	PARAMETERS			OUTPUTS	
Name=Triangular Wave Id Number=2 Input number=0 Output number=1 State number=0 Parameters number=3	EL_DOUBLE	amplitude	0.000000e+00	EL_DOUBLE	tria
	EL_DOUBLE	period[s]	6.000000e+00		
	EL_DOUBLE	phase(delay)[rad]	1.500000e+00		

The main parameters of activity scheduling for Like Real-time Simulation are collected in the Scheduling window, where the user is able to describe the timings of the complete system in terms of phases, periods and durations in order to emulate the real world.



OVERALL PERIOD = 1 simulation step

Current simulation step: 1.000000e-03

Continuous Plant

Live	Access	Function		Execution flags		Period	Duration	Phase
🟩	✓	CP	Rotating Table	On	more ...	1	1	0

AD converters

Live	Access	Function		Execution flags		Period	Duration	Phase
🟩	✓	AD1	AD1	On	more ...	1	NA	0

DA converters

Live	Access	Function		Execution flags		Period	Duration	Phase
🟩	✓	DA1	DA1	On	more ...	1	NA	0

Missions

Live	Access	Function		Execution flags		Period	Duration	Phase
🟩	✓	M1	Disturbance	On	more ...	1	1	0
🟩	✓	M3	Trajectory Generator	On	more ...	1	1	0
🟩	✓	M6	Performance	On	more ...	1	1	0

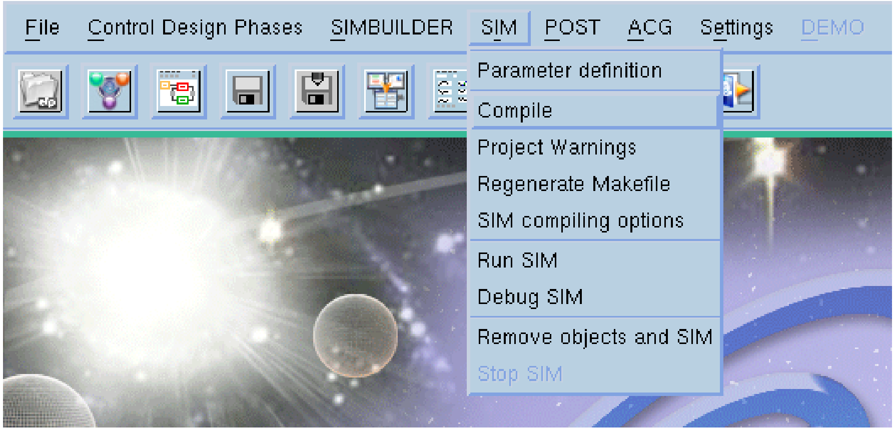
Processor n.1

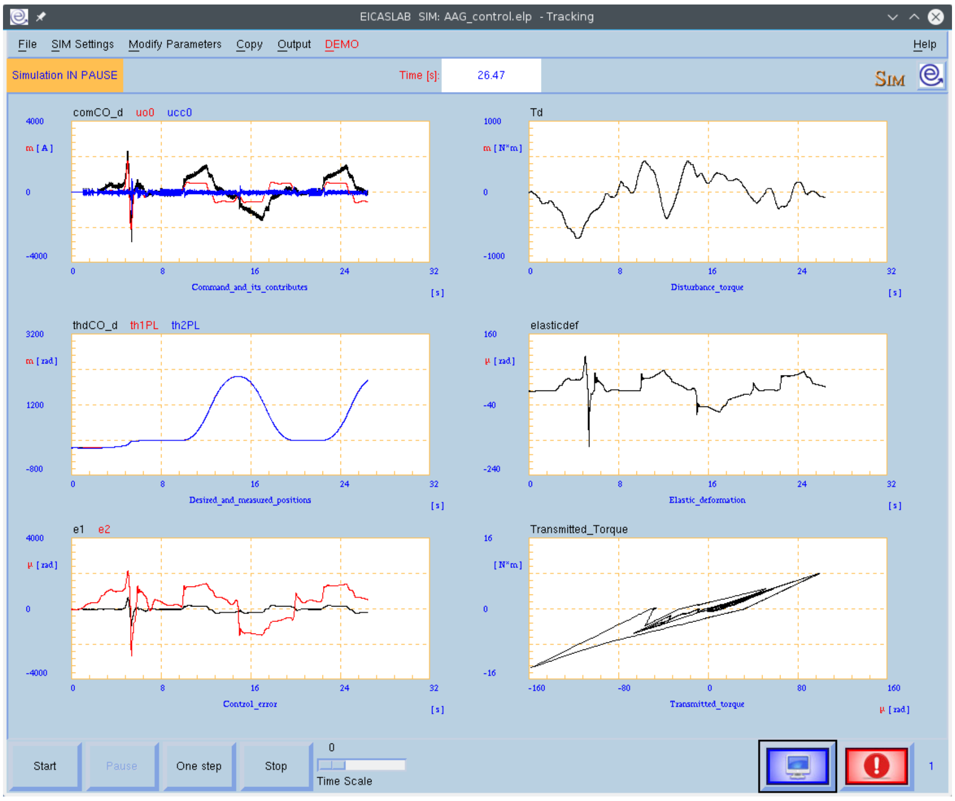
Live	Access	Control function		Execution flags		Period	Duration	Phase
🟩	✓	C1P1	EICAS_Control	On	more ...	1	1	0

SIM tool

The SIM tool is available in Modelling and Like Real-time Simulation operative mode. Its purpose is to validate the control described in the SIMBUILDER layout in a simulated environment.

Before running the SIM tool you have first compile the project by means of the compile step (SIMBUILDER menu SIM > Compile), therefore the simulation trials are managed by the SIM tool (SIMBUILDER menu SIM > Run SIM).





The SIM tool graphical user interface main section includes a set of time diagrams of quantities of interest, whereas its main controls are the Start and the Stop buttons in the left bottom corner. In the right bottom corner there are two useful log buttons (standard output and standard error), with the warning/error counter.

That graphical user interface configuration is common to SIM, RCP Manager, Slow Motion, HIL Manager and TARGET Manager in order to simplify the user experience even if the background execution is different from a tool to the other.

To execute a simulation trial and control the rotating table plant model, simply press the Start button in the SIM window. Some plots are displayed in real-time in the six graphical areas as an example, the quantities are listed in the following table.

In the SIM trial of the demo, some time diagrams representing preselected variables of interest are shown. The meaning of such variables is reported in the following table.

Plot area	Color	Name	Description
1 (Command and its contributes)	Black	comCO_d	Overall command (A)
	Red	uo0	Open loop command (A)
	Blue	ucc0	Closed loop command (A)
2 (Desired and measured	black	thdCO_d	Desired angular position (rad)
	Red	th1PL	Motor angular position (rad)[4]

[4] The motor angular position is reported on the load axis using the gear ratio.

positions)	Blue	th2PL	Load angular position (rad)
3 (Control error)	Black	e1	First error component (err)
	Red	e2	Second error component (err)
4	Black	Td	Disturbance torque (Nm)
5	Black	elasticdef	Elastic deformation (rad)
6	Black	Transmitted_torque	Transmitted torque (Nm)

POST tool

The POST is the tool to support data analysis for post-processing and it is launched by the MASTER main menu. All the trials recorded by the SIM and the other tools are in POST format, as described by the full EICASLAB user manual, so that the user could analyse and post-process them to keep on the control development.

The demo is set to record on disk a set of variables belonging to the following blocks:

- Rotating Table,
- Trajectory Generator,
- Disturbance,
- EICAS control,
- In1[5]

For each of these blocks there is a pre-defined selection of variables for POST recording[6] described in the following tables:

Rotating Table

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

[5] It is a *Target Input* block available only in the FVT operative mode.

[6] For all the demo projects except the User Control project.

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
Transmitted_Torque	N*m	Transmitted torque to the rotating table
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

Trajectory Generator

Variable	Unit	Description
time	S	Time
thdTG	Rad	Desired angular position
Se	Rad	Commanded position sequence

Performance

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

EICAS Control

Variable	**Unit**	**Description**
time	S	Time
th1CO_d	-	Motor angular position; it is reported after mechanical gear
thdCO_d	-	Desired angular position
comCO_d	-	Total command
uo0	A	Open loop command
ucc0	A	Closed loop command

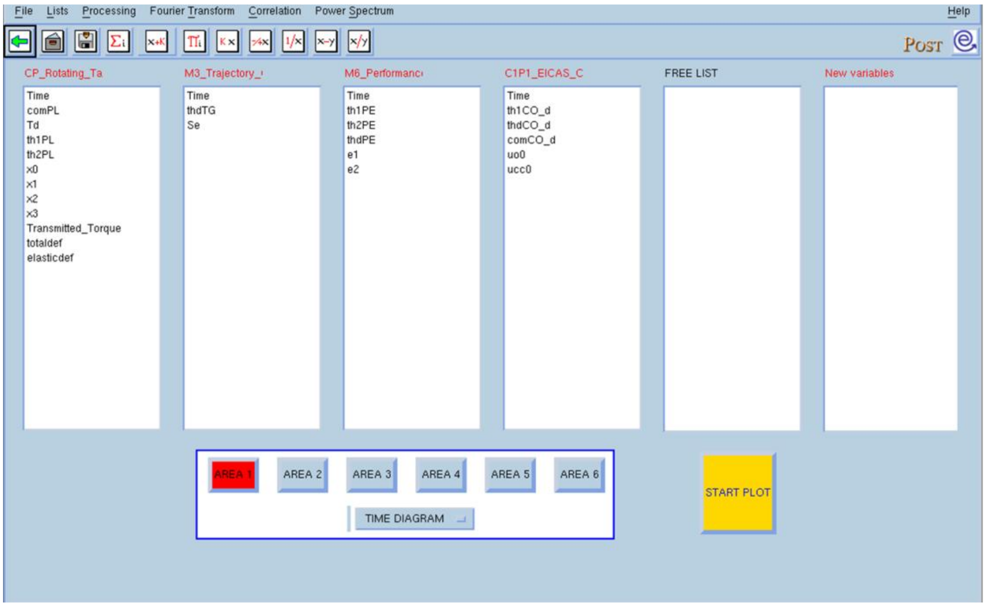
In1

Variable	**Unit**	**Description**
com1	A	Total command

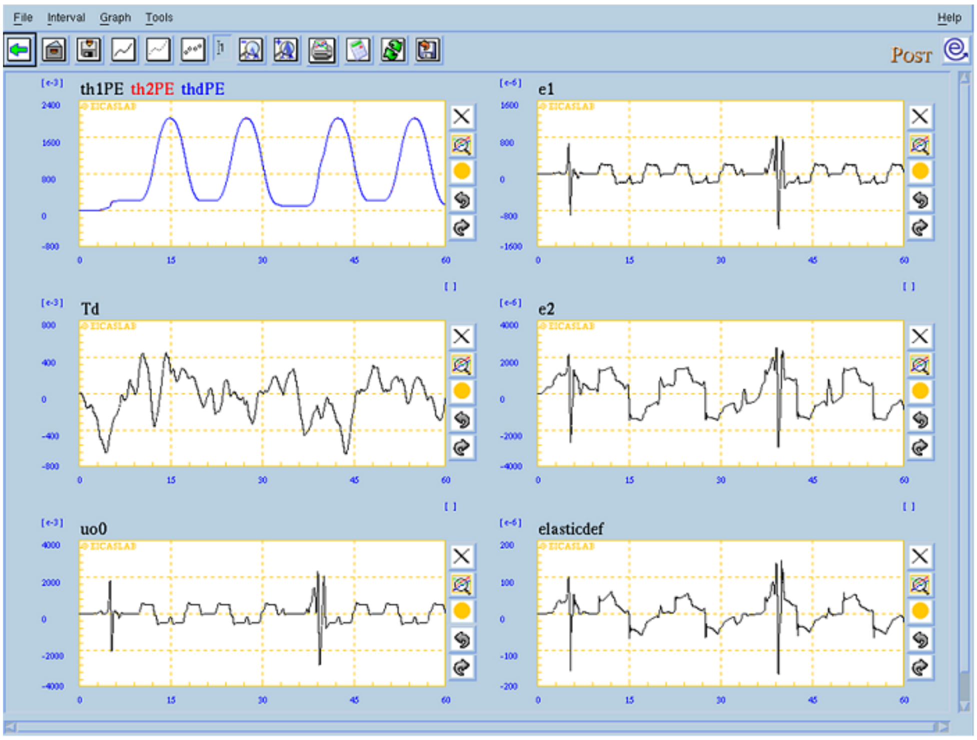
Note

You are free to change the pre-defined variable selection for POST recording selecting other variables as described in the full EICASLAB user manual.

Opening the POST tool its Processing Window (described in the full EICASLAB user manual) gets opened, listing all the recorded variables subdivided in columns depending on the block to which they belong.

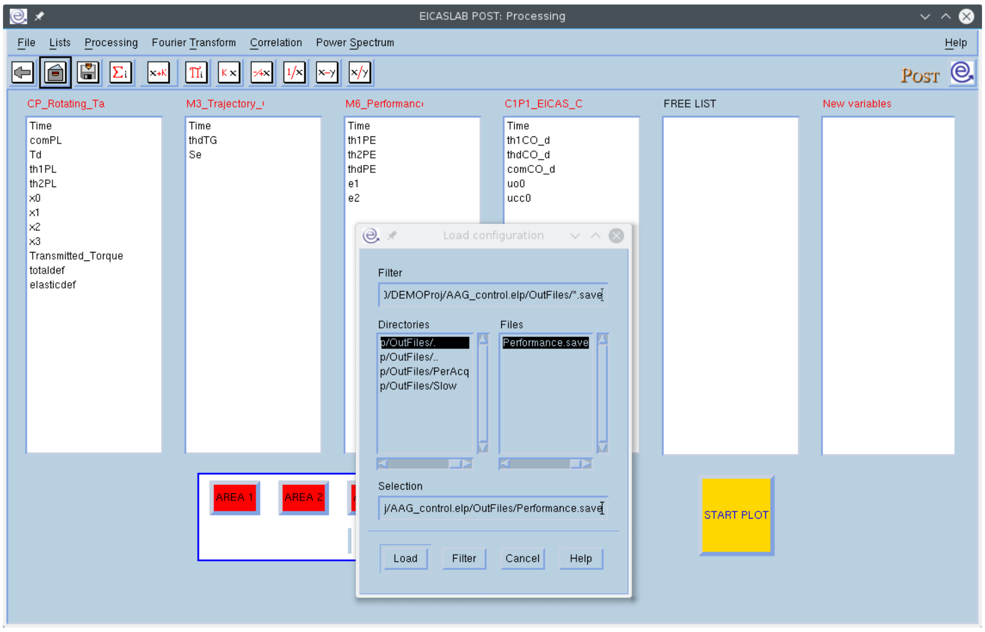


You can then select any variable indicating in which of the 6 available areas plotting it. Then pressing on the ***Start Plot*** button the Plotting window (described in the full EICASLAB user manual) gets opened representing all the graphs.



Recorded variable configurations

A pre-defined graph configuration is available: from the POST menu select File > Load configuration, click on "Performance.save" and press the Load button.



CP_Rotating_Ta: Time, comPL, Td, th1PL, th2PL, x0, x1, x2, x3, Transmitted_Torque, totaldef, elasticdef

M3_Trajectory_: Time, thdTG, Se

M6_Performanc: Time, th1PE, th2PE, thdPE, e1, e2

C1P1_EICAS_C: Time, th1CO_d, thdCO_d, comCO_d, uo0, ucc0

Filter:)/DEMOProj/AAG_control.elp/OutFiles/*.save

Directories: p/OutFiles/, p/OutFiles/.., p/OutFiles/PerAcq, p/OutFiles/Slow

Files: Performance.save

Selection: j/AAG_control.elp/OutFiles/Performance.save

After loading the configuration you can press on the ***Start Plot*** button for opening the Plotting window where all the graphs, representing a set of the available variables, are configured according to the selected configuration.

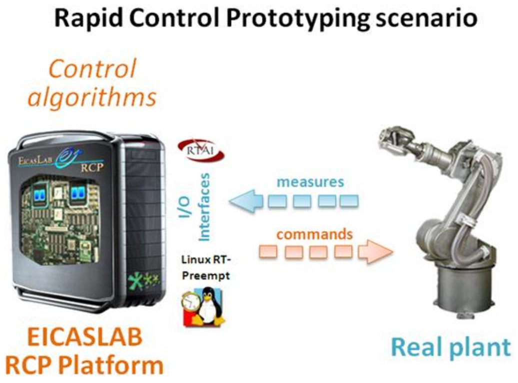
Note

Each operative mode has its variable configuration (always stored in a file *Performance*.save).

Rapid Control Prototyping

The Rapid Control Prototyping (RCP) operative mode of EICASLAB supports the control development in the mid-way step of porting the control algorithms from simulation environment to the real plant to be controlled. The EICASLAB RCP Platform[7] is the ideal instrument to analyse, tune and debug the control logic executing in real world before using the final target to run the control program.

[7] The EICASLAB RCP Platform is a standard PC with Linux RTOS (RT PREEMPT or RTAI), EICASLAB software suite and suitable hardware interfaces. That configuration allows to save cost and efforts by covering all the design phases on a single standard PC. Read more on the EICASLAB website.



The demo allows you to experience the two operative sub-modes of the RCP operative mode and their specific tools, i.e. the RCP Manager for the RPC On Field sub-mode and the Slow Motion tool for the Slow Motion sub-mode, as explained in the next two paragraphs. The following diagrams show the roles of the elements of the demo in RCP On Field and Slow Motion, respectively.

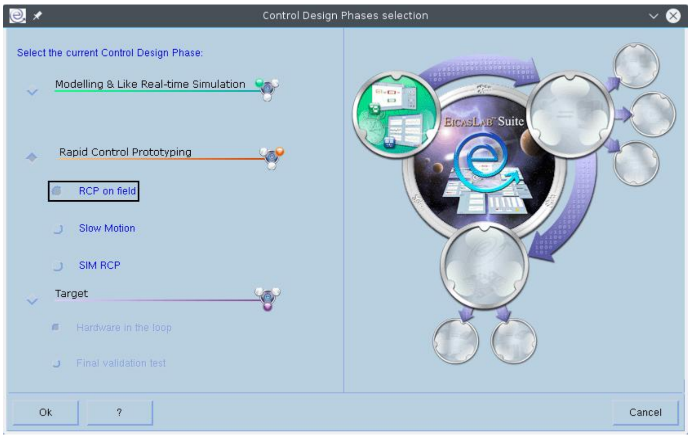
HW	EICASLAB RCP Platform	Embedded board target	Rotating Table emulator
SW	run control logic	unused	simulated by second project

HW	EICASLAB RCP Platform	Embedded board target	Rotating Table emulator
SW	run control logic	unused	unused

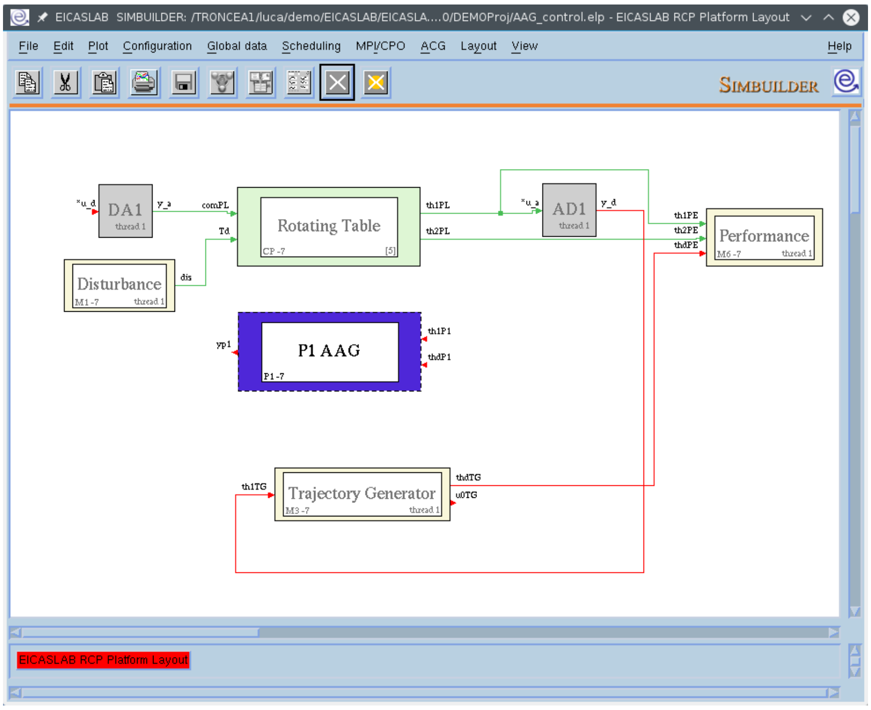
simulated plant from recorded data

RCP On Field sub-mode

Close the SIM tool and switch EICASLAB to RCP On Field sub-mode using the Control Design Phases > Operative Mode Selection item of the menu bar.

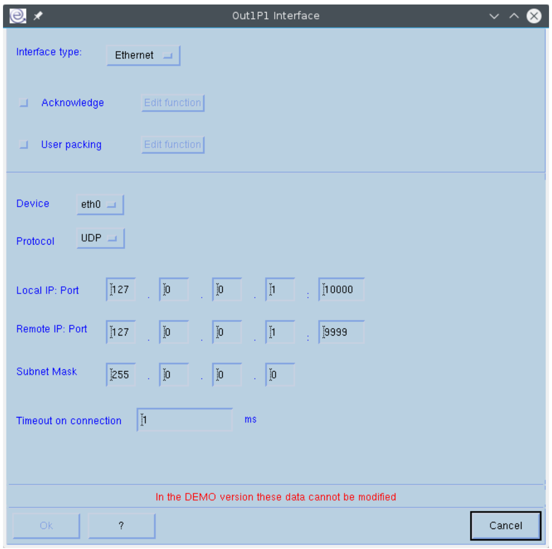

Control Design Phases selection
Select the current Control Design Phase:
Modelling & Like Real-time Simulation
Rapid Control Prototyping
RCP on field
Slow Motion
SIM RCP
Target
Hardware in the loop
Final validation test
EICASLAB Suite
Ok
?
Cancel

The SIMBUILDER will load the system layout of the RCP On Field configuration and the user can explore it and its parameters with the SIMBUILDER tool as done in Modelling and Like Real-time Simulation operative mode.



The demo shows the typical RCP On Field system layout in the SIMBUILDER: the Plant Area and Mission Area are disabled, since the control algorithms should be validated on the real plant connected to the EICASLAB RCP Platform by suitable interfaces.

In RCP On Field, internal layout of the Processor has real-time interfaces for real networking: the software managing the interface is automatically generated by EICASLAB to reduce the development work load, on the basis of a set of parameters (e.g. the IP address, port and timeout for the network interface).



Out1P1 Interface

Interface type: Ethernet

□ Acknowledge Edit function

□ User packing Edit function

Device eth0

Protocol UDP

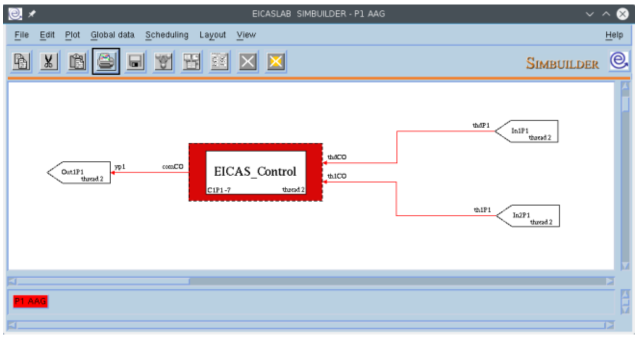
Local IP: Port 127 . 0 . 0 . 1 : 10000

Remote IP: Port 127 . 0 . 0 . 1 : 9999

Subnet Mask 255 . 0 . 0 . 0

Timeout on connection 1 ms

In the DEMO version these data cannot be modified



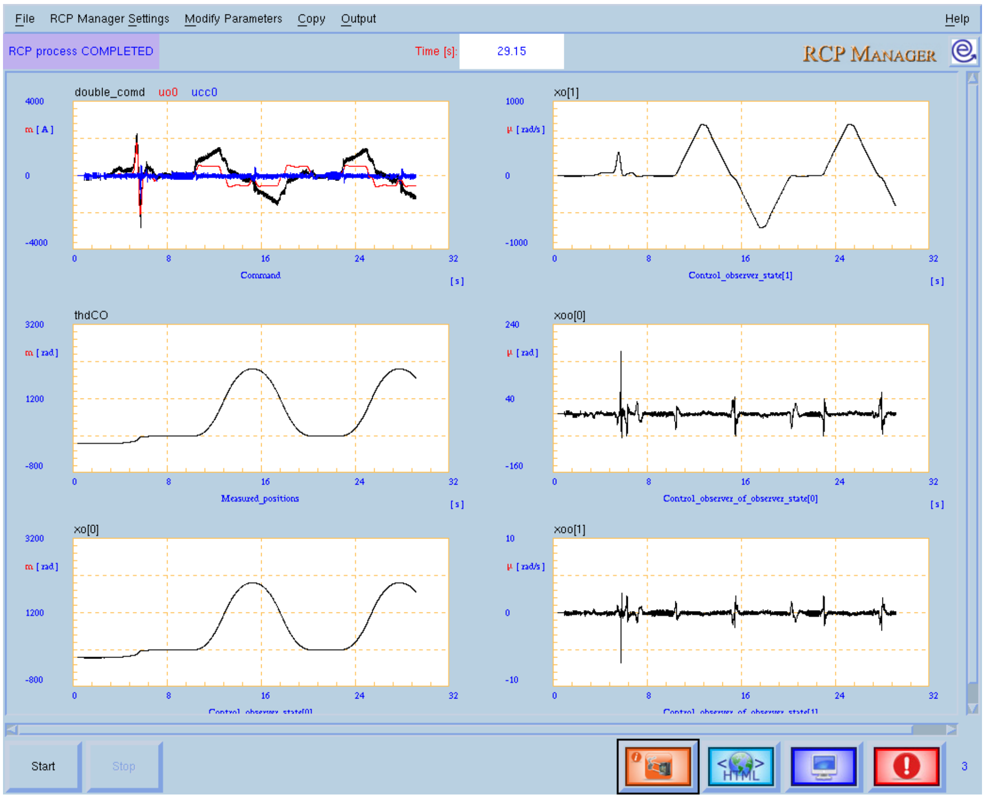
Note

The local communication between the plant and the control real-time applications passes through the loopback network interface “lo” which by default has IP address 127.0.0.1 and netmask 255.0.0.0 values. To check or edit the configuration, Network Manager or ifconfig tool could be used.

RCP Manager tool

The RCP Manager tool controls the execution of RCP On Field trials. To run the real-time trial, first compile the real-time application from RCP Manager > Compile (or the tool bar Compile button) and run the RCP Manager tool[8] from RCP Manager > Run RCP Manager or the Run button of tool bar, in a similar way as done in Modelling and Like Real-time Simulation operative mode to run the SIM tool.

[8] In order to access real-time capabilities the RCP Manager tool must be run as superuser. The EICASLAB installation procedure creates two links on the user desktop: the first one allows you to run EICASLAB as user while the other one allows you to run EICASLAB as superuser. In the latter case you will be asked to type the superuser password. Another way to run EICASLAB as superuser is to open a terminal, log in as superuser and run EICASLAB.



Note

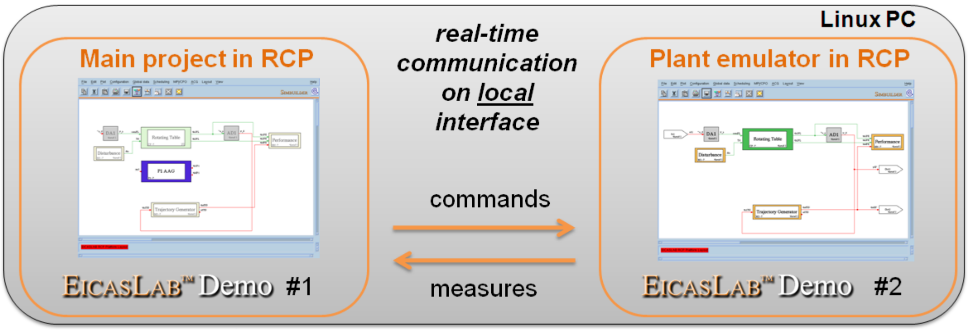
Even if the GUI looks like the one of the SIM, they are completely different behind the scenes: the RCP Manager tool is a real-time program with a multi-core and multi-thread scheduling configured by the user in SIMBUILDER and with automatic code generation for managing the interfaces with the real plant staying in the real world outside the EICASLAB RCP Platform.

In the specific case of this demo, the control project is designed to show the EICASLAB features without the need to purchase any hardware, therefore the “real” plant is emulated by a second project included in the demo (Plant project) to be executed in a second separated instance of EICASLAB program as described in the following “Plant real-time emulation for the demo in RCP” paragraph.

Plant real-time emulation for the demo in RCP

This demo is designed to avoid the need for real rotating table connected to the EICASLAB RCP Platform, so the user has to launch the rotating table emulator available in the EICASLAB demo suite with the following steps:

1. launch as superuser another instance of the MASTER tool in the same way as you have already done for the first one in “Run the MASTER tool” paragraph;
2. open the Plant project from the DEMO menu;
3. compile and run the RCP Manager.



Note

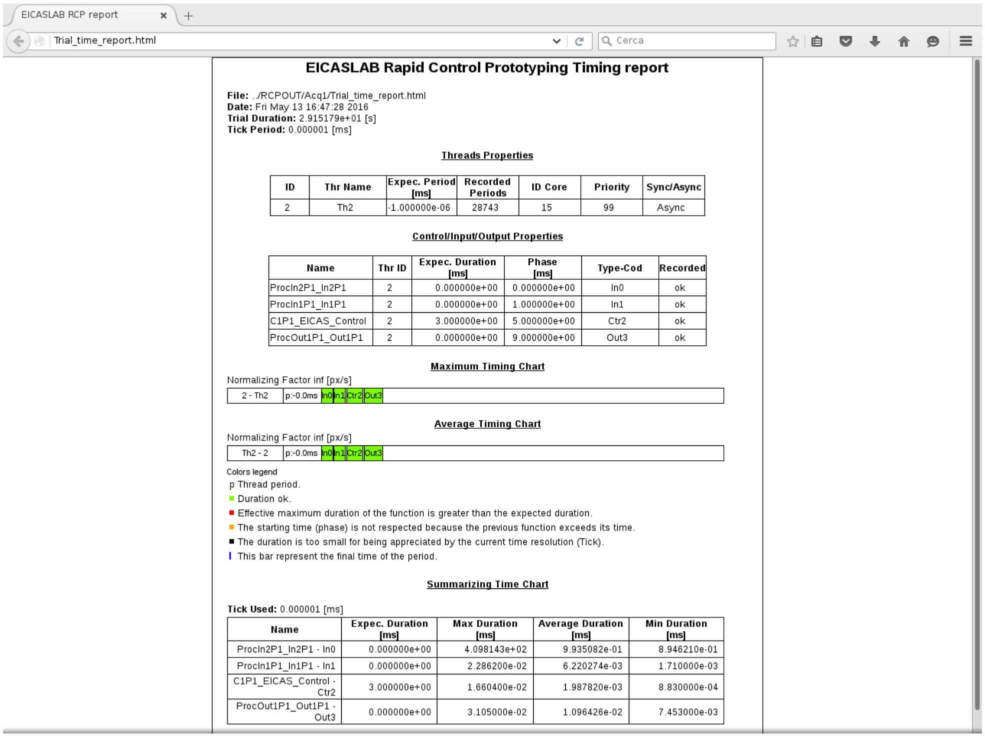
When the two EICASLAB instances with their two RCP Manager tools instances are opened, respectively for the real-time control and the real-time plant emulator, first start the control and then start the plant n order to have the best inter-process synchronization.

The graphical plot areas of the RCP Manager GUI will show a set of significant signals representing the trial going on (desired position, measures, simplified model state and command).

Note

If no graph or constant zero plots are displayed, please take a look at the F.A.Q. section at the end of the document for quick troubleshooting.

Among the features of the RCP Manager GUI which are not present in the SIM GUI, please note the HTML Timing Report (light blue button in the bottom right corner of the GUI) which generates automatically a document with data for real-time analysis of the trial.



EICASLAB Rapid Control Prototyping Timing report

File: ../RCPOUT/Acq1/Trial_time_report.html
Date: Fri May 13 16:47:28 2016
Trial Duration: 2.915179e+01 [s]
Tick Period: 0.000001 [ms]

Threads Properties

ID	Thr Name	Expec. Period [ms]	Recorded Periods	ID Core	Priority	Sync/Async
2	Th2	-1.000000e-06	28743	15	99	Async

Control/Input/Output Properties

Name	Thr ID	Expec. Duration [ms]	Phase [ms]	Type-Cod	Recorded
ProcIn2P1_In2P1	2	0.000000e+00	0.000000e+00	In0	ok
ProcIn1P1_In1P1	2	0.000000e+00	1.000000e+00	In1	ok
C1P1_EICAS_Control	2	3.000000e+00	5.000000e+00	Ctr2	ok
ProcOut1P1_Out1P1	2	0.000000e+00	9.000000e+00	Out3	ok

Maximum Timing Chart

Normalizing Factor inf [px/s]

2 - Th2	p:-0.0ms	In0	In1	Ctr2	Out3

Average Timing Chart

Normalizing Factor inf [px/s]

Th2 - 2	p:-0.0ms	In0	In1	Ctr2	Out3

Colors legend
p Thread period.
🟩 Duration ok.
🟥 Effective maximum duration of the function is greater than the expected duration.
🟧 The starting time (phase) is not respected because the previous function exceeds its time.
■ The duration is too small for being appreciated by the current time resolution (Tick).
| This bar represent the final time of the period.

Summarizing Time Chart

Tick Used: 0.000001 [ms]

Name	Expec. Duration [ms]	Max Duration [ms]	Average Duration [ms]	Min Duration [ms]
ProcIn2P1_In2P1 - In0	0.000000e+00	4.098143e+02	9.935082e-01	8.946210e-01
ProcIn1P1_In1P1 - In1	0.000000e+00	2.286200e-02	6.220274e-03	1.710000e-03
C1P1_EICAS_Control - Ctr2	3.000000e+00	1.660400e-02	1.987820e-03	8.830000e-04
ProcOut1P1_Out1P1 - Out3	0.000000e+00	3.105000e-02	1.096426e-02	7.453000e-03

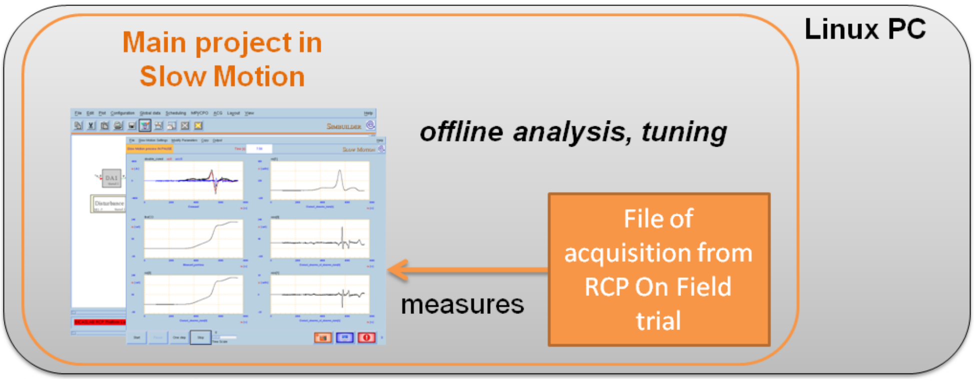
Note

The timing report and the error output could be used to identify real-time execution problems with respect to the 1 millisecond period of control algorithms.

Please see the F.A.Q. section of this user manual to solve common issues.

Slow Motion sub-mode

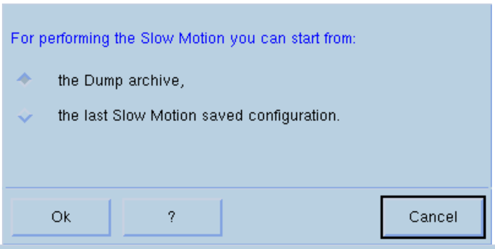
The Slow Motion sub-mode allows to replay offline the execution of the algorithms under development, giving them as input the recorded data from previous RCP On Field trials. The Slow Motion is extremely useful to analyse, tune and debug the control algorithms reducing the experimental testing time on field, and consequently the development time.



Linux PC
Main project in Slow Motion
offline analysis, tuning
File of acquisition from RCP On Field trial
measures

After your RCP On Field trials, close the RCP Manager tool and switch to Slow Motion operative sub-mode from the Operative Mode Selection window.

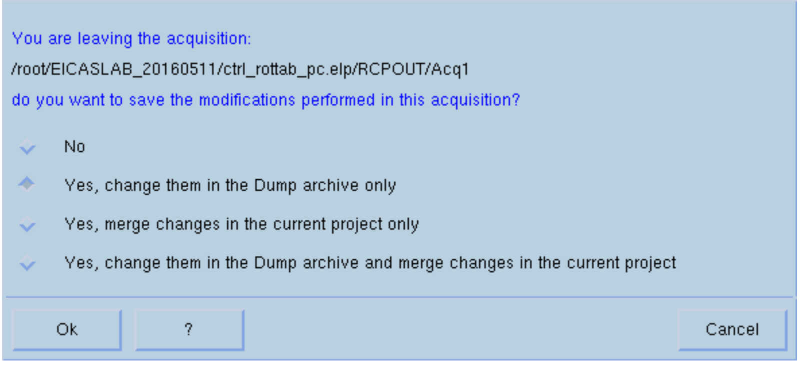
Entering and exiting the Slow Motion operative mode, EICASLAB will ask to select an option about the configuration to be used and if to save modifications to the parameters. In order to use the demo, it is enough to go ahead with the default options and press the “Ok” button. To understand more about that point, please read the “Slow Motion” chapter of the full EICASLAB user manual.



For performing the Slow Motion you can start from:

● the Dump archive,

○ the last Slow Motion saved configuration.



You are leaving the acquisition:
/root/EICASLAB_20160511/ctrl_rottab_pc.elp/RCPOUT/Acq1
do you want to save the modifications performed in this acquisition?

○ No

● Yes, change them in the Dump archive only

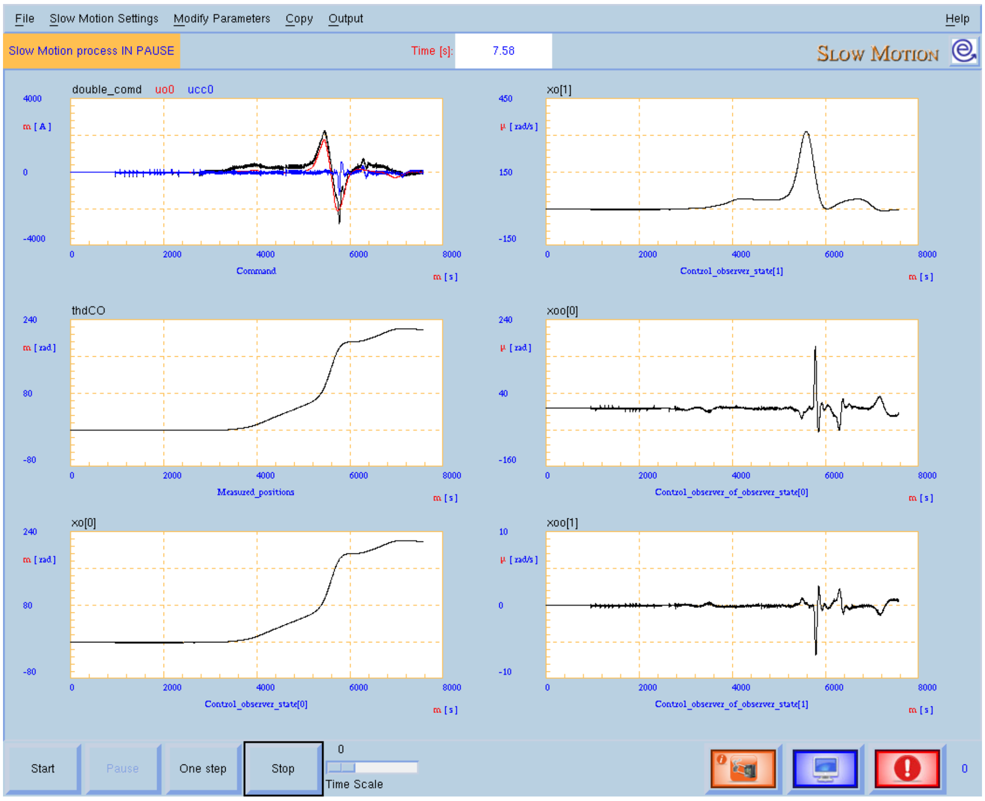
○ Yes, merge changes in the current project only

○ Yes, change them in the Dump archive and merge changes in the current project

In the MASTER tool, the Slow Motion > Acquisition selection menu item lets you choose the on field recording to be replayed by the Slow Motion tool. By default the Slow Motion tool repeats the last recorded RCP trial executed in RCP on field operative sub-mode.

Slow Motion tool

The Slow Motion tool is the EICASLAB tool which allows to replay the trials recorded previously during RCP on field activities. To proceed, compile and run the Slow Motion tool.



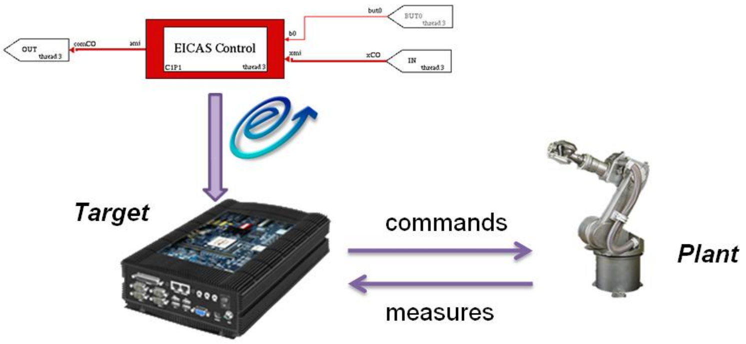
Slow Motion process IN PAUSE

Time [s]: 7.58

The Slow Motion will replay the selected recording and display the time diagrams of some significant signals specifically selected for the demo.

Target operative mode

The Target operative mode is the final step of the control design workflow in EICASLAB: its scope is to export and validate the control algorithms to the external hardware from the EICASLAB development platform.



This demo shows all the main features connected to the Target operative mode: Automatic Code Generation for target, Hardware-in-the-loop and Final Validation Test. The following diagrams show respectively the roles of the element of the scenario in the two operative sub-modes respectively.

HW	EICASLAB RCP Platform	Embedded board target	Rotating Table emulator
SW	simulates plant	runs control logic	unused

HW	EICASLAB RCP Platform	Embedded board target	Rotating Table emulator
SW	monitoring supervisor	runs control logic	simulated by second project

To start to operate in Target operative mode, open the Operative mode selection window from the MASTER tool and switch to Target operative mode by selecting the Hardware-in-the-loop operative sub-mode.

Automatic Code Generation for target

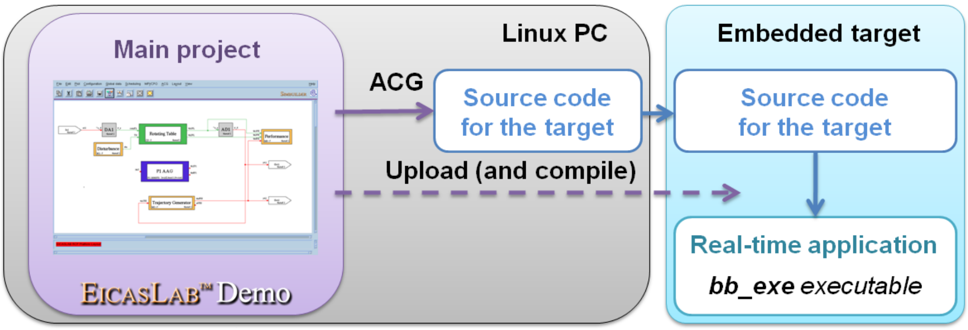
The Automatic Code Generation for target is a key feature of EICASLAB which automatically:

- generates the source code in ANSI-C language for the Application Software (AS) to be executed by the external target, i.e. the control algorithms tested in Modelling and Like Real-time Simulation and Rapid Control Prototyping operative modes;
- generates the source code in ANSI-C language for the Basic Software (BS) needed to run the AS on the external target hardware architecture, i.e. the code for the I/O interfaces, the scheduling routines,

the system configuration;

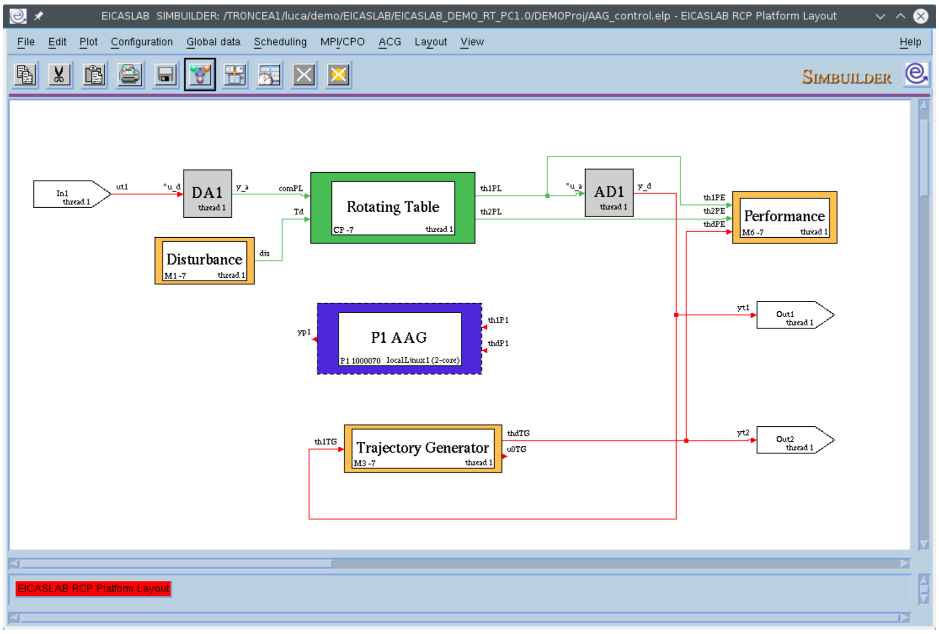
- transfers the source code to the external hardware if it will be compiled on the target or cross-compiles it in the EICASLAB environment and then transfers directly the executable binary to it.

Since the external hardware target of this demo is the BeagleBone Black board (or another Linux embedded board like the Raspberry Pi), we use the EICASLAB Automatic Code Generation support to export the control algorithms to the target.



The effort of all the operations listed above is moved from the control designer to the ACG feature of the EICASLAB software suite, so that the overall porting time is shortened and the user could directly develop and test the controls on it, even if he has only a limited knowledge of internal hardware details of target.

After changing to Hardware-in-the-loop operative mode, in the system layout represented in the SIMBUILDER you can see the processor block P1 is "assigned" to the "BeagleBoneBlack1" hardware target, as written in its label: that means that the routines and threads it includes are no more executed by the EICASLAB RCP Platform, as done previously during the RCP operations, because in Target operative mode the code has to be generated, exported and executed on the external target device.



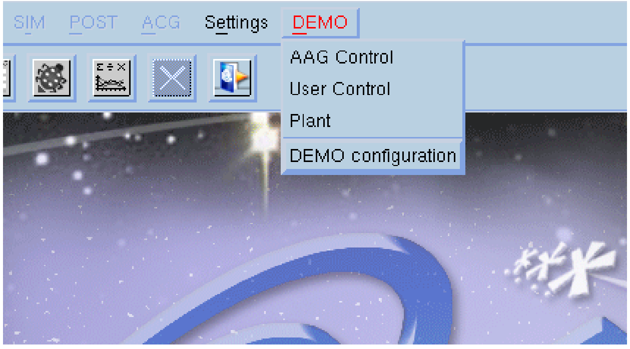
The label of the graphical block contains other information such as the Processor block identifier and the number of cores available in the processing unit of the target.

Note

In the Generic Linux version of the demo, the Target is of "remote generic Linux" type and the number of cores is 2. Do not mind if the model of embedded board has a different number of cores, because this demo uses only one of them.

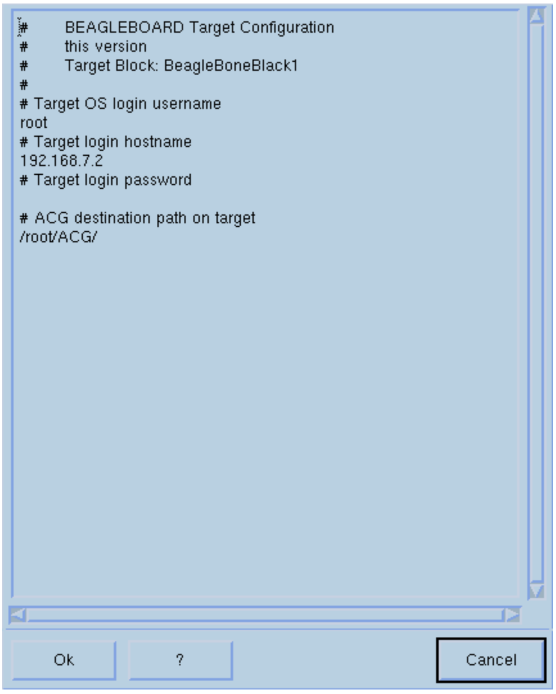
Interface settings

The networking activities between PC and target require that the communication interface parameters suit the actual network configuration, for that reason you may need to edit the IP addresses with respect to the default values of the demo projects as described in the "Network configuration" paragraph of the "Installation" section of this document. To edit that settings, you have to close temporarily the project (MASTER menu File > Quit project).



Target settings

Before generating and transferring the code to the external hardware, you have to check the target settings for exportation: right click on the Processor block in the system layout and move to the Target > Configuration > Settings menu to check the export parameters.

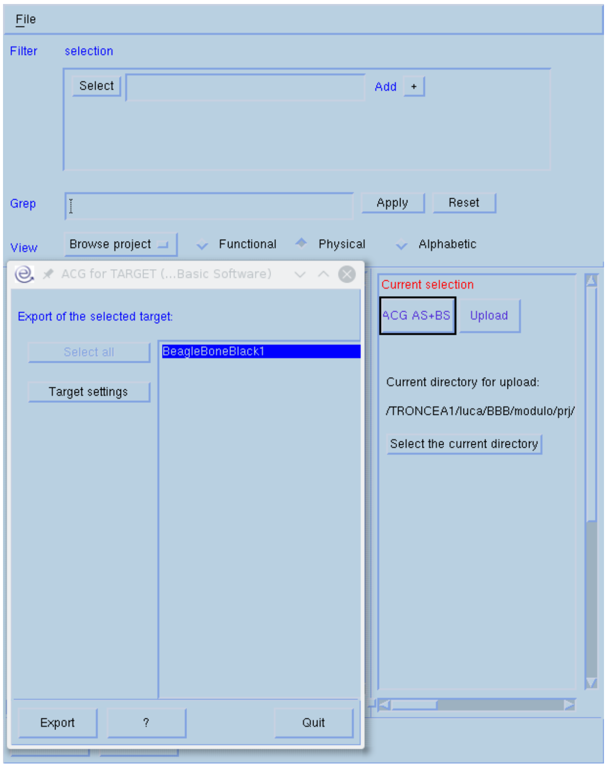


```
#       BEAGLEBOARD Target Configuration
#       this version
#       Target Block: BeagleBoneBlack1
#
# Target OS login username
root
# Target login hostname
192.168.7.2
# Target login password

# ACG destination path on target
/root/ACG/
```

Code generation and exportation

Once the target settings are correct, generate the real-time control program: open the Global Data window of SIMBUILDER tool (main menu > Global data > Open global data), move to the Physical View folder and select the target associated to the board.



Export of the selected target:

BeagleBoneBlack1

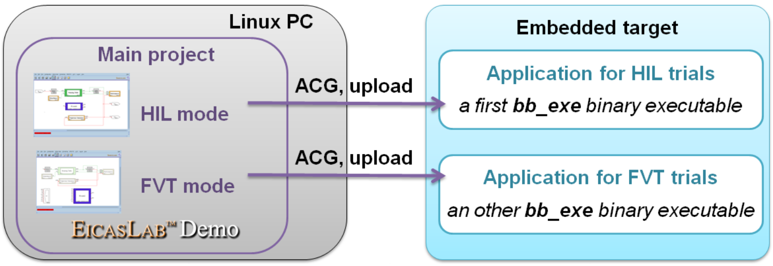
Current directory for upload:
/TRONCEA1/luca/BBB/modulo/prj/

Press the ACG AS+BS button and select the target, then click on the Export button and type the directory path. After a while, a dialog will notify the ACG is completed.

After that, press the Upload button and the source code will be compiled to a real-time binary application called *bb_exe* (or *lt_exe* for the generic Linux version of the demo) in the remote export directory.

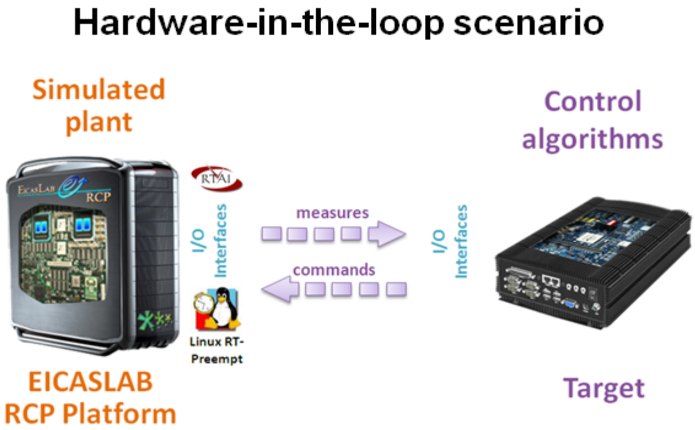
Note

The ACG procedure above must be done both in HIL and in FVT operative modes, because the two tests require two different real-time applications with different behaviours, as explained in the “Final Validation Test sub-mode” paragraph.

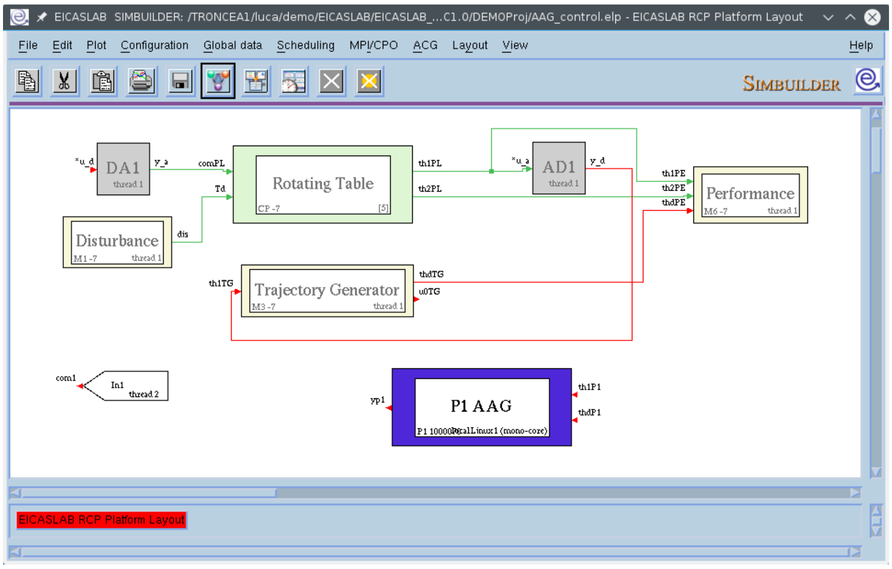

Linux PC
Main project
HIL mode
FVT mode
EICASLAB™ Demo
ACG, upload
ACG, upload
Embedded target
Application for HIL trials
a first bb_exe binary executable
Application for FVT trials
an other bb_exe binary executable

Hardware-in-the-loop sub-mode

The Hardware-in-the-loop (HIL) is a validation test consisting in the execution of the control algorithms by the target, which is connected to a simulated plant instead of the real one. The HIL test is often performed to validate the real-time behaviour of the hardware target without jeopardize the real plant.



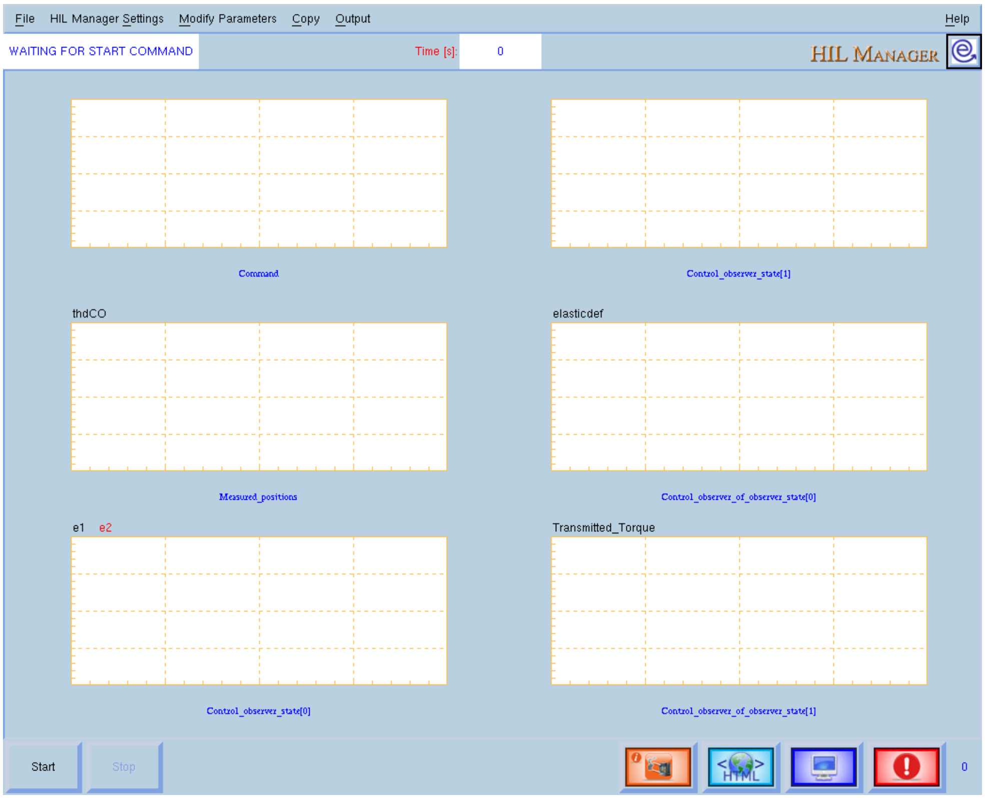
For that reason, the EICASLAB software suite includes the HIL operative sub-mode of the Target operative mode. In HIL, the Plant Area and the Mission Area are enabled and the plant simulation is then run by the HIL Manager tool (all blocks run on thread 1 in the demo, see figure below), whereas the control program is exported on the external hardware by ACG feature.



EICASLAB SIMBUILDER: /TRONCEA1/luca/demo/EICASLAB/EICASLAB_...C1.0/DEMOProj/AAG_control.elp - EICASLAB RCP Platform Layout

HIL Manager tool

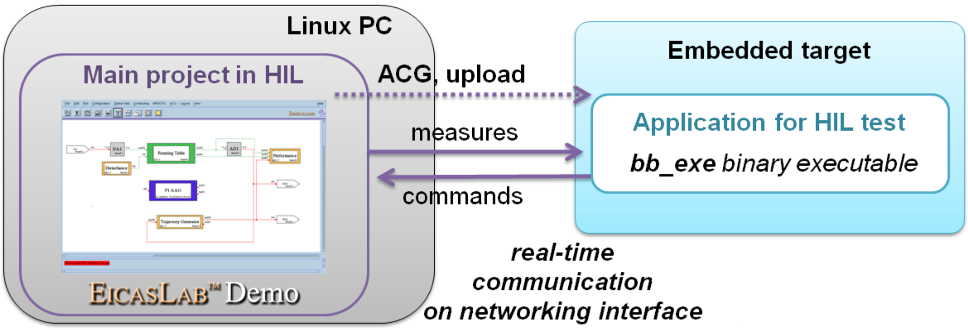
The HIL Manager GUI controls the real-time plant simulator for HIL and gives the user a set of time diagram areas to follow the status of the test. In order to proceed with the HIL trial, compile and run the HIL Manager tool[9] from button bar or top menu in the MASTER tool window.



[9] In order to access real-time capabilities the HIL Manager tool must be run as superuser. The EICASLAB installation procedure creates two links on the user desktop: the first one allows you to run EICASLAB as user while the other one allows you to run EICASLAB as superuser. In the latter case you will be asked to type the superuser password. Another way to run EICASLAB as superuser is to open a terminal, log in as superuser and run EICASLAB.

Execution of the control program on the target

Before starting the HIL trial, you must export the control program to the as already described in paragraph "Automatic Code Generation for target" and then launch it using a remote session from terminal emulator.



First of all, login remotely to the embedded target using the Secure Shell protocol (SSH), then change directory (the default directory is /root/ACG) and run the *bb_exe* (or *lt_exe*) real-time application on the remote target. The following lines are an example of login to the BeagleBone Black from the Linux PC, you must edit them to suit different LAN or user settings of your system configuration:

```
user@linux-pc$ ssh root@192.168.7.2

root@pandaboard# cd /root/ACG/BeagleBoneBlack1

root@pandaboard# ./bb_exe
```

While the remote control application is running, come back to EICASLAB and press the start button of the HIL Manager GUI to launch the trial. After that, the plot areas show in real time a set of significant signals.

Note

Please run first the remote application on the embedded board and, after that, start the HIL Manager. Such an order of launch provides correct synchronization of the two real-time programs.

When the first is already running and the second is not started yet, the standard output will print a periodical reception timeout because its counterpart is not transmitting data yet. Do not care about it unless the two programs are actually both running and the timeout is still notified.

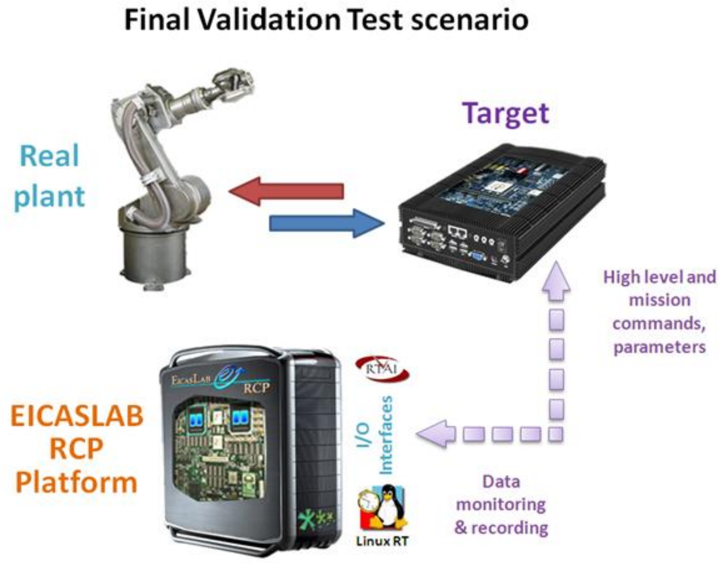
To terminate the HIL test, simply close the HIL Manager from its GUI on the PC and stop the remote target application pressing CTRL+C on the remote session on the embedded target.

As well as for the RCP Manager, the automatic generation of a timing performance report in HTML is available and a set of variables is recorded to be post-analysed by mean of the POST tool in the same way as the other operative modes.

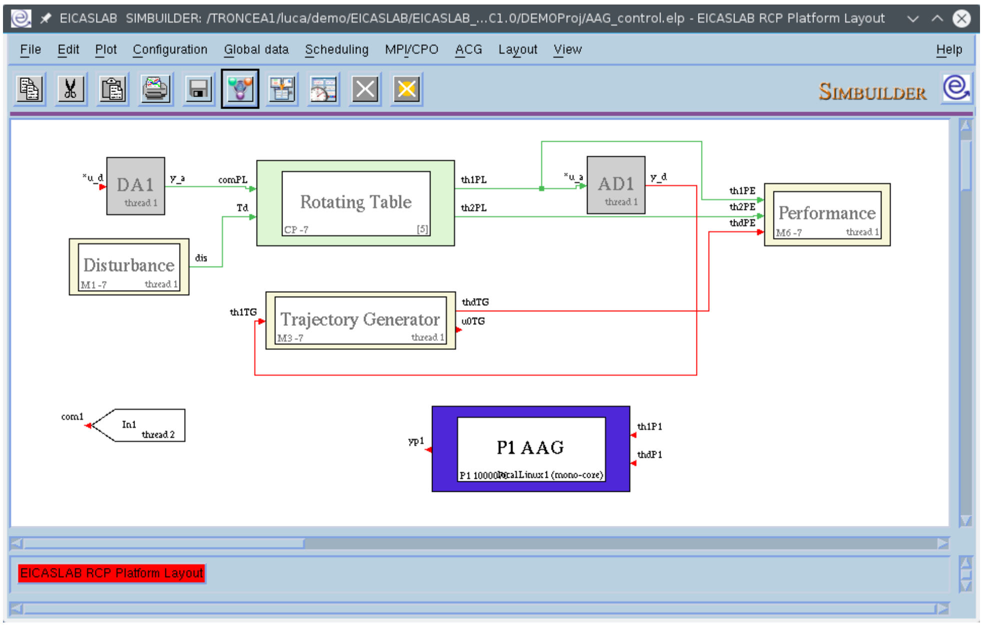
Final Validation Test sub-mode

The last step of the control validation is the execution on the final target connected to the real plant: in that configuration the PC has a role of monitor, data recorder and high-level supervisor. EICASLAB supports such a Final Validation Test (FVT) with its namesake operative sub-mode.

Please note that the typical FVT setup includes a minimum of three sub-systems, as graphically represented in the following picture.

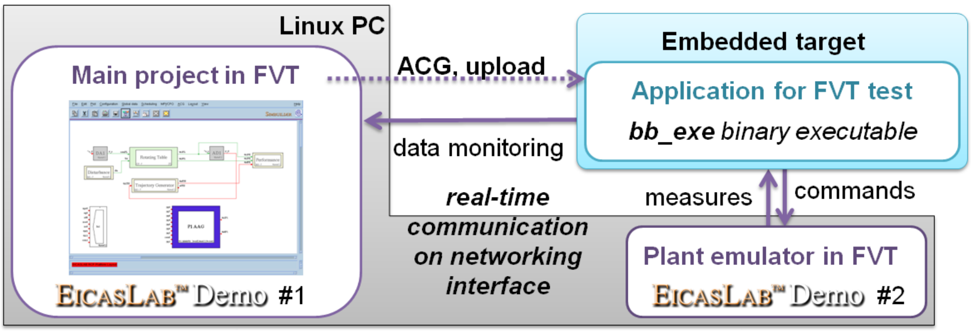

Final Validation Test scenario
Real plant
Target
High level and mission commands, parameters
EICASLAB RCP Platform
EICASLAB RCP
RTAI
I/O Interfaces
Linux RT
Data monitoring & recording

In the MASTER tool, switch to the FVT sub-mode of the Target operative mode from the Operative Mode Selection window. Again, it is possible to explore the SIMBUILDER layout and settings for FVT.



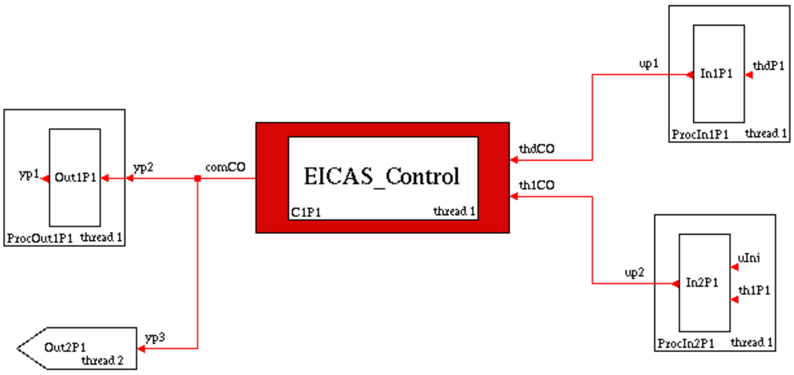
The FVT system layout has the plant disabled because the FVT setup expects that in this operative sub-mode the real plant is connected to the hardware target.

On the other hand, the demo is designed to avoid the need for the real plant, so the user must use again the second instance of EICASLAB to launch the software plant emulator, as well as done in the RCP On Field step of the demo.



The system layout includes an extra interface block "In1", which adds the monitoring function typical of the FVT phase: since during FVT the role of EICASLAB is external to the control loop, the "In1" interface listens to the remote control program for data packets containing the set of variables of interest to be recorded on disk and optionally plotted in the GUI of the TARGET Manager tool.

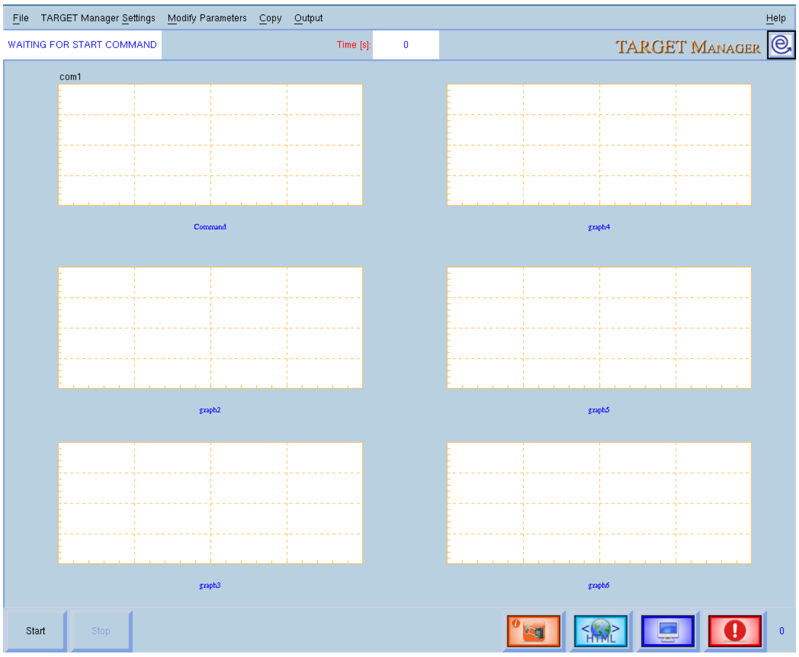
From the external target point of view, the processor layout (to be exported on the embedded target) has an extra interface matching the "In1" monitoring interface: the output interface "Out2P1" transmits the variables to be monitored, as shown in following figure. The "Out2P1" interface is executed on the thread n. 2, which has a longer period than the closed-loop thread because it is not so critical as the closed loop activities.



TARGET Manager tool

The TARGET Manager tool controls the FVT providing a user interface similar to the RCP Manager and HIL Manager ones. In order to proceed with the FVT trial, compile and run the TARGET Manager tool[10] from button bar or top menu in the MASTER tool window.

[10] In order to access real-time capabilities the TARGET Manager tool must be run as superuser. The EICASLAB installation procedure creates two links on the user desktop: the first one allows you to run EICASLAB as user while the other one allows you to run EICASLAB as superuser. In the latter case you will be asked to type the superuser password. Another way to run EICASLAB as superuser is to open a terminal, log in as superuser and run EICASLAB.



WAITING FOR START COMMAND

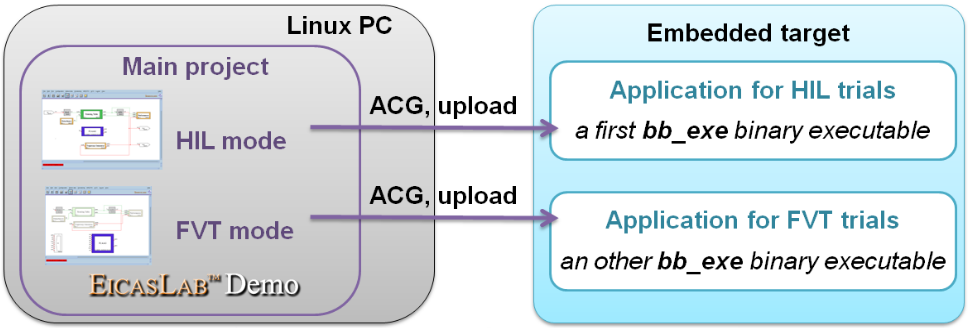
Before pressing Start and run the FVT trial, read the following paragraph about the execution of the three elements of the FVT scenario.

Plant real-time emulation and target ACG in FVT

Summarizing, the FVT scenario of the demo requires three real-time processing:

- the control program running on the external target, created and exported by ACG feature;
- the TARGET Manager in EICASLAB , which monitors the interaction of the control with the plant;
- the plant emulator project, running on the Linux PC as TARGET Manager of the second instance of EICASLAB in FVT operative sub-mode.

Please note that the FVT requires to generate a suitable control program to be exported to the embedded target, different from the one of the HIL, because during the FVT trial the embedded system has to send monitoring data to the EICASLAB TARGET Manager. To do the ACG for FVT, follow the same procedure done previously for the ACG for the HIL step using the Global Data window ("Code generation and exportation" and "Execution of the control program on the target" paragraphs).


Linux PC
Main project
HIL mode
FVT mode
EICASLAB™ Demo
ACG, upload
ACG, upload
Embedded target
Application for HIL trials
a first bb_exe binary executable
Application for FVT trials
an other bb_exe binary executable

Note

In order to synchronize correctly the interaction of the three real-time processes, pay attention to start them in the following order:

1) the external target executable *bb_exe* (or *lt_exe*)

2) the TARGET Manager of the ctrl_XXX main project (ctrl_XXX.elp)

3) the TARGET Manager of the plant emulator project (plant_XXX.elp)

When the *bb_exe* (or *lt_exe*) is already running and the plant emulator is not started yet, the standard output will signal a periodical reception timeout because its counterpart is not transmitting data yet. Do not care about it unless the two programs are currently running and the timeout is still notified.

N.B. : All the three real-time applications mentioned above must be run as superuser.

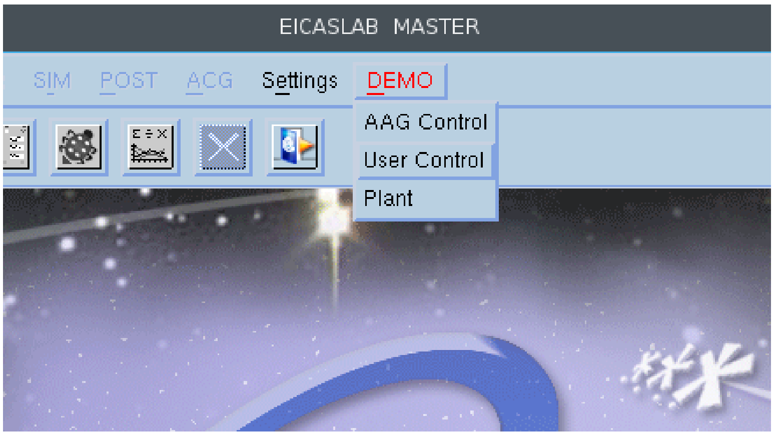
As in the other trials, a set of plots gives the idea of what is taking place within the rotating table model during FVT.

To terminate this last test, press CTRL+C for the external target executable and press Stop in the GUI of the two instances of the TARGET Manager.

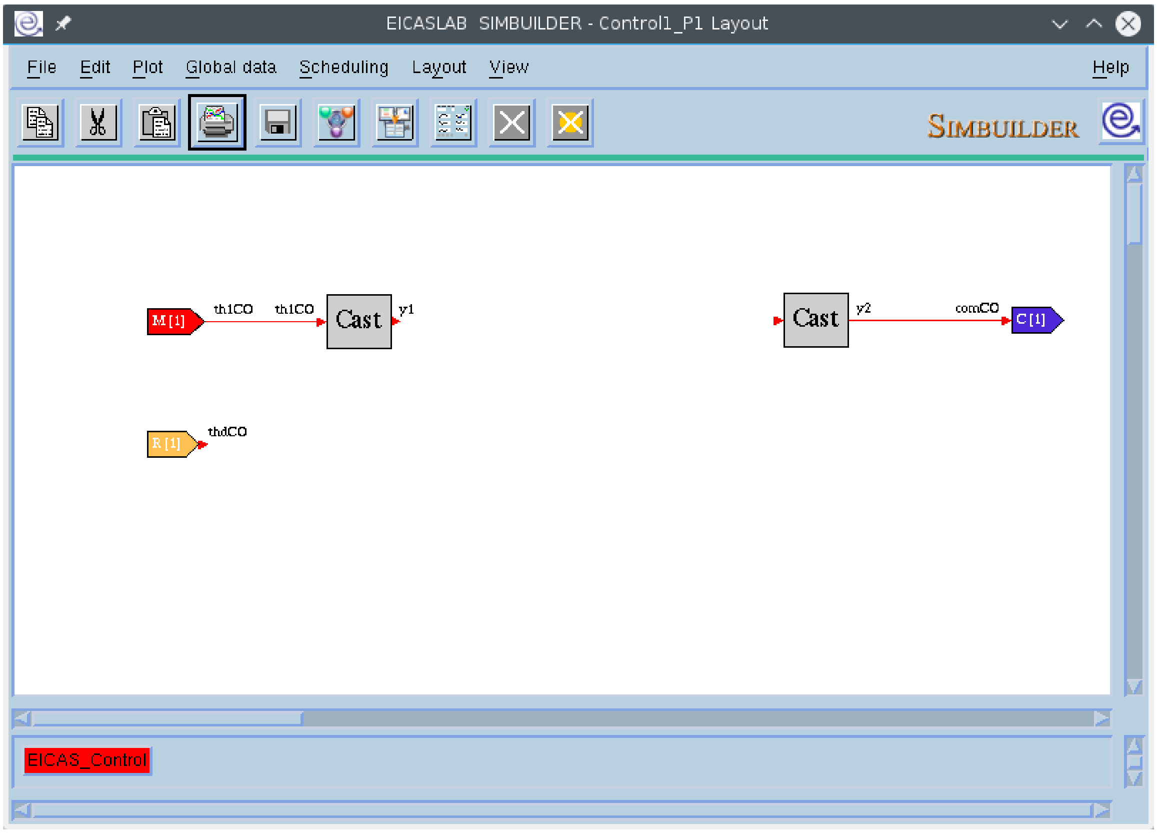
Congratulations! With this step the tour across the operative modes of the demo is completed.

The user defined control

The demo allows you to design and test your custom control algorithm instead of the one generated automatically by the EICASLAB AAG feature: close the AAG Control project using the MASTER menu "File > Quit project" and open the User Control one from the MASTER DEMO menu.



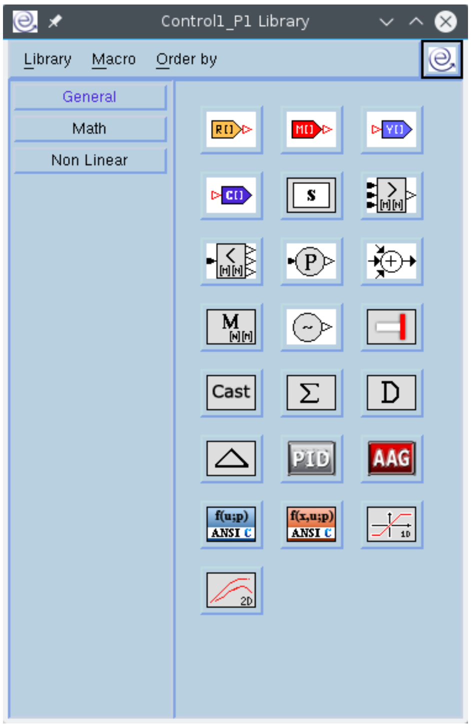
The User Control project has the same System and Processor layout as the AAG Control, but the Control block is an empty graphical control block. The th1CO, thdCO and comCO are respectively the digital measure (in rad), analog reference (in rad) and digital command (in ampere) for the plant.



In graphical programming mode, which is the default programming mode, you can use the blocks from the EICASLAB control library to program your algorithm. For example you can use a PID block or blocks from the Math and Non Linear sub-libraries.

Otherwise you can switch to the C language programming mode with right click on the Control block to access Control menu > Programming mode options.

Once you have connected inputs and output of the control with your algorithm, you can repeat all the steps of simulation and validation across the operative modes of EICASLAB that are described in the previous sections of this document.



Note

For more information on the control design and the programming modes in SIMBUILDER, please refer to the EICASLAB DEMO documentation and the full EICASLAB user manual.

To compare the overall performance of execution of the AAG and User projects, you can use the performance values available from menu DEMO > Performances of the SIM, RCP Manager, HIL Manager and TARGET Manager tools. The following table summarises the values for the Modelling and Like Real-time simulation of AAG Project on 60 s trial (PE stands for "Positioning Error" and m.s. means "mean squared").

Configuration	mean PE (PEm)	mean squared PE (PEt)
Point to point	-0.00234 mrad	0.0345 mrad

Configuration	mean E (Em)	m.s. E tot (Et)	m.s. E low freq (Elf)	m.s. E high freq (Ehf)
Tracking	0.0621 mrad	0.959 mrad	0.878 mrad	0.386 mrad

Frequently Asked Questions

All the following common questions refer specifically to the demo and not on general EICASLAB usage or other of its demos. If none of the next ones answers to your question, please contact us writing to support@eicaslab.com e-mail address.

I cannot find the target executable after the ACG for target. Where is it?

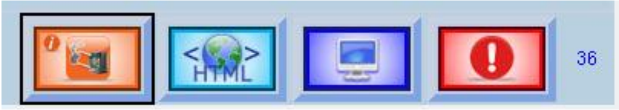
It is created after the upload operation. So, first of all be sure of having press the Upload button in the Physical View of the Global Data for the target.

Once the upload and remote compile is successfull, connect to the remote target using SSH in a terminal emulator (e.g. “ssh root@192.168.7.2” for the BeagleBone Black) and change to the export directory (“cd /root/ACG”) and enter the export directory to find the executable to be run (*bb_exe* or *lt_exe*).

Why I cannot run the RCP/HIL/TARGET Manager?

The real-time execution tools requires EICASLAB to be run with administrator permissions in order to access the highest priority scheduling policy, so please run EICASLAB and the target executable from ACG as administrator user (e.g. as root or with sudo in the command line).

Why I read an increasing number of errors and constant curves in the GUI?



The errors are taking place because the input interfaces do not receive the expected input packets, therefore the variables are not updated and remains to their initial value.

The most common cause is that the other real-time application of the current demo scenario is not running yet: for example the plant emulator in the RCP On Field phase of the demo.

Please check if the other real-time applications are running as described in the Quick Tour and the Operative Modes of this user manual.

Create your control algorithm and try it now!

Enjoy your work with EICASLAB DEMO!

excellence and passion
in automatic control design

EICAS Automazione S.p.A.