



ETAS RTPC V6.3.7

User's Guide

Copyright

The data in this document may not be altered or amended without special notification from ETAS GmbH. ETAS GmbH undertakes no further obligation in relation to this document. The software described in it can only be used if the customer is in possession of a general license agreement or single license. Using and copying is only allowed in concurrence with the specifications stipulated in the contract.

Under no circumstances may any part of this document be copied, reproduced, transmitted, stored in a retrieval system or translated into another language without the express written permission of ETAS GmbH.

© Copyright 2019 ETAS GmbH, Stuttgart

The names and designations used in this document are trademarks or brands belonging to the respective owners.

V6.3.7 R01 EN - 05.2019

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 5 |
| 1.1 | System Concept | 5 |
| 1.1.1 | "LABCAR Test Systems" Use Case | 5 |
| 1.1.2 | "EVE - Validation of AUTOSAR Software" Use Case | 5 |
| 1.2 | Target Group | 6 |
| 1.3 | Data Protection Declaration | 6 |
| 1.3.1 | Data Processing of ETAS RTPC | 6 |
| 1.3.2 | Data and Data Categories | 6 |
| 1.3.3 | Technical and Organizational Measures | 6 |
| 1.4 | About This Manual | 7 |
| 1.4.1 | Using This Manual | 7 |
| 2 | Installation | 9 |
| 2.1 | Delivery Scope | 9 |
| 2.2 | System Requirements | 9 |
| 2.2.1 | Hardware Requirements | 10 |
| 2.2.2 | Hardware Support | 10 |
| 2.2.3 | Special BIOS Settings | 11 |
| 2.3 | Installation | 12 |
| 2.3.1 | Installing ETAS RTPC on the Real-Time PC | 12 |
| 2.3.2 | Installing ETAS RTPC in a Virtual Machine | 14 |
| 2.3.3 | Hardware Integration | 14 |
| 2.3.4 | System Configuration | 15 |
| 2.3.5 | Licensing | 16 |
| 2.3.6 | Resetting the Boot Setting | 16 |
| 2.3.7 | Cloning Installed Versions | 17 |
| 2.3.8 | RTIO Configuration of the ES1130 | 19 |
| 2.3.9 | Clock Synchronization | 19 |
| 2.4 | PCI CAN and FlexRay Boards | 20 |
| 3 | Operation | 21 |
| 3.1 | Operating Modes | 21 |
| 3.1.1 | "Simulation Controller" Mode | 21 |
| 3.1.2 | "Idle/Configuration" Mode | 22 |
| 3.1.3 | "Ethernet Bridge" Mode | 22 |
| 3.1.4 | "Power Down" Mode | 22 |
| 3.2 | ETAS RTPC in the Experiment Environment LABCAR-EE | 23 |
| 3.3 | Runtime Traces | 24 |
| 3.4 | CAN Trace | 26 |
| 3.5 | Step-by-Step Execution of the Simulation | 30 |
| 3.6 | Real-Time Plugins (RT Plugins) | 32 |
| 3.6.1 | Preparation of the LABCAR-OPERATOR Project | 32 |
| 3.6.2 | The Plugin Files | 33 |
| 3.6.3 | Special Preprocessor Directives | 34 |
| 3.6.4 | Templates and Examples | 35 |
| 3.6.5 | Procedure | 35 |
| 3.7 | CPU Core Load | 36 |

| | | |
|----------|--|-----------|
| 3.8 | Benchmark Indicator | 37 |
| 3.9 | Testing Hardware Performance with "cyclictest" | 37 |
| 3.10 | CAN/CANFD-Abstraction and LIN-Abstraction | 39 |
| 3.10.1 | General Functional Principle | 39 |
| 3.10.2 | Abstraction of the Bus Interfaces During the Build Process | 40 |
| 3.10.3 | Converting the Channel Description into Concrete Bus Interfaces at Runtime | 41 |
| 3.10.4 | Template for a Channel Configuration | 41 |
| 4 | Configuration | 44 |
| 4.1 | Applications | 44 |
| 4.1.1 | Model-in-the-Loop (MiL) | 44 |
| 4.1.2 | Hardware-in-the-Loop (HiL) | 44 |
| 4.1.3 | Hardware-in-the-Loop with Measuring and Calibration Access (M&C) .. | 45 |
| 4.1.4 | Hardware-in-the-Loop with Load and Error Simulation | 45 |
| 4.1.5 | Massive I/O | 45 |
| 4.2 | The Web Interface | 46 |
| 4.2.1 | Access to the Web Interface | 46 |
| 4.2.2 | Navigation | 47 |
| 4.2.3 | Functions | 48 |
| 5 | Troubleshooting | 59 |
| 6 | Glossary | 63 |
| 7 | ETAS Contact Addresses | 65 |
| | Index | 66 |

1 Introduction

ETAS RTPC transforms your standard PC into a high-performance simulation target. As ETAS RTPC consistently uses standard commercial components, you benefit from the continuous development of PC technology with regard to the available processing power and communication bandwidth. ETAS RTPC uses all processor cores of state-of-the-art Intel multi-core processors.

1.1 System Concept

1.1.1 "LABCAR Test Systems" Use Case

Once the ETAS RTPC software has been installed, your standard PC is transferred into a real-time PC for LABCAR test systems.

Fig. 1-1 shows the hardware setup of a ETAS RTPC system. The real-time PC is connected to the user PC via Ethernet and can be selected as a simulation target in the LABCAR software.

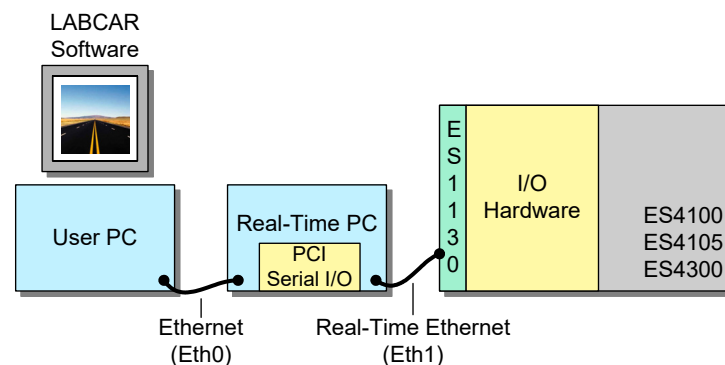


Fig. 1-1 RTPC System

The real-time PC is controlled entirely from the user PC via an intuitive Web interface. Keyboard and monitor are not needed (apart from during installation).

The I/O hardware communicates with the real-time PC via Ethernet – the I/O data gets to its target deterministically and securely via a highly optimized point-to-point connection.

A PC with ETAS RTPC is also used as a simulation PC in the open and scalable ETAS HiL system PT-LABCAR which is designed for closed-loop tests of ECUs in the drive train area.

1.1.2 "EVE - Validation of AUTOSAR Software" Use Case

ISOLAR-EVE allows to validate AUTOSAR compliant software on different platforms. ETAS RTPC is one of those platforms.

ETAS RTPC allows to create and operate a so called VECU (virtual electronic control unit) with access to external hardware resources like CAN, DIO, ADC, PWM.

ISOLAR-EVE runs on a user PC and operates the real-time PC mainly via automation interfaces in order to build the executable and to control the execution.

General configuration like licensing and monitoring may be done via ETAS RTPC's intuitive Web interface.

1.2 Target Group

This manual addresses qualified personnel working in the fields of automobile control unit development and calibration. Specialized knowledge in the areas of measurement and control unit technology is required.

Users should be familiar with the Microsoft Windows XP, Windows Vista, or Windows 7 operating system. Furthermore, the users should be familiar with the Windows file storage system, especially the connections between files and directories.

The users have to know how to use the basic functions of the Windows File Manager and Program Manager or the Windows Explorer, respectively, and they should be familiar with the "drag-and-drop" functionality.

ISOLAR-EVE users not yet familiar with automotive software descriptions based on the AUTOSAR framework are advised to get a suitable training upfront.

Knowledge about the general features and operation of Eclipse based software products will be beneficial but is not mandatory.

1.3 Data Protection Declaration

1.3.1 Data Processing of ETAS RTPC

Please note that personal or person-related data or data categories are processed when using ETAS RTPC.



The purchaser of ETAS RTPC is responsible for the legal conformity of processing the data in accordance with Article 4 No. 7 of the General Data Protection Regulation (GDPR).

As the manufacturer, ETAS GmbH is not liable for any mishandling of this data.

1.3.2 Data and Data Categories

When using the ETAS License Manager in combination with user-based licenses, particularly the following personal or person-related data or data categories are recorded for the purposes of license management:

Communication data: IP address

User data: User ID

1.3.3 Technical and Organizational Measures

ETAS RTPC does not encrypt the collected personal or person-related data.

Please ensure the data security of the recorded data by suitable technical or organizational measures of your IT system, e.g. by classical anti-theft and access protection on the hardware.

1.4 About This Manual

This manual is intended, on the one hand, for all users who install, maintain and uninstall ETAS RTPC on the target PC, and, on the other, for system administrators who make ETAS RTPC available on a file server for installation via the network. It provides important information on the delivery scope, hardware and software requirements for single station and network installation as well as on how to prepare the installation.

This manual consists of the following chapters:

- "Introduction" on page 5
This chapter
- "Installation" on page 9
This chapter contains information on system requirements and on how to install ETAS RTPC including details of the environment configuration.
- "Operation" on page 21
This chapter describes the operating modes of ETAS RTPC, as well as the creation and migration of projects for ETAS RTPC.
- "Configuration" on page 44
This chapter describes the uses and configuration of ETAS RTPC with the web interface.
- "Troubleshooting" on page 59
This chapter presents the description of some typical problems with possible solutions.
- "Glossary" on page 63

1.4.1 Using This Manual

1.4.1.1 Representation of Information

All activities to be carried out by the user are shown in what we call a "Use-Case" format, i.e. the target to be achieved is defined briefly in the title and the individual steps necessary to achieve this target are then listed. The information is displayed as follows:

Target definition

Any introductory information...

1. Step 1
Possibly an explanation of step 1...
2. Step 2
Possibly an explanation of step 2...

Any concluding remarks...

Concrete example:

To create a new file

If you want to create a new file, no other file may be open.

1. Select **File** → **New**.
The "Create file" dialog box appears.

- 2. Enter a name for the file in the "File name" field.
The file name must not exceed 8 characters.
- 3. Click **OK**.


The new file is created and saved under the name specified. You can now work with the file.

1.4.1.2 **Typographic Conventions**

The following typographic conventions are used:

| | |
|--|---|
| Select File → Open . | Menu commands are shown in boldface. |
| Click OK . | Buttons are shown in boldface |
| Press <ENTER>. | Keyboard commands are shown in angled brackets in block capitals. |
| The "Open File" dialog box appears. | Names of program windows, dialog boxes, fields etc. are shown in quotation marks. |
| Select the file <code>setup.exe</code> . | Text in drop-down lists, program code, as well as path and file names are shown in the <code>Courier</code> font. |
| A conversion between the file types logical and arithmetic is <i>not</i> possible. | Content markings and newly introduced terms are shown in <i>italics</i> |

Important notes for the user are shown as follows:

 **TIP**

Important note for the user.

2 Installation

This chapter includes information on the following topics:

- "Delivery Scope" on page 9
- "System Requirements" on page 9
 - "Hardware Requirements" on page 10
 - "Special BIOS Settings" on page 11
- "Installation" on page 12
 - "Installing ETAS RTPC on the Real-Time PC" on page 12
 - "Installing ETAS RTPC in a Virtual Machine" on page 14
 - "Hardware Integration" on page 14
 - "System Configuration" on page 15
 - "Licensing" on page 16
 - "Resetting the Boot Setting" on page 16
 - "Cloning Installed Versions" on page 17
 - "RTIO Configuration of the ES1130" on page 19
 - "Clock Synchronization" on page 19
- "PCI CAN and FlexRay Boards" on page 20

2.1 Delivery Scope

The delivery scope of ETAS RTPC V6.3.7 includes

- ETAS RTPC Software DVD
- This User's Guide
- Possibly a PC from ETAS

2.2 System Requirements

You need the following to install and run the ETAS RTPC software:

- ETAS RTPC Software CD
- A ETAS RTPC software license
- Suitable PC hardware (see "Hardware Requirements" on page 10)
- Ethernet crossover cable (2x RJ45 socket) for the connection between the real-time PC and user PC

2.2.1 Hardware Requirements

ETAS RTPC is compatible with most standard PCs. To run the software, the following requirements must be fulfilled:

| | |
|--|--|
| Processor | Intel Xeon family Intel Core i family Intel Core 2 family (Duo and Quad) |
| Ethernet adapter for the connection to the user PC | All Ethernet adapters supported by Linux kernel 3.18 |
| Ethernet adapter for the connection to the ES1130 | All Ethernet adapters supported by Linux kernel 3.18 e.g. - Intel PRO/100 card S, revision 12 (0C) or later - Intel PRO/1000 family - Intel PRO/1000 PT Quad Port Server Adapter - Intel Gigabit CT Desktop Adapter |
| RAM | Min. 4 GB, 8 GB and more recommended |
| Hard disk | IDE or SATA, min. 250 GB |
| Graphic | VGA-compatible |
| Monitor | Is only required for basic installation |
| Keyboard | PS/2 or USB, is only required for basic installation |
| CD-ROM drive | IDE, SATA or USB, is only required for basic installation (bootable) |
| Operating system | Not necessary |

TIP

During ETAS RTPC operation, success or error messages are emitted in the form of sound. If your PC does not have an output device for acoustic signals, you can access information on the current status via the web interface (see section "The Web Interface" on page 46).

For more precise details on the compatibility of PCs made by different manufacturers, refer to the compatibility list (<http://192.168.40.14/documentation/>).

2.2.2 Hardware Support

ETAS RTPC supports a maximum number of hardware connections related to the ES5300 series.

TIP

Before you design or configure a system with the specified maximum number of hardware connections, get in touch with your ETAS contact person. The ETAS contact person supports you to select the individual hardware.

The following table contains the maximum number of the hardware connections:

| Hardware | Maximum Number |
|-------------------------|----------------|
| CAN Boards | 32 |
| Network Ports | 32 |
| FlexRay Boards | 8 |
| RS232 Ports/RS485 Ports | 16 |

If you extend the RS232/RS485 port number to 16, a change of the port order is possible. If you use these ports, verify the hardware configuration and change it accordingly.

2.2.3 Special BIOS Settings

2.2.3.1 Boot Settings

To ensure the installation from CD-ROM works, the system's BIOS has to be configured so that the PC boots from the CD-ROM ("primary boot device = CD-ROM"). If you want to use the PC later without a keyboard, you also have to specify this setting ("no halt on keyboard"). The names of the settings in brackets may vary - for more details, refer to your PC's User's Guide.



TIP

For details of the ETAS PC hardware BIOS settings, refer to <http://192.168.40.14/documentation/>.

2.2.3.2 Hyper-Threading

The usage of hyper-threading has an impact on the real-time performance of the used real-time PC as two hyper-threaded cores share some CPU resources, e.g. the FPU and the ALU are typically shared among two hyper-threaded cores.

This leads to influences in the execution of the code running on the hyper-threaded cores as one hyper-threaded core may have to wait for the other hyper-threaded core to release a shared CPU resource.

If you are running in issues getting real-time violations it might be helpful to deactivate hyper-threading in the BIOS settings of the used real-time PC. Especially if you are running simulation models with very fast cycle times (faster than 100 μ s), it is recommended to disable hyper-threading completely as the additional time for waiting for CPU resources may break the cycle times.

However there are some situations where hyper-threading can be used without getting real-time violations. In these cases you may benefit from the additional CPU power.

Typical situations for this are setups that are using simulation models with moderate cycle times (slower than 500 μ s) or also for real-time PCs that are used as build machines only. In these situations the higher overall computation power can be used beneficially.

2.3 Installation

The installation of ETAS RTPC is divided into four parts which are each described in the sections below:

- "Installing ETAS RTPC on the Real-Time PC" on page 12
- "Installing ETAS RTPC in a Virtual Machine" on page 14
- "Hardware Integration" on page 14
- "System Configuration" on page 15
- "Licensing" on page 16
- "Resetting the Boot Setting" on page 16
- "Cloning Installed Versions" on page 17
- "RTIO Configuration of the ES1130" on page 19
- "Clock Synchronization" on page 19



TIP

ETAS also offers ETAS RTPC as a preinstalled version on PC hardware recommended and tested by ETAS. In this case, please start by reading the section "Hardware Integration" on page 14.

2.3.1 Installing ETAS RTPC on the Real-Time PC

If ETAS RTPC has already been installed by ETAS, please continue with the next section "Hardware Integration" on page 14.

To install ETAS RTPC on the real-time PC

1. Connect the monitor and keyboard to the PC hardware.
2. Switch the PC and the monitor on.
3. Insert the ETAS RTPC Software DVD into the real-time PC's DVD drive.
4. In the start screen select the desired option (default:
ETAS RTPC Vx.y.z - Standard Install).



TIP

With this option the file system "ext3" is installed – if you have to be compatible with ETAS RTPC versions before V5.1.0. select the "Install for parallel use with RTPC versions before V5.1.0" option.

5. Wait until the ETAS RTPC installation dialog box is displayed.
6. Select the option [Install RTPC software] using the cursor keys.
7. Press <ENTER>.



TIP

All data on the hard disk is deleted by the ETAS RTPC installation!

8. Confirm the warning by selecting the option [Yes, continue with installation].
9. Wait until the ETAS RTPC installation has been completed (takes about 5 minutes).
10. Remove the ETAS RTPC Software DVD.
11. Select the option [Reboot].

The PC reboots.



12. Wait until the PC plays an ascending sequence of notes.

ETAS RTPC has been installed successfully.



13. Shut down the PC by quickly pressing the on/off switch.

After just a few seconds, the PC plays a descending sequence of notes and shuts down.

This indicates that ETAS RTPC has been shut down correctly and can be disconnected from the mains.

For more details on these acoustic signals, refer to the chapter "Operation" on page 21.

i TIP

ETAS RTPC uses a robust file system, but hardware and software can still be damaged when powered off without shutting down or during a sudden voltage drop. ETAS thus recommends shutting down ETAS RTPC by briefly pressing the on/off switch to avoid these problems.

14. Remove monitor and keyboard (see "Special BIOS Settings" on page 11).

To test the reachability of ETAS RTPC

1. Toggle to the host.
2. Select **Run** from the Start menu.
3. Enter "cmd".

The command prompt opens.

4. Enter the following:

```
ping 192.168.40.14

U:\>ping 192.168.40.14

Pinging 192.168.40.14 with 32 bytes of data:

Reply from 192.168.40.14: bytes=32 time=3ms TTL=128
Reply from 192.168.40.14: bytes=32 time<10ms TTL=12
Reply from 192.168.40.14: bytes=32 time<10ms TTL=12
Reply from 192.168.40.14: bytes=32 time<10ms TTL=12

Ping statistics for 192.168.40.14:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 0ms
```

5. Alternatively, you can open the web interface (<http://192.168.40.14>) in your browser.

2.3.2 Installing ETAS RTPC in a Virtual Machine

The ETAS RTPC software can be installed in a virtual machine. Use the installation image for this. The virtual machine must be configured according to the system requirements of the ETAS RTPC (see "System Requirements" on page 9). For the configuration and installation of the virtual machine, please refer to the instructions of the corresponding tool.

To operate the ETAS RTPC in a virtual machine, ETAS recommends the use of a "host-only" interface as network interface eth0. The steps for the configuration of such a network interface can be found in the documentation of the corresponding tool.

2.3.3 Hardware Integration

This section describes the integration of the real-time PC into a Hardware-in-the-Loop test system (HiL). Other configurations, such as Model-in-the-Loop or HiL system with measuring and calibration access, are described in the section "Applications" on page 44.

ETAS RTPC uses the identifier Eth0 (Host) for the onboard Ethernet interface – the Ethernet board has the identifier Eth1 (HW).

To connect the user PC and the ES1130

1. Connect the real-time PC to the experimental target interface of the user PC using the onboard Ethernet interface (Eth0) with a crossover cable.



TIP

Crossover cables have crossover connections and are often indicated on one or both ends with yellow or orange connectors.

2. Connect the real-time PC to the ES1130 Simulation Controller Board using the Ethernet interface (Eth1). The ETAS CBE100.1 cable (for ES1130.2 and ES1130.3) or K107 (for ES1130.1) can be used for this connection.

2.3.4 System Configuration

ETAS RTPC is easy to configure from the user PC via a Web browser (e.g. Internet Explorer or Mozilla Firefox) – the configuration dialog can be accessed via the IP address 192.168.40.14.

Carry out the steps described below to configure ETAS RTPC for Hardware-in-the-Loop tests (HiL). For more details on configuration modes refer to the section "Applications" on page 44.

To open the configuration dialog

1. Launch the real-time PC by pressing the on/off switch on the front.
2. Switch the chassis with the I/O hardware on.
3. Wait a few seconds until you hear the ascending sequence of notes.
4. Launch a Web browser on the user PC.
5. Invoke the URL <http://192.168.40.14>.



You can also assign this IP address an alias which is easier to remember – for more details refer to the section "Access to the Web Interface" on page 46.

The ETAS RTPC configuration dialog opens.

The simulation controller has to be stopped to configure the ETAS RTPC Software (see "Simulation Controller" Mode" on page 21).



As long as the simulation controller is running, the *Main Page* link is inactive!

To stop the simulation controller

1. Click the **Stop Simulation Controller** button.
2. Click the **Main Page** link.

To run the configuration

1. Click the **Configure ETAS RTPC** link.
2. Click the **Autoconfigure** button.

The autoconfiguration is run and the result is set in the "RTPC_USAGE_ETH1" box.

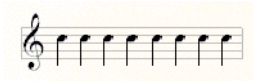
To restart the simulation controller

1. Click the **Main Page** link.
2. Click the **Simulation Controller** link.

3. Click the **Start Simulation Controller** button.



When you hear the ascending sequence of notes, the simulation controller is active again.



TIP

If instead you hear the sequence of error notes, the simulation controller could not be started. For more details refer to the chapter "Troubleshooting" on page 59.

The configuration for HiL mode has now been carried out successfully. For more details on other configuration modes refer to the section "Troubleshooting" on page 59.

2.3.5 Licensing

To operate ETAS RTPC V6.3.7, you require a license which can be obtained in the form of a license file from the [ETAS License Portal](#). For more details, please refer to the manuals "LABCAR-OPERATOR V5.4.9 - Getting Started" or "ETAS ISOLAR-EVE V3.2 - Installation Guide for Tooling PC".or "COSYMV2.1.0 - Installation and Administration Guide".

TIP

The license for ETAS RTPC is not installed using the ETAS License Manager – as described above – but in the web interface (see "License Management" on page 57).

License Versions

- RTPC_EVE
To execute ISOLAR-EVE projects
- LCS_RTPC_RT
To execute LABCAR-OPERATOR projects in real time
- COSYM_RT
For the Build process of a COSYM project and its real-time execution
- LCS_RTPC_OLC
When running LABCAR-OPERATOR projects and COSYM projects, the minimum period duration is 200 ms.

TIP

No license is required for the Build process of a LABCAR-OPERATOR project!

2.3.6 Resetting the Boot Setting

If there are problems when changing boot settings (e.g. after repartitioning), these settings can be reset.

To reset boot settings

1. Select **Main Page >> Configuration**.

2. Go to the "System Boot Settings" section.
3. Select a version
4. Click **Set Boot Version**.

System Boot Settings

Caution: Modification of the values below requires a fundamental knowledge of RTPC.

Select version to boot

This version will be used with the next system boot.
Info: The current system runs from partition sda7

sda7: ETAS-RTPC V6.3.7

Set Boot Version

[Expert Edit](#) [Reset Boot Settings](#)

Reboot System

Reboot RTPC using the selected boot version.

Reboot System


5. Reboot the system with **Reboot System**.

2.3.7 Cloning Installed Versions

You can clone versions of ETAS RTPC that have already been installed by generating an installable file (*.tgz) and downloading this file to your operating PC.

To clone a version

1. Select **Main Page >> Installation / Update >> Clone Installed Version**.
2. Select one of the installed versions.

 **TIP**

The currently active version cannot be cloned!

3. Enter a name for the clone to be created.

- Click **Generate Clone Image**.

Clone Installed Version

1: ETAS-RTPC V6.3.7

2: ETAS-RTPC V6.3.7 (Active)

Create a software image from a currently installed version

rtpcimage-imagename.tgz

Generate Clone Image

The procedure is started and the progress displayed in the web interface.



Cloning finished

- [Select Boot Version](#)
- [Return to Installation / Update](#)

Cloning-Progress

```
Cloning Partition '/dev/sda5' to 'rtpcimage-V6.3.7.tgz'
action=creating-tar
Creating tar file from partition '/dev/sda5' ...
tar: ./var/run/proftpd/proftpd.sock: socket ignored
Special processing of some files ...
/etc/fstab
action=compressing-tar
Compressing the tar file (this takes a while) ...
--- finished ---
```

Once the procedure has been completed, you can download the clone to use on your operating PC.

To download a clone

- Click the **Return to Installation / Update** link.
- Go to "Download Software Image".
- Select the version just created.
- Click **Download**.

Download Software Image

rtpcimage-V6.3.7.tgz

Download

A download dialog box opens in which you can save this file.

- If you want to delete this file from your real-time PC, select the clone under "Delete Installed Software Images".

6. Click **Delete**.

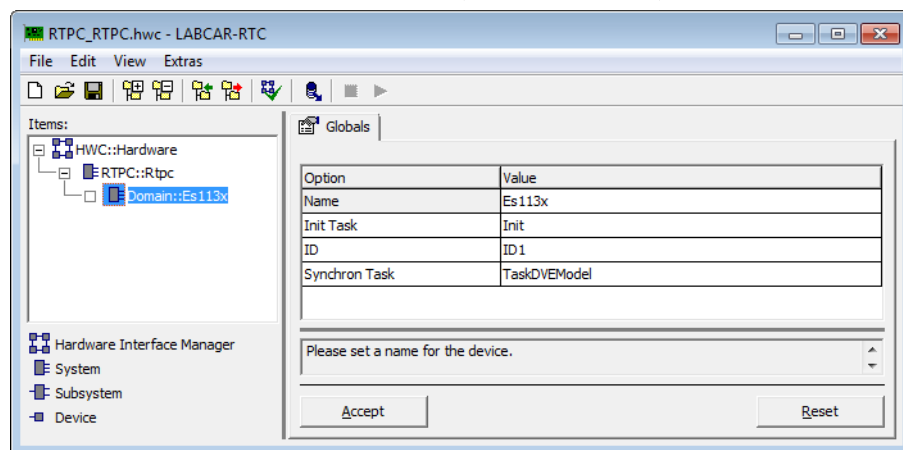


The file is deleted from your real-time PC.

2.3.8 RTIO Configuration of the ES1130

In the case of an ES1130 which is logically connected after a real-time PC, communication between ETAS RTPC and the ES1130 takes place synchronously to a selectable task. This selection is made in LABCAR-RTC (RTIO) in the "Globals" tab of the ES1130 ("Synchron Task").

The communication between ETAS RTPC and ES1130 is optimized to this task. The best results are obtained when the simulation model and the periodic hardware accesses run in exactly one task ("TaskDVEModel").



2.3.9 Clock Synchronization

To synchronize the time on the user PC and the real-time PC, you can activate the NTP¹ server integrated in Windows.

Activation takes place by making entries in the Windows registry – you will find the file for setting the relevant keys via the web interface at

<http://192.168.40.14/tools/NTP/NTP-LocalParameter.reg>

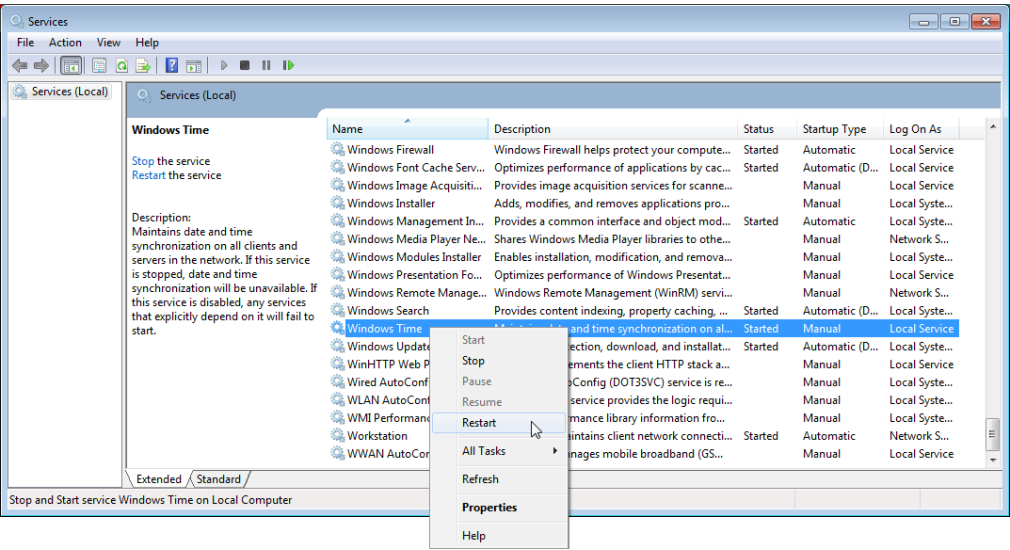
This configures the Windows service "ws32time" – the server must be rebooted after the keys have been modified.


To reboot the NTP server

1. Select **Control Panel** → **Administrative Tools** → **Services**.
2. Select the "Windows Time" service.
3. Open the shortcut menu by pressing the right-hand mouse button.

¹Network Time Protocol

4. Select **Restart**.



 **TIP**

Please note that when the service has been rebooted, it can take a few minutes until the time is precise enough to be made available to other clients.

In the web interface it is possible to carry out an ad hoc synchronization of real-time PCclock and host clock (see **“Clock Synchronization”** on page 54).

2.4 PCI CAN and FlexRay Boards

For communication between the model and the ECU, you can purchase CAN and FlexRay boards for the ETAS real-time PC.

3 Operation

This chapter includes information on working with ETAS RTPC V6.3.7.

The individual sections are:

- "Operating Modes" on page 21
- "ETAS RTPC in the Experiment Environment LABCAR-EE" on page 23
- "Runtime Traces" on page 24
- "CAN Trace" on page 26
- "Step-by-Step Execution of the Simulation" on page 30
- "Real-Time Plugins (RT Plugins)" on page 32
- "CPU Core Load" on page 36
- "Benchmark Indicator" on page 37
- "Testing Hardware Performance with "cyclictest"" on page 37

3.1 Operating Modes

ETAS RTPC V6.3.7 has four operating modes, one of which is always active:

- Simulation Controller
- Idle/Configuration
- Ethernet Bridge
- Power Down

Use the Web interface to configure which operating mode is to be activated when the real-time PC is switched on (see "**RTPC_POWER_UP_MODE**" on page 51).

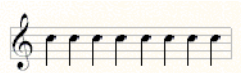
3.1.1 "Simulation Controller" Mode

The "Simulation Controller" mode is used to run real-time models. An ascending sequence of notes indicates that the simulation controller has been started successfully.



The simulation controller is activated automatically by default when the real-time PC is powered up.

Eight identically pitched notes indicate that the simulation controller cannot be started because of a configuration problem.



In this case, ETAS RTPC is automatically set to "Idle/Configuration" mode (see "'Idle/Configuration' Mode" on page 22).

Possible causes of error can be found in the chapter "Troubleshooting" on page 59.

TIP

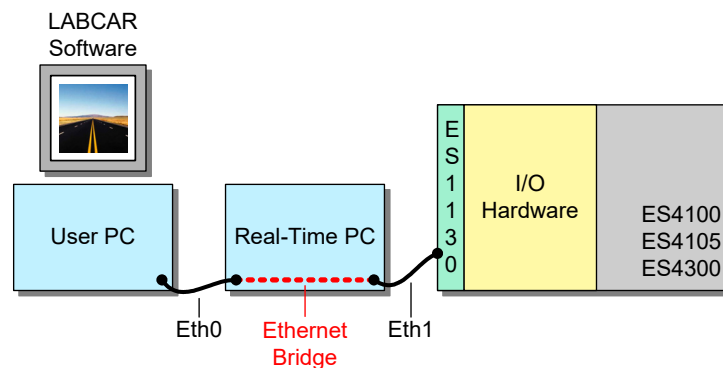
In a LABCAR system a VMEbus reset takes place every time the simulation controller is started!

3.1.2 "Idle/Configuration" Mode

The "Idle/Configuration" mode is used to configure ETAS RTPC and to update the software (Update). ETAS RTPC cannot run real-time models in this mode.

3.1.3 "Ethernet Bridge" Mode

This mode establishes a transparent connection between the two Ethernet interfaces Eth0 and Ethn. This connection enables the user to access the ES1130 Simulation Controller Board directly from the user PC; manual exchanging of Ethernet cables is thus unnecessary.



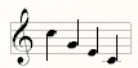
TIP

The "Ethernet Bridge" mode can be configured separately for each "Ethn" port!

The "Ethernet Bridge" mode is activated automatically for the firmware update of the I/O hardware (HSP or LABCAR firmware update).

3.1.4 "Power Down" Mode

The "Power Down" operating mode is activated either by quickly pressing the on/off switch on the front of the real-time PC or via the Web interface. ETAS RTPC shuts down the system, plays a sequence of notes (four descending notes) and switches the real-time PC off.



The real-time PC can then be disconnected from the mains.

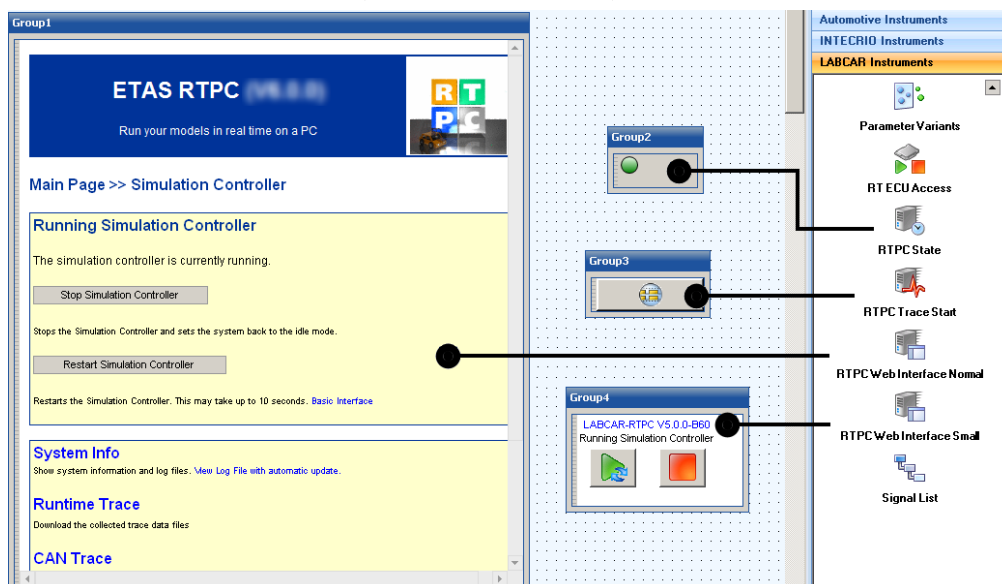
TIP

ETAS RTPC uses a robust file system, but hardware and software can still be damaged when powered off without shutting down or during a sudden voltage drop. ETAS thus recommends shutting down ETAS RTPC by briefly pressing the on/off switch to avoid these problems.

3.2 ETAS RTPC in the Experiment Environment LABCAR-EE

The experiment environment LABCAR-EE contains a range of instruments that enable access to ETAS RTPC.

These are contained in the LABCAR-EE window "Instruments" in the "LABCAR Instruments" group. They can be dragged into the main workspace from there in the usual manner ("Instrumentation" tab).



These are:

- RTPC State
A running real-time PC is indicated by a green LED
- RTPC Web Interface Normal
This instrument contains the complete representation of the web interface as in the HTML browser
- RTPC Web Interface Small
This instrument enables the simulation controller to be started (or restarted) and stopped
- RTPC Trace Start
This is used to start an RTPC trace. The files (and the access to them) are described in the section "Runtime Traces" on page 24.

3.3 Runtime Traces

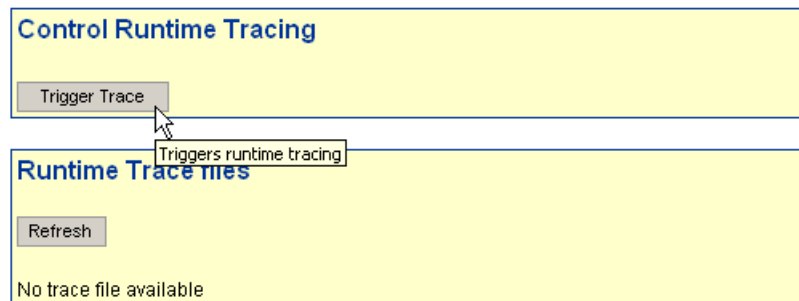
A runtime trace logs processes and tasks on the real-time PC and saves them in files with the ending .rtt which can be analyzed with RTA-TRACE and saved as a summary in the form of xml, html and csv files.

To run a runtime trace

- Select **Main Page >> Simulation Controller >> Runtime Trace**.
- Under "Control Runtime Tracing", click **Trigger Trace**.



Main Page >> Simulation Controller >> Runtime Trace



The runtime trace is launched and recorded in a file.

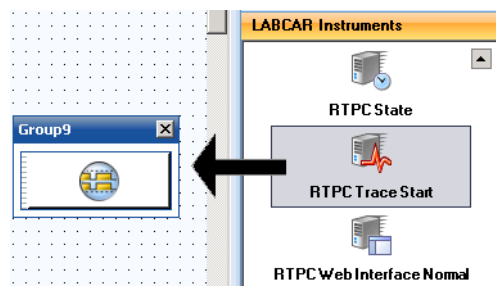
i TIP

The scope of an runtime trace (= the number of recorded events) is determined in the variable "RTOS_TRACE_BUFFERSIZE" (see "**RTPC Configuration**" on page 49).

To start a runtime trace in LABCAR-EE

You can also start runtime traces from the experiment environment (LABCAR-EE).

1. To do this, select the instrument "RTPC Trace Start" (from the "LABCAR Instruments" in the "Instruments" window) and drag it into a layer.

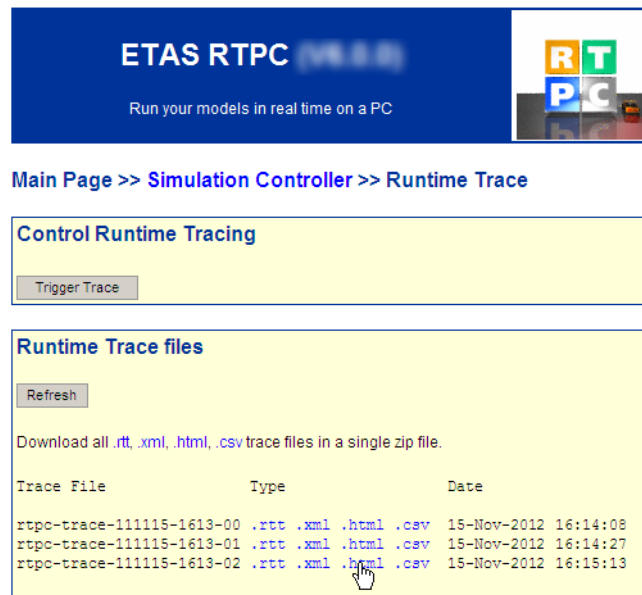


2. To start the runtime trace, click the button.

To open a file with runtime trace

The files with the results of a runtime trace are accessible at the same point:

1. Select a format next to the entry for the relevant file.



Depending on the format selected, the file is opened or a dialog box for saving the file is opened.

2. To display the HTML file of an RTPC trace, click the file ending in the "Type" column.

The file opens in the browser.

Process Runtime Statistic 2 - 15-Nov-2012 16:15:13

Task "Config" (CPU core: 0)

| Process Name | Id | Calls | RTmin | RTavg | RTmax |
|--|----|-------|-------|-------|-------|
| [Task "Config"] | 4 | 81 | 6 | 7 | 11 |
| Es4408loadchassis_ConfigCode_Config_HWCF | 10 | 81 | 0 | 0 | 1 |

All runtimes in microseconds.

Task "TaskDVEModel" (CPU core: 0)

| Process Name | Id | Calls | RTmin | RTavg | RTmax |
|---|----|-------|-------|-------|-------|
| [Task "TaskDVEModel"] | 8 | 812 | 13 | 28 | 138 |
| lcrt_OneStep_Outputs_IdleController | 17 | 812 | 2 | 4 | 82 |
| lcrt_OneStep_States_IdleController | 19 | 812 | 2 | 3 | 41 |
| ConversionModule_Execute | 16 | 812 | 1 | 3 | 16 |
| Es113x_SyncReceiveCode_TaskDVEModel_HWCF | 11 | 813 | 1 | 3 | 15 |
| Es4408ctrlStatus_TaskDVEModel_HWCF | 12 | 813 | 0 | 1 | 5 |
| Es4408ctrlPresence_TaskDVEModel_HWCF | 13 | 813 | 0 | 1 | 4 |
| Es4456rbpiezoloadStatus_TaskDVEModel_HWCF | 14 | 813 | 0 | 0 | 4 |
| Es4456rbpiezoloadPresence_TaskDVEModel_HWCF | 15 | 812 | 0 | 0 | 3 |
| Es113x_SyncSendCode_TaskDVEModel_HWCF | 18 | 812 | 0 | 0 | 3 |

All runtimes in microseconds.

The columns have the following meaning:

- Id: the internal ID of the process
 - Calls: number of calls (while the trace was active)
 - RTmin, RTavg, RTmax: minimum, average and maximum runtime of the relevant process.
3. You can also download all displayed files of a specific type as a ZIP archive. Click the relevant link in the "Download all ..." line.

3.4 CAN Trace

The CAN trace functionality enables CAN traffic to and from the IXXAT CAN cards to be monitored – the content of the CAN messages (send and receive) is transferred to a host PC as a UDP/IP stream.

For this purpose, you need a UDP monitor on your host PC, such as Netcat for Windows. The CAN trace is then started in the web interface (see "To start CAN trace in the web interface" on page 26).

Another possibility is to use the perl script provided, `labcar-rtpc-can-trace.pl` (select **CAN Trace** and click **Help**), with which the CAN trace is activated and the data output in a window of the command prompt.

To start CAN trace in the web interface

1. Launch the simulation controller in the web interface of ETAS RTPC.
2. Download your project (in the experiment environment).
3. Start the simulation.
4. Toggle to the ETAS RTPC web interface.

When the simulation controller is running, the **Main Page >> Simulation Controller** page is displayed.

5. Select **CAN Trace**.
6. Enter the following data:
 - IP Address
The IP address to which the stream is to be sent.
 - UDP Port
The UDP port to which the stream is to be sent (see "**port=n**" on page 28).
 - Ctrl Mask
The bit mask for selecting the board and the relevant controller (see "**ctrlmask=n**" on page 28).

7. Stream output mode

Currently, only "ascii" is possible.

[Main Page >> Simulation Controller >> CAN Trace](#)

CAN Trace Control

The tracing is currently stopped

Specify data stream receiver:

IP Address:

UDP Port:

Ctrl Mask:

Mode:

[Help](#)

CAN Trace Filter

Filter inactive

Specify a space-separated list of CAN ids to be added to the filter list.
An '!' before an id removes this id from the filter list.
The value "reset" deactivates the CAN trace filter.
Each start of the CAN trace resets the filter values.
Example: '0x120 80 !0x100'

[Help](#)

8. Click **Start Tracing** to send the HTTP request created with these arguments

```
http://192.168.40.14/cgi-bin/can-trace?
cmd=start&ipaddr=192.168.40.240&port=4000&ctrl-
mask=0xffff
```

to the real-time PC.

9. If you would like to filter specific CAN-IDs, you can do so under "CAN-Trace Filter".

10. To end, click **Stop Tracing**.

The Arguments of the CGI Script

This section describes the arguments possible for the "can-trace" CGI script for running an HTTP request.

- cmd=[start|stop|status]

The command to be run – this argument is mandatory.

- cmd=start
Starts the CAN trace – other arguments must be specified.
- cmd=stop
Stops the CAN trace – other arguments are not necessary.
- cmd=status
Outputs the current status of the CAN trace as a text

- `format=[html|text]`
The output format – "HTML" (default) or "Text" are possible. Specifying this argument is optional.
- `ipaddr=a.b.c.d`
The IP address (in dotted-decimal notation) to which the UDP messages are to be sent. If no value is specified, the IP address of the caller is used.
- `port=n`
The UDP port to which the UDP messages are to be sent. This port must be queried from the UDP monitor (= the counterpart). A port must be specified.
- `ctrlmask=n`
The bit mask for selecting boards (and the controller on them), whose traffic is to be transferred. A unique bit is assigned to every controller on each CAN board; calculation takes place according to the formula:

$$1 \ll (2 * \text{board number} + \text{controller number})$$

where "board number" can have values between 0 and n,
"controller number" 0 or 1.

The following table once more illustrates the assignment between board and controller and the relevant bits.

| Bit | .. | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----|---|---|---|---|---|---|---|
| Board | .. | 3 | 2 | 2 | 1 | 1 | 0 | 0 |
| Controller | .. | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

The width of the register is 32 bits; 8 CAN boards (= 16 controllers) are currently supported.

The value for "ctrlmask" must be specified as a decimal or hexadecimal number (starting with "0x"). If no value is specified, the traffic on every controller is recorded.

Example: If the first and second controllers of the third board are to be monitored, bits 4 and 5 of the register are set - the value of the bit mask is then $2^4 + 2^5 = 16 + 32 = 48$ (0x00030).

Description of the Answer

The CAN messages are transferred as pure ASCII code in UDP frames; each individual message is terminated by the character for a line feed \n (ASCII: 0x0A).

A typical output thus has the following format:

```
....
4101 1373.0778 1355.6203 E:0 CAN:0,0 RX ID: 7d DLC:8 DATA: 40 a3
02 03 00 e0 00 00
4102 1373.0789 -.---- E:0 CAN:0,1 TX ID: 7b DLC:8 DATA: 70 96
04 08 04 f0 ff 08
....
```

The meaning of the elements of an answer line is as follows:

- Line number
Using the consecutive line numbering, the completeness of the recording (= no streaming errors, see "E:n" below) can be verified.

- Two time stamps
The first value in the line specifies the time at which the CAN message was registered by the simulation model (on the real-time PC) in seconds.
The second value is a hardware time stamp of the CAN board and indicates when a receive message was received. Naturally, this value does not occur with send messages.
- E:n
"E:0" means: no error.
"E:1" shows a streaming error resulting from too high a load on the CAN-bus which can lead to partial data loss.
- CAN:x,y
Contains the number of the CAN board x (0...n) and the number of the controller y on it (0,1)
- RX/TX
Describes the direction of a message from the point of view of the real-time PC (TX = send message, RX = receive message)
- ID: nn
The hex identifier of the CAN message
- DLC: n
The data length code of the CAN message
- DATA: XX XX XX XX XX XX XX XX
The 8 data bytes of the CAN message (in hex notation)

3.5 Step-by-Step Execution of the Simulation

The "Model Step" function enables you to pause the automatic simulation and then run the simulation as individual steps (in the form of tasks and processes).

To do this, proceed as follows:

1. Start the simulation.
2. In the web interface, select the **Model Step** link.

The "Model Step" page is displayed.



To pause automatic simulation



3. To pause the simulation, click the **Pause** icon.

Automatic simulation execution stops.

To select the acquisition task for the display in LABCAR-EE

- If you follow the step-by-step execution of the simulation in one or more instruments, select the task that is also used for updating the instruments under "Acquisition Task". "Acquisition" is selected here by default.

Regardless of the model step executed manually, the Acquisition Task is always run after it and the display instruments in the experiment environment thus updated.

To run an individual step (task or process)

- In the list, click a task (or a process) you want to run.

```
T5,Manager
T6,Analyze
T7,Acknowledge
T8,TaskDVEModel
  P11,CAN_Receive_board0_controller0
  P12,CAN_Receive_board0_controller1
  P13,Es113x_SyncReceiveCode_TaskDVEModel_HWCF
  P14,Pb1651adclAnaIn_TaskDVEModel_HWCF
  P15,lcr1_OneStep_Outputs_IdleController
  P16,Pb1651adclLedDrv_TaskDVEModel_HWCL
  P17,Es113x_SyncSendCode_TaskDVEModel_HWCF
```

The task (process) is run.

The task (process) last run is highlighted in yellow.

```
T6,Analyze
T7,Acknowledge
T8,TaskDVEModel
  P11,CAN_Receive_board0_controller0
  P12,CAN_Receive_board0_controller1
  P13,Es113x_SyncReceiveCode_TaskDVEModel_HW
```

To run a sequence of tasks or processes

- You can run a sequence of tasks or processes you enter in the "Sequence" box.

Sequence

| | |
|---|--------------|
| <input type="text"/> | Run Sequence |
| <p>The sequence may be entered as comma separated list of actions. Possible actions: Tx - Run task x, Px - Run process x, Rx - Realtimetest x, Hx - Pluginhook x. Lists can be specified: Tx-y - Run tasks x-y, Tx*4 - Run task x four times.</p> | |

Individual components of the sequence are separated by commas.

Components of a sequence are:

- Tx: run task x
- Px: run process x
- Tx-y: run task x to task y
- Tx*4: run task x four times in succession.

- To run the sequence, click **Run Sequence**.

The sequence is run.

To continue automatic simulation



- To continue the automatic running of the simulation, click the **Continue** icon.

Simulation is continued.

3.6 Real-Time Plugins (RT Plugins)

RT plugins can be used to load code dynamically at runtime. This allows the user to extend real-time models dynamically.

This section contains information on:

- "Preparation of the LABCAR-OPERATOR Project" on page 32
- "The Plugin Files" on page 33
- "Special Preprocessor Directives" on page 34
- "Templates and Examples" on page 35
- "Procedure" on page 35

3.6.1 Preparation of the LABCAR-OPERATOR Project

The project has to be prepared to work with real-time plugins. "Hooks" have to be placed at suitable positions within the process list. These hooks define the calling context used for executing the dynamically loaded code. The hooks are typically defined within a C code module.

Hooks API

All functions are defined in the C file header `rtos_realtime_test.h`, so please include this header with

```
#include "rtos_realtime_test.h"
```

Types:

```
rtos_handle_t
```

A generic handle type that is used to address the hooks.

Example:

```
static rtos_handle_t rttest_hook;
```

Init Code

```
rtos_handle_t realtime_test_hook_create(  
                                const char *hookname);
```

This function creates a named hook and returns a handle.

Example:

```
rttest_hook = realtime_test_hook_create("SGPlugin");
```

Exit Code

```
int realtime_test_hook_delete(  
                                const rtos_handle_t hook);
```

This function deletes a previously created hook.

Example:

```
realtime_test_hook_delete(rttest_hook);
```

Real-time Code

```
void realtime_test_hook_execute(  
                                const rtos_handle_t hook,  
                                void *data);
```



```
const rtos_handle_t hook);
```

This function calls the plugin functions that are attached to the hook. It defines the calling context and the exact calling position. The call to this function should be placed within a suitable periodic task.

Example:

```
realtime_test_hook_execute(rttest_hook);
```

3.6.2 The Plugin Files

The plugins to be loaded have to be defined as C code and header files on the user PC. All files have to be transferred to ETAS RTPC using a suitable tool (see "Preparing to Download Plugins" on page 35).

Basic Plugin Initialization

Each plugin must implement the four basic plugin functions

```
int on_load(void);
int on_initialize(void);
void on_terminate(void);
void on_unload(void);
```

The functions `on_load` and `on_initialize` are called while the plugin is loaded. Both functions have to return 0 when successful. The functions `on_terminate` and `on_unload` are called when the plugin is unloaded.

The function `on_initialize` is typically used to call API functions that attach plugin functions to existing plugin hooks.

For this, a static interface object of type "I_realtime_test" has to be implemented. This object holds pointers to the callback functions that are called in a real-time context.

The type "I_realtime_test" is defined as follows:

```
typedef struct I_realtime_test_struct {
    int (*init)(realtime_test_obj *obj);
    void (*exit)(realtime_test_obj *obj);
    void (*execute)(realtime_test_obj *obj, rtos_time_t ns);
    void (*background)(realtime_test_obj *obj);
}
I_realtime_test;
```

`init` is called at the initialization time of the real-time test object; `exit` is called when the real-time test object is to be deleted.

`execute` is called from the hook in a real-time context, and `background` is called periodically (approx. every 0.5 seconds).

Creating the Real-Time Test Object

The interface object (type `I_realtime_test`) is used to create a real-time test object. For this, the pointer to the interface object as well as an (optional) name and an (optional) user pointer might be used.

The function

```
realtime_test_obj *realtime_test_create(
    const char *name, const I_realtime_test *interface,
    void *data, const size_t resultstring_size);
```

is used to create a real-time test object.

Attaching a Real-Time Test Object to a Hook

The real-time test object can be attached to an existing hook. This is done by calling

```
int realtime_test_hook_add(
    const char *hookname, realtime_test_obj *obj);
```

Detaching a Real-Time Test Object from a Hook

To remove a real-time test object from a hook, use the

```
int realtime_test_hook_remove(
    const char *hookname, realtime_test_obj *obj);
```

function.

Deleting a Real-Time Test Object

The function

```
int realtime_test_hook_delete(
    const rtos_handle_t hook);
```

deletes an existing real-time object.

3.6.3 Special Preprocessor Directives

A number of additional preprocessor directives are available with the plugin files. They are intended to ease the interaction with the real-time executable currently running.

The following directives can be used:

#label

- `#label name path`

Generates a define "name" that represents the access to the element specified by "path".

#labelstruct

- `#labelstruct name path`

Generates a define "name" that represents the base type (the final .xxx omitted) of the element specified by "path".

#labeltype

- #labeltype name path

Generates a define "name" that represents the (scalar) C base type of the element specified by "path".

#a2laddr

- #a2laddr name path

Generates a define "name" that holds the A2L pseudo address as a string. This string can be used directly with the function `rtpc_a2l_addr_value_ptr()` that is defined in the header file `a2l_address_resolution.h`.

3.6.4 Templates and Examples

There is an example of a C code file for implementing hooks (`PluginHook.c`) on your real-time PC under <http://192.168.40.14/tools/rtpplugin/templates/>.

You will also find a template for a plugin there (`plugin-template.c`).

3.6.5 Procedure**Preparing to Download Plugins**

A tool you can use to download the plugins to the real-time PC is available in the following directory:

<http://192.168.40.14/tools/rtpplugin/RTPCRuntimeAccess/>

It contains two DLLs and two executables:

- `EE.RealTimePlugin.Core.Interfaces.dll`
- `EE.RealTimePlugin.Core.dll`
- `RTPC-Plugin-GUI.exe` (version with GUI)
- `RTPC-Plugin.exe` (instruction line version)

To be able to use this tool, you must copy both DLLs and one of the executables to a directory on your user PC.

To integrate hooks in a project (LABCAR-IP)

The file named in "Templates and Examples" on page 35 (`PluginHook.c`) enables you to generate a hook using a template.

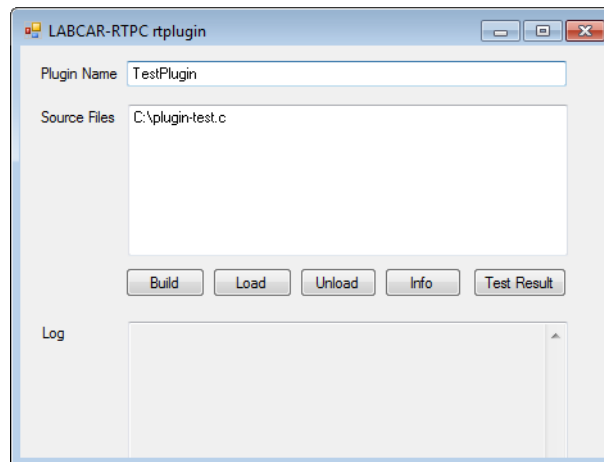
1. Generate an new C code module in LABCAR-IP
2. Add the relevant code.
3. Add a process.
4. Generate the code (**Project** → **Build**).
5. Toggle to the "OS Configuration" tab and assign the new process to the desired task.

To run the plugin (LABCAR-EE)

6. Open your project in the experiment environment and run it.

7. Start the RTPC-Plugin-GUI.exe file named in "Preparing to Download Plugins" on page 35.

The following GUI opens.



8. Under "Plugin Name" enter a name for the subsequent list of source files.
9. Enter the source files you want to belong to this plugin under "Source Files".
10. Alternatively you can drag files to this field using Drag & Drop.
11. Select **Build** to load the files to the real-time PC and to generate code there.
12. Select **Load** to load the plugin into the experiment you have running.
13. Select **Unload** to stop running the plugin.

3.7 CPU Core Load

The distribution of the process load to the processor cores can be issued in an XML file using

[http://192.168.40.14/cgi-bin/cpu-load\[?detail=n\]](http://192.168.40.14/cgi-bin/cpu-load[?detail=n])

Depending on the degree of detail, the XML file contains different information:

- detail=0 (default)

This only issues the CPU load (in %):

```
<?xml version="1.0" encoding="UTF-8" ?>
- <load>
  <cpuload cpu="0" load="73.9" loadunit="%" />
</load>
```

- detail=1

The total load and the distribution to the individual simulation tasks are issued.

- detail=2

In addition, all real-time processes are issued.

- detail=3
In addition, all non-real-time processes are issued.
- detail=4
In addition, all commands are issued.

3.8 Benchmark Indicator

To compare the simulation performance of ETAS RTPC on different PCs, a standardized calculating routine can be used.

Include the header file (in /opt/etas/include)

```
rtpc-benchmark.h
```

in any C code module and run one of the following routines:

```
double rtpc_benchmark_standard1(double salt);
double rtpc_benchmark_standard2(double salt);
```

The first routine calculates 2000 variables with 8000 instructions, the second 50 variables with 400 instructions – arguments and return values have no significance.

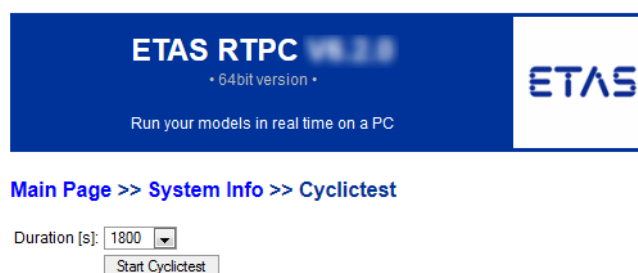
3.9 Testing Hardware Performance with "cyclicttest"

The "cyclicttest" application is used to test how good specific PC hardware is for real-time applications. This involves the measuring of the interrupt latency between the activation of the timer and the running of the relevant code.

To launch Cyclicttest

1. In the web interface, select **Main Page >> System Info** and scroll to **Cycle Test - Test Program**.
2. Click **Cyclicttest Control / Results**.

The program can be launched on the following page. Results of tests previously executed are shown below this.



3. Select "Duration" and click **Start Cycletest**.

Example:

A timer is triggered periodically every 100 ms and then a test is carried out to see when the relevant Software routine is actually run: The timestamp from the start of the routine is compared to the timestamp when the timer was triggered.

Results of executed Cyclictests on this RTPC

0: Wed Sep 30 06:19:19 2015

```
# /dev/cpu_dma_latency set to 0us
T: 0 ( 7098) P:99 I:100 C: 99982 Min: 1 Act: 1 Avg: 1 Max: 26
T: 1 ( 7099) P:99 I:110 C: 90910 Min: 1 Act: 1 Avg: 1 Max: 4
T: 2 ( 7100) P:99 I:120 C: 83334 Min: 1 Act: 1 Avg: 1 Max: 5
T: 3 ( 7101) P:99 I:130 C: 76924 Min: 1 Act: 1 Avg: 1 Max: 6
T: 4 ( 7102) P:99 I:140 C: 71413 Min: 1 Act: 2 Avg: 1 Max: 5
T: 5 ( 7103) P:99 I:150 C: 66633 Min: 1 Act: 2 Avg: 1 Max: 3
T: 6 ( 7104) P:99 I:160 C: 62468 Min: 1 Act: 2 Avg: 1 Max: 2
T: 7 ( 7105) P:99 I:170 C: 58794 Min: 1 Act: 2 Avg: 1 Max: 4
```

1: Wed Sep 30 05:56:48 2015

```
# /dev/cpu_dma_latency set to 0us
T: 0 ( 6106) P:99 I:100 C: 599981 Min: 1 Act: 1 Avg: 1 Max: 34
T: 1 ( 6107) P:99 I:110 C: 545455 Min: 1 Act: 1 Avg: 1 Max: 6
T: 2 ( 6108) P:99 I:120 C: 499993 Min: 1 Act: 2 Avg: 1 Max: 6
T: 3 ( 6109) P:99 I:130 C: 461524 Min: 1 Act: 1 Avg: 1 Max: 3
T: 4 ( 6110) P:99 I:140 C: 428553 Min: 1 Act: 2 Avg: 1 Max: 55
T: 5 ( 6111) P:99 I:150 C: 399972 Min: 1 Act: 2 Avg: 1 Max: 3
T: 6 ( 6112) P:99 I:160 C: 374966 Min: 1 Act: 1 Avg: 1 Max: 3
T: 7 ( 6113) P:99 I:170 C: 352910 Min: 1 Act: 2 Avg: 1 Max: 3
```

The meaning of the data issued is as follows:

- T
Cyclictest starts a periodical task T_n per core n , i.e. task T_0 runs on core 0, task T_1 on core 1 etc. The number in brackets represents the process ID of the relevant task.
- P
The priority of the task (here: fixed 99)
- I
The invoke interval of the task in μs (here: starting with 100 μs , becoming larger in increments of 10 μs)
- C
The number of task calls during the test run
- Min, Act, Avg, Max
The values are the latencies to the timer trigger point in μs . "Act" is the current value - statistics are also created (Min/Avg/Max).
A value of 15 for "Max" means that it took 15 μs for the software to react to a timer event. When "Max" is 45 μs it is thus hardly possible to use this PC to calculate a 50 μs task.
The "Max" values are shown in color (with increased latencies):
 - Max $\leq 30 \mu\text{s}$: no color
 - $30 \mu\text{s} < \text{Max} \leq 50 \mu\text{s}$: yellow
 - Max $> 50 \mu\text{s}$: red

3.10 CAN/CANFD-Abstraction and LIN-Abstraction

3.10.1 General Functional Principle

If a project is used on LABCAR systems with differently developed CAN/CANFD- and LIN interfaces without the CAN/CANFD and LIN configuration having to be changed, the feature of the CAN/CANFD and LIN abstraction (General bus abstraction) can be used. This feature is used in conjunction with the RTPC with the CAN and LIN editors or the network module.

The assignment of combinations from board-ID and controller-ID to channels is essential for the bus abstraction. This mapping is shown on the RTPC in the form of ini-files. These files are divided into sections. The sections contain pairs of a variable name and a value, which are connected by the equal sign.

At the bus abstraction the section describes the bus type as well as the number of the canal. Valid identifiers for bus types are `can`, `canfd` and `lin`. Channels are defined by any natural numbers. The order of the sections in an ini-file is arbitrary. Combinations of board-ID and controller-ID are formed by the variables `boardid` and `controllerid` within the associated section. The values of these variables are natural numbers.

Examples:

```
[can0]
boardid=0
controllerid=1
```

```
[canfd12]
boardid=1
controllerid=0
```

```
[lin4]
boardid=0
controllerid=0
```

The first example describes the assignment of the second CAN controller, of the first board to channel 0. The board with the ID 1 in the second example is CANFD capable. For this reason, the first controller is assigned on the CANFD channel 12. The first LIN controller of the first board is referred to as LIN channel 4.



The numbering of the LIN controllers starts with "0" for each board regardless of how many CAN controllers are present on the board.

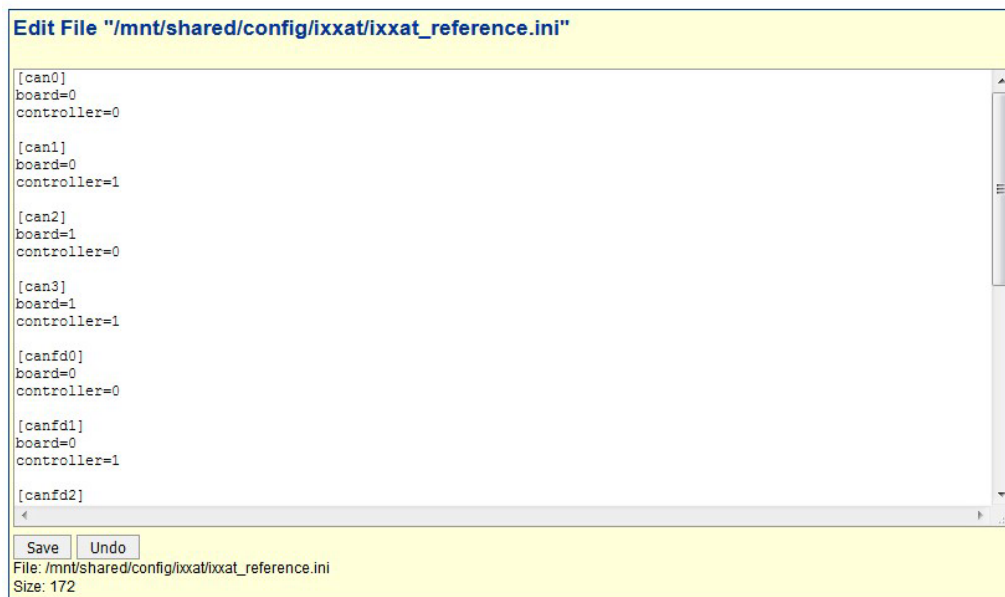
The bus abstraction is exclusively performed on the RTPC. The code generated by the CAN and LIN editors or the network module is modified before the build process. The combinations of board ID and controller ID that are configured in the editors or the network module are assigned to the corresponding channels. Then the built project is neutral with respect to the CAN/CANFD and LIN interfaces as it internally uses the channels. At runtime, concrete interfaces are addressed via the appropriate combination of board ID and controller ID from the information about the channels.

3.10.2 Abstraction of the Bus Interfaces During the Build Process

During the build process, an existing configuration of CAN/CANFD and LIN interfaces is converted into a general configuration. This general configuration can then be used on different LABCAR systems without the project having to be adjusted. The transformation is based on a reference configuration that must be present on the RTPC on which the project is being built. The reference configuration can be edited using the web interface. All CAN/CANFD and LIN interfaces that are listed in the reference configuration will be replaced by the respective channels when building the project.

Missing combinations of board ID and controller ID are ignored and not replaced by channels.

The editor for the reference configuration is available in the web interface of the RTPC as follows **Configuration → CAN/LIN Abstraction Configuration → Expert Edit**



© 2003-2017 ETAS GmbH

The bus interfaces are still described with the board ID and the controller ID in the CAN and LIN editors or the network module. The conversion into an abstract description with channels takes place during the building of the project on the basis of the reference configuration.

3.10.3 Converting the Channel Description into Concrete Bus Interfaces at Runtime

After downloading the project to the simulation RTPC, the back transformation of the abstract CAN/CANFD and LIN channels into the corresponding combinations of board ID and controller ID takes place. The runtime configuration is used for this. The runtime configuration can be edited using the web interface. Combinations of controller ID and board ID that were not replaced by channels when the project was built remain in place and are addressed directly at runtime.

The editor for the runtime configuration is available in the web interface of the RTPC as follows **Configuration → CAN/LIN Abstraction Configuration → Expert Edit**

Configuration >> Expert Edit



3.10.4 Template for a Channel Configuration

Each time the RTPC boots, a template for the configuration files is generated. This template does not overwrite the existing configuration files. It can be used as an overview of the installed CAN/CANFD and LIN hardware.

The template of the configuration file can be downloaded in the web interface as follows **Configuration → CAN/LIN Abstraction Configuration → Download template**



An example for a configuration:

```
; 0000000000000000: iPC-I XC16/PCI, 2 CAN, 0 LIN  
; HW473609: CAN-IB200/PCIE, 2 CAN, 0 LIN  
; HW471605: CAN-IB600/PCIE, 2 CAN/CAN-FD, 0 LIN
```

```
[can0]  
board=0  
controller=0
```

```
[can1]  
board=0  
controller=1
```

```
[can2]  
board=1  
controller=0
```

```
[can3]  
board=1  
controller=1
```

```
[can4]  
board=2  
controller=0
```

```
[can5]  
board=2  
controller=1
```

```
[canfd0]  
board=2  
controller=0
```

```
[canfd1]  
board=2  
controller=1
```

The example shows a configuration with one XC16, one IB200 and one IB600 card each. The variable `RTOS_IXXAT_CAN_DRIVER_ORDER` is set so that the XC16 card is detected before the IB200 card and IB600 card. The controllers of the IB600 card are CAN-FD-capable, but they can also be used as standard CAN controllers. For this reason, the corresponding combinations of board ID and controller ID are listed both as CAN channels (`can4` and `can5`) as well as CANFD channels (`canfd0` and `canfd1`).

4 Configuration

This chapter describes the uses and configuration of ETAS RTPC with the web interface.

4.1 Applications

There are three applications for ETAS RTPC:

- A Model-in-the-Loop (MiL)
- B Hardware-in-the-Loop (HiL)
- C Hardware-in-the-Loop with Measuring and Calibration Access (M&C)

Depending on the application, the „RTPC_USAGE_ETH1“ parameter has to be set in a specific way (in the web interface under **Main Page → Configuration** (see “Configuration” on page 49).

4.1.1 Model-in-the-Loop (MiL)

In MiL tests, the Unit-under-Test consists of a model of a controller functionality written in software. No I/O hardware is necessary.

The following setting has to be made:

| Parameter | Setting |
|-----------------|---------|
| RTPC_USAGE_ETH1 | Off |
| ... | ... |
| RTPC_USAGE_ETHn | Off |

Tab. 4-1 Setting for MiL Use

4.1.2 Hardware-in-the-Loop (HiL)

HiL tests are the standard application for ETAS RTPC. In HiL operation, the Unit-under-Test exists as hardware. It is connected to the test system via I/O boards.

The real-time PC addresses the I/O hardware via an Ethernet connection. Normally, the Eth1 (HW) port of the real-time PC is used.

The following setting has to be made:

| Parameter | Setting |
|-----------------|---------|
| RTPC_USAGE_ETH1 | ES1130 |

Tab. 4-2 Setting for HiL Use

4.1.3 Hardware-in-the-Loop with Measuring and Calibration Access (M&C)

With "HiL with measuring and calibration access", it is possible to access the UuT simultaneously during test operation using serial protocols (e.g. ETK or XCP).

Measuring and calibration access can be realized using a compact ES590 ECU interface module. The ES590 and RTPC are connected directly to the user PC via an ES600 network module.

The following setting has to be made:

| Parameter | Setting |
|-----------------|---------|
| RTPC_USAGE_ETH1 | ES1130 |

Tab. 4-3 Settings for HiL Use with Measuring and Calibration

4.1.4 Hardware-in-the-Loop with Load and Error Simulation

If an ES4408 system is used to simulate loads or an ES4440 Compact Failure Simulation Module to simulate errors, these modules communicate with the real-time PC via Realtime UDP (rtudp).

The following settings then have to be made for the relevant Ethernet ports on the real-time PC:

| Parameter | Setting |
|-----------------|---------------------|
| RTPC_USAGE_ETHn | rtudp_x (x = 0...4) |

Tab. 4-4 Settings for Systems for Load or Error Simulation

4.1.5 Massive I/O

Massive I/O means that the HiL system can be assigned a large number of I/O channels. With ETAS RTPC V6.3.7, this means that up to eight VME/VXI I/O systems (ES4100, ES4105, ES4300) are supported.

Each system is controlled by a System Controller Board. To ensure real-time data transfer, there is a point-to-point connection between the real-time PC and each I/O system. The DVE model thus has access to all I/O boards regardless of the system the individual board is contained in.

The following settings have to be made:

| Parameter | Setting |
|-----------------|--|
| RTPC_USAGE_ETHn | ES1130, ES1130_1, ES1130_2 or ES1130_3 |

Tab. 4-5 Settings when Using Several I/O Systems (Massive I/O)

4.2 The Web Interface

You can do the following via the ETAS RTPC web interface:

- configure ETAS RTPC
- install software updates
- display status messages and system information

The access to the web interface, navigation within the interface and the functions are described in this section.

4.2.1 Access to the Web Interface

You can access the web interface of ETAS RTPC V6.3.7 via the URL

`http://192.168.40.14`

Simply launch a web browser on the user PC and navigate to this address.

Instead of the IP address `192.168.40.14`, you can also use the alias "rtpc" if you declare it in the Windows system file `hosts`. Depending on the operating system, this file is in one of the following directories:

| Operating System | Directory Path |
|------------------|--|
| Windows XP | <code>C:\Windows\system32\drivers\etc\hosts</code> |
| Windows 7 | <code>C:\Windows\System32\drivers\etc\hosts</code> |
| Windows 10 | <code>C:\Windows\System32\drivers\etc\hosts</code> |

Add the following line to the existing entries:

```
192.168.40.14    rtpc
```

This makes it possible to reach the web interface of ETAS RTPC via the URL `http://rtpc`.

4.2.2 Navigation

The navigation line directly under the blue headline is an orientation guide in the web interface (see Fig. 4-1).



Fig. 4-1 The Web Interface with Running Simulation Controller

You can also return (when the simulation controller is stopped) to higher-order pages, e.g. the "Main Page", quickly using the navigation line.

Not all functions of the web interface can be invoked in the operating modes "Simulation Controller" and "Ethernet Bridge". Once the web interface has been started, you are automatically led to the relevant subpages.

The following functions are available when the simulation controller is running (see Fig. 4-1 on page 47):

- **System Info**
This is where you obtain information on the system and find log files. For more details, refer to the section "System Info" on page 56.
- **Runtime Trace**
This is where you can start runtime traces in which processes and tasks are logged on the real-time PC. For more details, refer to the section "Runtime Traces" on page 24.
- **CAN Trace**
The CAN Trace functionality enables the monitoring of CAN traffic to and from the IXXAT CAN cards. This functionality is described in the section "Runtime Traces" on page 24.

- **Model Step**

This function enables step-by-step execution of the simulation. For more details, refer to the section “Step-by-Step Execution of the Simulation” on page 30

- **Power Control**

This is where you can shut down or reboot the system (see “Power Control” on page 49).

4.2.3 Functions

The following figure shows an overview of the ETAS RTPC web interface structure and how the different web pages can be accessed in the relevant operating modes.

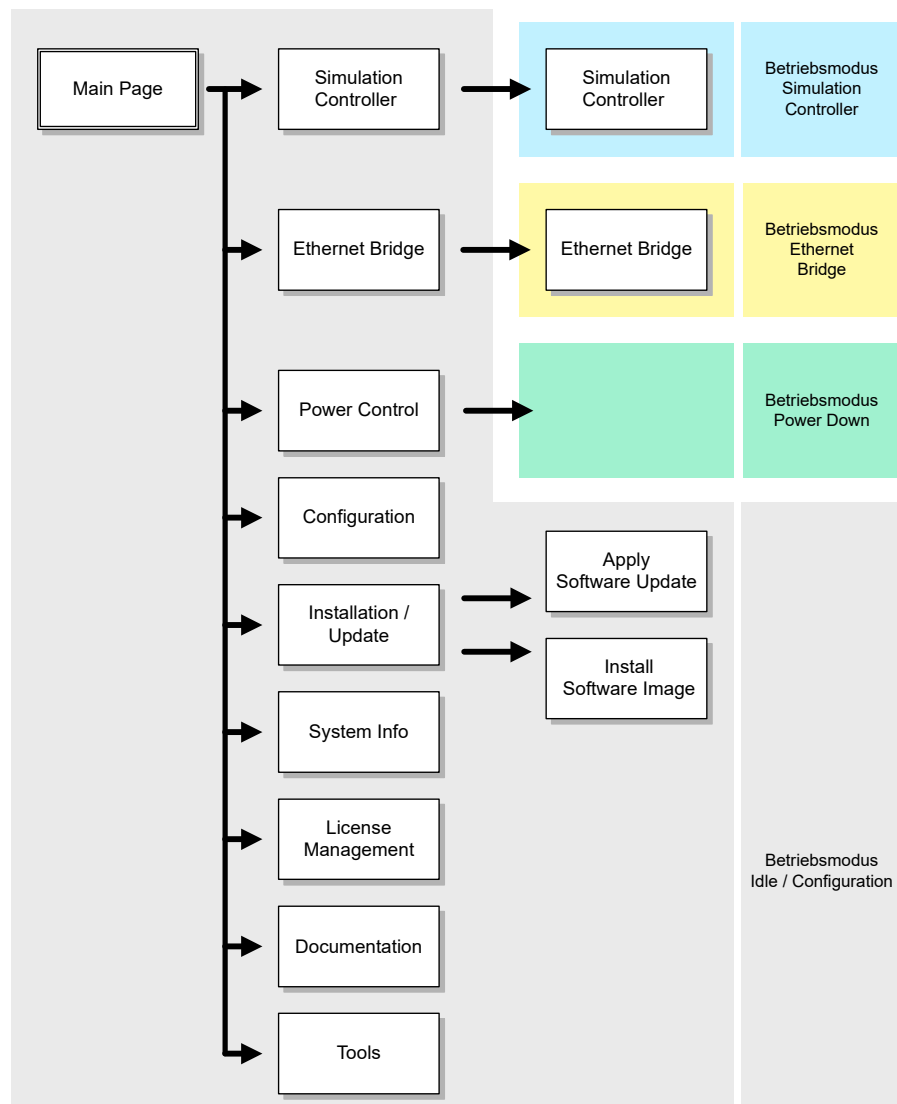


Fig. 4-2 Structure of the Web Interface

The individual pages are described in detail below.

4.2.3.1 Main Page

All functions of the ETAS RTPC web interface can be selected via the Main Page.

4.2.3.2 Simulation Controller, Ethernet Bridge

These menu items are used to start the relevant operating modes "Simulation Controller" and "Ethernet Bridge". The operating modes have already been described in the section "Operating Modes" on page 21.

Once an operating mode has been activated, only the relevant subpages can be accessed.

4.2.3.3 Power Control

If the on/off switch on the real-time PC is not accessible or does not exist, ETAS RTPC can be shut down or rebooted using this web interface link.

To boot ETAS RTPC via the network ("Wake-On-LAN"), you require a corresponding tool on the relevant network PC to send the "magic packets" (e.g. the free tool "WOL - Magic Packet Sender 2007"). Data on how to configure this tool can be found under "Wake On LAN Settings".

4.2.3.4 Configuration

This is the configuration dialog of ETAS RTPC which consists of four parts:

- Configuration Support
 - Automatic ES1130 Configuration
Automatically configures the Ethernet connection(s) (Eth1...Ethx). The real-time Ethernet connection is checked and – if an ES1130 is connected – the "RTPC_USAGE_ETH1" parameter is set to the value "es1130".



TIP

The "RTPC_USAGE_ETHn" parameter is only displayed if the relevant Ethernet interface has also been configured.

Refer to "To run the configuration" on page 15 for a description of autoconfiguration.

- ES1130 Detection
Checks whether the connection to the ES1130 works.
- RTPC Configuration
The "RTPC Configuration" section is used for setting the configuration mode and activating the debug options.
In section „Host Ethernet Configuration (ETH0)“ the user PC (connected to Ethernet adapter ETH0) ist configured.
 - IP Address
This is where you can specify the IP address (IPv4) of the Ethernet adapter used to connect to the host PC (Default: 192.168.40.14). We seriously recommend that you use an IP address for a private Class-C network (Address: 192.168.x.y, netmask: 255.255.255.0).

- Netmask
The netmask for ETH0
- DHCP
This is where you can specify whether the IP address of ETH0 should be obtained via DHCP or not.
- Ethernet Negotiate
This is where you can specify whether the controllers involved can independently negotiate and configure the maximum possible transfer speed and the duplex procedure with each other ("auto") or manually select certain settings.

In section "Realtime Ethernet Configuration" additional realtime Ethernet connections are configured.

**TIP**

To identify the card assigned to the Ethernet interface "Eth_n", use the mouse to click *Blink*.

- Usage
"ETH1", "ETH2" etc. can be used to connect an ES1130. If an ES1130 Simulation Controller Board is connected, select "es1130".
The settings "es1130_n" are intended for additional ES1130 Simulation Controller Boards that can be connected; the setting "rtudp_n" is for real-time communication with appropriate ETAS hardware (such as the ES4440 Compact Failure Simulation Module or the ES4408 Load Chassis). Details on the real-time UDP (rtudp) driver API can be found on the web interface at <http://192.168.40.14/api/rtudp.html>.
Individual Ethernet connections can be configured in Bridge mode here (see section 3.1.3 on page 22).
- IP Address
This is where you can specify the IP addresses (IPv4) of additional Ethernet adapters (ETH1, ETH2, ...) which connect the Real-Time PC with other devices (e.g. 192.168.100.23).
We seriously recommend that you use an IP address for a private Class-C network
(Address: 192.168.x.y, netmask: 255.255.255.0).

- Ethernet Negotiate

This is where you can specify whether the controllers involved can independently negotiate and configure the maximum possible transfer speed and the duplex procedure with each other ("auto") or manually select certain settings.

**TIP**

To change the designation "ETHn" of a board, activate the option "Show options to control...". Then you can assign (column "Ethernet Order") a certain board (using the boards MAC address).

**TIP**

If the option "auto" has been selected for "Ethernet Order", a new installation can result in a change of the "MAC Address:ETHn" assignment. To freeze the current assignment, select *Freeze current Ethernet Order*.

Section "General Parameters":

- RTPC_POWER_UP_MODE

This parameter specifies the operating mode for ETAS RTPC after start.

The following values can be set:

idle: ETAS RTPC starts in "idle" mode. This mode is used for configuration and system update.

simulate: ETAS RTPC starts in simulation mode. ETAS RTPC cannot be configured in this mode.

bridge: ETAS RTPC starts in "Ethernet Bridge" mode, i.e. the Ethernet interfaces of ETAS RTPC are connected logically. ETAS RTPC cannot be configured in this mode.

- RTPC_LOG_LEVEL

Events are logged and written to a file during ETAS RTPC operation.

The level for the events to be logged can be set here – "debug" logs all events, "emerg" only logs those that make the system unusable. A useful setting for normal operation is "warning" or "error".

- RTPC_NAME

This is where the particular real-time PC can be assigned a name.

This name is helpful if the user is working with several ETAS RTPC systems - it is shown both on all web interface pages and in the browser title.

- RTPC_TIMEZONE

Defines the time zone

- RTPC_PARALLEL_MAKE

Enables the parallel execution of the GNU make process (i.e. simultaneously on several CPU cores).

The feature is disabled with "no" and GNU make is only called up for one job. The feature is enabled with "auto" providing the real-time PC permits. When "force" is set, this feature is always used.

– RTPC_TASK_MONITOR_0/1

The task monitor makes it possible to monitor two tasks during run-time. The task to be monitored can be selected here by its number.

While the task is active, the relevant bit (0 or 1) at the LPT interface is set to "High". This can be monitored using an oscilloscope. Note that using the task monitor causes a processor load on the ETAS RTPC.

To deactivate the task monitor, set the values to "-1".

The following table lists the assignment of the relevant pins of the 25-pin DSUB connector, LPT.

| Pin | Function |
|---------|------------|
| 2 | Data bit 0 |
| 3 | Data bit 1 |
| 18 - 25 | Ground |

– RTPC_TASK_TIMING_STATISTIC

This parameter defines the recording frequency (in seconds) for the statistics of the task runtimes. "0" deactivates recording.

The statistics contain the following information:

- minimum/maximum time dt (in ns) between two task activations
- minimum/average/maximum runtime (in ns) of the task



TIP

The statistics are not output in real-time mode to prevent any disturbances of real-time behavior. The measuring functions are not implemented atomically. Individual events may therefore go missing.



TIP

The recorded events depend on the value of the "RTPC_LOG_LEVEL" parameter (see Page 51).

– RTPC_INVALID_CPU_CORE_HANDLING

This parameter determines how to handle an invalid specification of CPU cores (in the "OS Configuration" tab in LABCAR-IP). Possible values are "warn" (default) and "stop".

"warn": If, in the OS Configuration, non-existent cores are specified, a warning is issued in the ETAS RTPC log file and all real-time tasks are run on a single CPU core.

"stop": In the case of an invalid specification of CPU cores, the real-time model does not start and an error message is issued in the ETAS RTPC log file.

– RTOS_TASK_STACKSIZE

ETAS RTPC uses a separate stack for each running task. The size of this stack (in kB = 1024 Byte) can be defined using this parameter. A useful value is 1024 kB.



TIP

If models with lots of tasks are used, the stack size may have to be reduced due to the limited size of the RAM.

– RTOS_TRACE_BUFFERSIZE

ETAS RTPC has an automatic internal serial protocol which is useful when, for example, tests are carried out to assess the performance of simulation models.

This parameter is used to determine the size of the buffer (in the unit "number of events") before the data is written to a protocol file (*.rtt).

This file can be accessed using **Main Page → Simulation**

Controller → Trace File Access and then displayed and analyzed with RTA-TRACE.

– RTPC_COMPILE_OPTIMIZATION

ETAS RTPC uses the GNU C Compiler "gcc" – code generation optimization can be set in 4 steps (0 ... 3) which correspond to the gcc compiler options -O0 ... -O3. The larger the number, the better the optimization - this does, however, also mean that more time is required for compilation.

If you are not sure about this setting, select "1" or "2".

– RTPC_COMPILE_LINK_DEBUG

If you set this parameter to "yes", the compiled object files and the executable file are assigned debug information. This can be useful when detecting errors in the simulation model (see "**Debug/Disassemble**" on page 56).



TIP

Only set this option if it is absolutely necessary as the executable file becomes very large as a result.

• RTPC_COMPILE_SWAPSPACE_MB

When using large Simulink models, the C files generated can be very large. This can lead to the computer not having enough disk space available.

This parameter enables you to make additional disk space available for the compilation phase. This space is in the `compile.swapfile` file and is only used for the compilation of the model, not for the simulation.

Possible values are 0, 256, 512, 768 and 1024 - these figures specify the size of the swap file on the hard disk in megabytes.

If the value is set to 0, any swap file created previously is deleted. The value is set to 0 by default - if there are no problems during compilation, continue to use this setting.

The selected settings are saved using the **Save RTPC Configuration** button – **Discard Changes** discards changed settings.

- Clock Synchronization
Click **Synchronize** to compare and synchronize real-time PCclock and host clock.
- System Configuration
This part is used to configure the PC hardware. The options are automatically set correctly the first time ETAS RTPC is booted. Modification is therefore not necessary.
If an Ethernet interface is installed at a later date, the "RTPC_SYS_ETHn" (n = 1 ... 4) option has to be changed from "None" to "present".
Save your selection by clicking the **System Configuration** button.
- System Boot Settings
ETAS RTPC saves two complete software versions on separate partitions of the real-time PC. This enables a risk-free return to a familiar software version.
You can select which software version is to be used when the system is next booted in the selection box. This selection is saved using the **Set Boot Version** button.
- Reboot System
Reboot System restarts ETAS RTPC with the selected software version.

4.2.3.5 Installation/Update

- Apply Update
The current ETAS RTPC version can be updated with software updates via the web interface. These software updates can be obtained directly from ETAS.
 - Under "Upload Update File" click the **Browse...** button to select the file with the update (extension *.tgz).
 - Select **Upload** to transfer the file to the real-time PC.
The uploaded file is now selected under "Select Update to Apply".
 - If several updates have been uploaded, select the relevant update to be activated.
 - Under "Select System to Update" select the software version to be updated. The software version currently active is shown ("Active").
 - Click the **Apply Update** button.
Updates carried out to date are listed under "Applied Updates".
Under "Delete Installed Update Files" you can delete files no longer required.
 - Activate the files to be deleted and click **Delete**.
- Install New Version
The "Install New Version" function is used to execute a complete new installation. ETAS RTPC saves all software versions installed to date from CD on the hard disk of the real-time PC. These versions can be written back to any partition.
There are two ways of making installation files available on the hard disk of the real-time PC:

- Upload Image file

An image (on your local system or on a network resource) is transferred to the real-time PC.

- Check CD/DVD-ROM disk

This is where the optical drive of the user PC is checked for the presence of a ETAS RTPC installation disk. If one is found, the image file is transferred to the real-time PC.

The images provided in this way can now be selected using "Select Software Image" (see below).

- Select the partition installation should take place in under "Install Partition".

Every partition except the active one can be overwritten.

- Select which version is to be installed under "Select Software Image".
- Use the "Use the existing configuration...." option to specify whether the existing configuration of ETAS RTPC is to be used for this version or whether standard values are to be used.
- If you leave the "Skip implicit updates" option disabled, all available updates for the relevant software version are also automatically installed.
- Click **Install**.

The version to be booted is selected in "Configuration" (see "**System Boot Settings**" on page 54).

Images on the hard disk of the real-time PC which are no longer needed can be marked under "Delete Installed Software Images" and then deleted using **Delete**.

- Clone Installed Version

You can clone versions of ETAS RTPC that have already been installed by generating an installable file (*.tgz) and downloading this file to your operating PC. For a detailed description of this procedure, refer to the section "Cloning Installed Versions" on page 17.

4.2.3.6 System Info

The "System Info" function shows various information on the status of ETAS RTPC. If an error occurs, this page can be saved and used for analysis purposes.

- Log File messages

The messages recorded are shown in this window. It is also possible

- to have the complete file displayed (**View complete log file**),
- to generate a ZIP archive which contains the log file as well as other files with system information (**Generate ZIP file report**) or
- to generate an encrypted ZIP archive (password: etas) (**Encrypted ZIP**).



TIP

As the ETAS firewall only accepts encrypted ZIP files, you must select this option if you want to send a report archive to ETAS by e-mail.

- Realtime Process Report

This provides detailed information on running real-time processes.

- System Information

System information of the real-time PC is shown here.

- Cyclictest - Test Program

This is where the test program "cyclictest" can be launched - the results are also shown on this page. For a detailed description, please refer to "Testing Hardware Performance with "cyclictest"" on page 37.

- CPU Performance Monitor

Uses the "CPU Performance Monitor" registers to analyze CPU performance.

- LABCAR I/O Hardware Information

This provides detailed information on the connected I/O hardware (VME-bus and PCI/PCIe boards).

- Applied Updates

Contains information on installed updates.

- Debug/Disassemble

If the "RTPC_COMPILE_LINK_DEBUG" parameter has been set (see Page 53), the model can be examined here for errors.

4.2.3.7 License Management

This is where the licenses for ETAS RTPC and the FlexRay boards are managed.

- ETAS RTPC Runtime Licenses
 - License Information

Shows the MAC address to which the license is linked, the names of the active licenses and the license file itself.
 - License File Upload

This is where you can upload a license file on the host PC to the real-time PC (**Upload**).

ETAS RTPC (V6.3.7)
Run your models in real time on a PC

Main Page >> License Management >> License Handling

License Information

MAC Address
88:9c:28:1d:78:ee

Active Licenses

- LABCAR
LCS_RTPC_RT: Success

License Files

☒ LCS_RTPC_RT.2014-12.lic (Fri Jul 11 18:05:50 2014)

```
INCREMENT LCS_RTPC_RT ETAS 6.0 31-dec-2014 uncounted \
  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140 1141 1142 1143 1144 1145 1146 1147 1148 1149 1150 1151 1152 1153 1154 1155 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 1170 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194 1195 1196 1197 1198 1199 1200 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215 1216 1217 1218 1219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 1243 1244 1245 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260 1261 1262 1263 1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 1274 1275 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290 1291 1292 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316 1317 1318 1319 1320 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335 1336 1337 1338 1339 1340 1341 1342 1343 1344 1345 1346 1347 1348 1349 1350 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 1365 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 1414 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455 1456 1457 1458 1459 1460 1461 1462 1463 1464 1465 1466 1467 1468 1469 1470 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485 1486 1487 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511 1512 1513 1514 1515 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530 1531 1532 1533 1534 1535 1536 1537 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589 1590 1591 1592 1593 1594 1595 1596 1597 1598 1599 1600 1601 1602 1603 1604 1605 1606 1607 1608 1609 1610 1611 1612 1613 1614 1615 1616 1617 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 1628 1629 1630 1631 1632 1633 1634 1635 1636 1637 1638 1639 1640 1641 1642 1643 1644 1645 1646 1647 1648 1649 1650 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 1674 1675 1676 1677 1678 1679 1680 1681 1682 1683 1684 1685 1686 1687 1688 1689 1690 1691 1692 1693 1694 1695 1696 1697 1698 1699 1700 1701 1702 1703 1704 1705 1706 1707 1708 1709 1710 1711 1712 1713 1714 1715 1716 1717 1718 1719 1720 1721 1722 1723 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734 1735 1736 1737 1738 1739 1740 1741 1742 1743 1744 1745 1746 1747 1748 1749 1750 1751 1752 1753 1754 1755 1756 1757 1758 1759 1760 1761 1762 1763 1764 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 1775 1776 1777 1778 1779 1780 1781 1782 1783 1784 1785 1786 1787 1788 1789 1790 1791 1792 1793 1794 1795 1796 1797 1798 1799 1800 1801 1802 1803 1804 1805 1806 1807 1808 1809 1810 1811 1812 1813 1814 1815 1816 1817 1818 1819 1820 1821 1822 1823 1824 1825 1826 1827 1828 1829 1830 1831 1832 1833 1834 1835 1836 1837 1838 1839 1840 1841 1842 1843 1844 1845 1846 1847 1848 1849 1850 1851 1852 1853 1854 1855 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115 2116 2117 2118 2119 2120 2121 2122 2123 2124 2125 2126 2127 2128 2129 2130 2131 2132 2133 2134 2135 2136 2137 2138 2139 2140 2141 2142 2143 2144 2145 2146 2147 2148 2149 2150 2151 2152 2153 2154 2155 2156 2157 2158 2159 2160 2161 2162 2163 2164 2165 2166 2167 2168 2169 2170 2171 2172 2173 2174 2175 2176 2177 2178 2179 2180 2181 2182 2183 2184 2185 2186 2187 2188 2189 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2201 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 25
```

Main Page >> License Management >> Elektrobit License Manager

Device List

```
INFO: =====
INFO: Searching for known PCI devices...
INFO:      Index 0: EB 5100 with Serial Number 249
```

Device License (Serial Number '249')

```
Target ready
Available licenses:
EB_tresos_Busmirror_FlexRay (05/2011), 2009.a:OK, 2009.a.sr1:OK, 2010.a:OK, 3.3:OK, :
```

Write License File to Elektrobit Device

Serial Number 249

License File Browse...

Write

Read License File from Elektrobit Device

Serial Number 249

Read

Update

- To transfer a license file from the user PC to a EB_5X00 board, select the board serial number.
- Select the file with **Browse** and click **Write**.
- To transfer a license file from a board to the user PC, select the board serial number and click **Read**.

4.2.3.8 Documentation

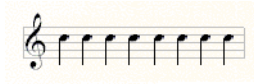
The "Documentation" link leads you to a directory containing the following documents:

- The manual on ETAS RTPC V6.3.7
- The current release notes
- A list of PCs approved by ETAS
- Two documents with information on the BIOS settings of the PCs from ETAS

5 Troubleshooting

This chapter presents the description of some typical problems with possible solutions.

A When ETAS RTPC is started up, the sequence of error notes is played



- Cause:
The Simulation Controller could not be started.
- Remedy:
See “**Simulation Controller mode cannot be activated**” on page 61

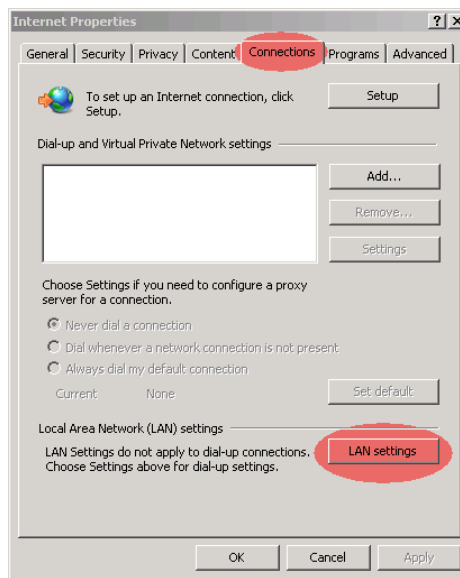
B The ETAS RTPC Web interface cannot be accessed

- Cause:
Ethernet connection does not exist, is defective or the settings are incorrect.
- Remedy 1:
Execute the command `ping 192.168.40.14`. If you receive an answer from ETAS RTPC, check the settings of your Web browser (e.g. proxy settings).
- Remedy 2:
If you do not receive an answer from ETAS RTPC, check the Ethernet cable. Make sure that you are using a crossover cable.
- Remedy 3:
If the cable is OK, check the configuration of the Ethernet interface (IP address: 192.168.40.240, net mask: 255.255.255.0)
- Remedy 4:
In your Internet configuration, a proxy server is configured for the LAN. To be able to access the ETAS RTPC web interface, the IP address of the real-time PC (192.168.40.14) must be included in the list of exceptions that should not use the proxy server.

To do so, proceed as follows:

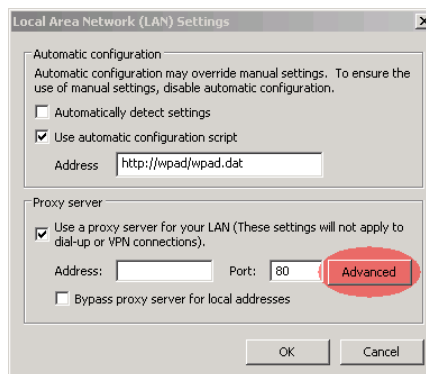
1. In the Windows Start menu, select **Settings → Control Panel → Internet Options**.

The "Internet Properties" window opens.



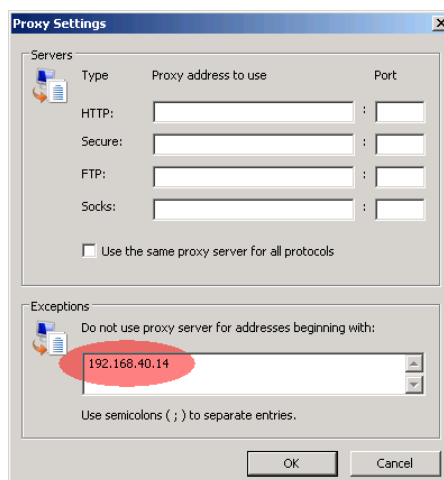
2. Select the "Connections" tab and click **LAN settings**.

The "Local Area Network (LAN) Settings" window opens.



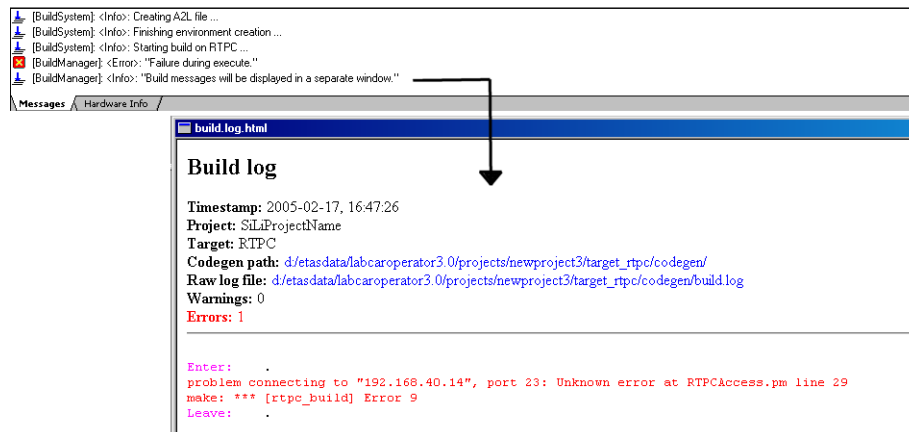
3. Click **Advanced** in the "Proxy server" field.

The "Proxy Settings" window opens.



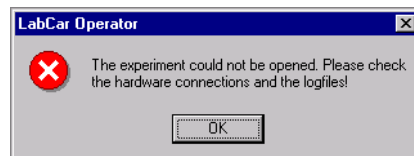
4. Enter the IP address of the real-time PC in the "Exceptions" field (192.168.40.14).
5. Close one window after the other with **OK**.

C Compiling models does not work



- Cause:
The real-time PC is not connected to the user PC or is switched off.
- Remedy:
Connect the real-time PC or switch it on. Check communication by calling the Web interface (<http://192.168.40.14> or <http://rtpc>).

D The project download ("Open Experiment") in LABCAR-OPERATOR does not work



- Cause 1:
Simulation Controller has not been started
- Remedy 1:
Use the Web interface to check whether the Simulation Controller has been started. Activate Simulation Controller mode if necessary.
- Cause 2:
ES1130 firmware is not compatible with ETAS RTPC.
- Remedy 2:
Check whether the firmware which is compatible with ETAS RTPC has been programmed on the ES1130. See the configuration page in the web interface for more details.

E Simulation Controller mode cannot be activated

- Cause 1:
The ES1130 has not been connected or is switched off. (HiL mode).
- Remedy 1:
Check whether the ES1130 has been connected and is switched on. Call the "ES1130 Detection" function under "Configuration Support". The window below shows the result of the hardware scan. The following line should be visible:

Ethernet Controller [eth1]: Connected to Target ES1130

- Cause 2:
The settings for RTPC_USAGE_ETH1 are not correct
- Remedy 2:
Check the settings for RTPC_USAGE_ETH1 in the configuration dialog and correct these (see “Applications” on page 44).
- Cause 3:
There is an interrupt conflict in the Ethernet interface.
- Remedy 3:
If an interrupt conflict has been registered on the configuration page of the web interface, insert the board into another slot or deactivate unnecessary peripheral devices in the PC BIOS (e.g. USB or sound support).

6 Glossary

This chapter explains technical terms used in the manual. The terms are listed in alphabetic order.

API

Application Programmer's Interface

BIOS

Basic Input Output System - the firmware of the PC

CAN

Controller Area Network (ISO 15765), a multi-master broadcast serial bus standard for connecting ECUs

CPU

Central Processing Unit

DSUB

A common hardware connector type

DVD

Digital Versatile Disc

DVE

Driver-Vehicle-Environment

EVE

ETAS Virtual ECU

Eth0

Designation of the Ethernet port for connecting the host PC

Eth1

Designation of the Ethernet port for connecting hardware like ES1130 or ES4440

GB

Gigabyte

GNU

Open source software community

HiL

Hardware-in-the-Loop

HTTP

Hypertext Transfer Protocol

ID

Identification key

IDE

Integrated Drive Electronics

IP

Internet Protocol

INCA

Measurement, Calibration, and Diagnosis tool available from ETAS

ISO

International Organization for Standardization

| | |
|-------------|---|
| LAN | Local Area Network |
| LPT | Parallel port (IEEE-1284) |
| MB | Megabyte |
| NTP | Network Time Protocol |
| PC | Personal computer (desktop or laptop) |
| PCIe | PCI Express (Peripheral Component Interconnect Express) |
| PS/2 | Serial input for PCs |
| RAM | Random Access Memory |
| RTPC | Real Time PC |
| UDP | User Datagram Protocol – a network protocol |
| URL | Uniform Resource Locator |
| VME | A common bus standard |
| VT-d | Intel's Virtualization Technology for Directed I/O |
| XP | A Microsoft Windows release |
| ZIP | Compressed archive file format |

7 ETAS Contact Addresses

ETAS HQ

ETAS GmbH

Borsigstraße 24

70469 Stuttgart

Germany

Phone: +49 711 3423-0

Fax: +49 711 3423-2106

WWW: www.etas.com

ETAS Subsidiaries and Technical Support

For details of your local sales office as well as your local technical support team and product hotlines, take a look at the ETAS website:

ETAS subsidiaries WWW: www.etas.com/en/contact.php

ETAS technical support WWW: www.etas.com/en/hotlines.php

Index

| | |
|--|--------|
| A | |
| Automatic configuration | 49 |
| B | |
| Benchmark indicator | 37 |
| BIOS | |
| Settings | 11 |
| C | |
| CAN Board | 20 |
| CAN Trace | 26 |
| CAN/CANFD-Abstraction | 39 |
| Clock Synchronization | 19, 54 |
| Clone | |
| install versions | 55 |
| Configuration | 44 |
| CPU Core Load | 36 |
| CPU Performance Monitor | 56 |
| cyclictest | 37 |
| D | |
| Data Protection Declaration | 6 |
| Delivery scope | 9 |
| E | |
| ETAS Contact Addresses | 65 |
| Ethernet Order | 51 |
| F | |
| FlexRay Board | 20 |
| H | |
| Hardware requirements | 10 |
| Hardware-in-the-Loop (HiL) | 44 |
| Hyper-threading | 11 |
| I | |
| Installation | 12 |
| L | |
| LABCAR I/O hardware | |
| information | 56 |
| License Management | 57 |
| Licensing | 16 |
| LIN-Abstraction | 39 |
| M | |
| Massive I/O | 45 |
| Model-in-the-Loop (MiL) | 44 |
| O | |
| Operating mode | |
| Ethernet Bridge | 22 |
| Idle/Configuration | 22 |
| Power Down | 22 |
| Simulation Controller | 21 |
| R | |
| Real-Time Plugins | 32 |
| Realtime Process Report | 56 |
| RTOS_TASK_STACKSIZE | 53 |
| RTOS_TRACE_BUFFERSIZE | 53 |
| RTPC_COMPILE_LINK_DEBUG | 53 |
| RTPC_COMPILE_OPTIMIZATION | 53 |
| RTPC_COMPILE_SWAPSPACE_MB | 53 |
| RTPC_INVALID_CPU_CORE_HANDLING | 52 |
| RTPC_LOG_LEVEL | 51 |
| RTPC_NAME | 51 |
| RTPC_POWER_UP_MODE | 51 |
| RTPC_TASK_MONITOR_0/1 | 52 |
| RTPC_TASK_TIMING_STATISTIC | 52 |
| Runtime Trace | 24 |
| S | |
| Simulation | |
| step-by-step execution | 30 |
| Step-by-step execution of the simulation | 30 |
| System Boot Settings | 54 |
| System configuration | |
| HiL test system | 15 |
| System requirements | 9 |
| T | |
| Troubleshooting | 59 |
| V | |
| VMEbus reset | 22 |
| W | |
| Web interface | 46 |
| functions | 48 |
| navigation | 47 |

