

---

# RTA-OSEK

Getting Started Guide



## Contact Details

---

### ETAS Group

[www.etasgroup.com](http://www.etasgroup.com)

### Germany

ETAS GmbH  
Borsigstraße 14  
70469 Stuttgart

Tel.: +49 (711) 8 96 61-102  
Fax: +49 (711) 8 96 61-106

[www.etas.de](http://www.etas.de)

### Japan

ETAS K.K.  
Queen's Tower C-17F,  
2-3-5, Minatomirai, Nishi-ku,  
Yokohama, Kanagawa  
220-6217 Japan

Tel.: +81 (45) 222-0900  
Fax: +81 (45) 222-0956

[www.etas.co.jp](http://www.etas.co.jp)

### Korea

ETAS Korea Co., Ltd.  
4F, 705 Bldg. 70-5  
Yangjae-dong, Seocho-gu  
Seoul 137-899, Korea

Tel.: +82 (2) 57 47-016  
Fax: +82 (2) 57 47-120

[www.etas.co.kr](http://www.etas.co.kr)

### USA

ETAS Inc.  
3021 Miller Road  
Ann Arbor, MI 48103

Tel.: +1 (888) ETAS INC  
Fax: +1 (734) 997-94 49

[www.etasinc.com](http://www.etasinc.com)

### France

ETAS S.A.S.  
1, place des États-Unis  
SILIC 307  
94588 Rungis Cedex

Tel.: +33 (1) 56 70 00 50  
Fax: +33 (1) 56 70 00 51

[www.etas.fr](http://www.etas.fr)

### Great Britain

ETAS UK Ltd.  
Studio 3, Waterside Court  
Third Avenue, Centrum 100  
Burton-upon-Trent  
Staffordshire DE14 2WQ

Tel.: +44 (0) 1283 - 54 65 12  
Fax: +44 (0) 1283 - 54 87 67

[www.etas-uk.net](http://www.etas-uk.net)

**People's Republic of China**

2404 Bank of China Tower  
200 Yincheng Road Central  
Shanghai 200120

Tel.: +86 21 5037 2220

Fax: +86 21 5037 2221

[www.etas.cn](http://www.etas.cn)

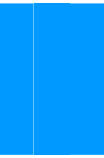
**LiveDevices**

LiveDevices Ltd.  
Atlas House  
Link Business Park  
Osbalwick Link Road  
Osbalwick  
York, YO10 3JB

Tel.: +44 (0) 19 04 56 25 80

Fax: +44 (0) 19 04 56 25 81

[www.livedevices.com](http://www.livedevices.com)



## Copyright Notice

---

© 2002 - 2007 LiveDevices Ltd. All rights reserved.

Version: RTA-OSEK v5.0.2

No part of this document may be reproduced without the prior written consent of LiveDevices Ltd. The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such a license.

## Disclaimer

---

The information in this document is subject to change without notice and does not represent a commitment on any part of LiveDevices. While the information contained herein is assumed to be accurate, LiveDevices assumes no responsibility for any errors or omissions.

In no event shall LiveDevices, its employees, its contractors or the authors of this document be liable for special, direct, indirect, or consequential damage, losses, costs, charges, claims, demands, claim for lost profits, fees or expenses of any nature or kind.

## Trademarks

---

RTA-OSEK and LiveDevices are trademarks of LiveDevices Ltd.

Windows and MS-DOS are trademarks of Microsoft Corp.

Adobe and Acrobat are trademarks of Adobe Systems Incorporated.

OSEK/VDX is a trademark of Siemens AG.

All other product names are trademarks or registered trademarks of their respective owners.



---

## Contents

1	About this Guide .....	1-1
1.1	Who Should Read this Guide? .....	1-1
1.2	Conventions .....	1-1
2	RTA-OSEK Documentation .....	2-1
3	System Requirements .....	3-1
3.1	Host Hardware Requirements .....	3-1
3.2	Software Requirements .....	3-1
4	Installation .....	4-1
4.1	Licensing .....	4-1
4.1.1	Installing the License Keys .....	4-1
4.1.2	Merging License Files .....	4-1
4.1.3	Moving the License File after Installation .....	4-1
4.2	Installing RTA-OSEK .....	4-2
4.3	RTA Folder Structure .....	4-3
4.4	Troubleshooting .....	4-4
4.4.1	Installing a Target .....	4-4

	4.4.2 Using RTA-OSEK.....	4-4
5	Configuration .....	5-1
	5.1 Updating toolinit.bat.....	5-1
	5.2 Customizing RTA-OSEK.....	5-1
	5.2.1 Application Settings.....	5-2
	5.2.2 Auxiliary Files Settings.....	5-3
	5.2.3 Global Settings .....	5-4
	5.2.4 Licensing Settings .....	5-6
6	Building the Example Application .....	6-1
	6.1 What Does the Example Application Do?.....	6-1
	6.2 Target Specific Configuration .....	6-2
	6.2.1 target.h.....	6-2
	6.2.2 target.c .....	6-2
	6.3 Building the Example Application .....	6-3
	6.4 Verifying Program Execution .....	6-4
	6.5 Troubleshooting .....	6-5
7	Support.....	7-1
	7.1 Getting Help.....	7-1
	7.2 Email.....	7-1
	7.3 Telephone .....	7-1
	7.4 Fax.....	7-1





# 1 About this Guide

---

In this guide you will find out how to install and configure RTA-OSEK as well as learning about the other manuals that you can read. You will also find instructions for building the example application that is supplied with your distribution.

## 1.1 Who Should Read this Guide?

---

It is assumed that you are a developer who wishes to install RTA-OSEK. You should read this guide before you install RTA-OSEK and before you read any other manuals.

## 1.2 Conventions

---

**Important:** Notes that appear like this contain important information that you need to be aware of. Make sure that you read them carefully and that you follow any instructions that you are given.

**Portability:** Notes that appear like this describe things that you will need to know if you want to write code that will work on any processor running RTA-OSEK Component.

The following terms are used in this guide:

RTA-OSEK	refers to the complete Real-Time Operating System product including the tools that run on the host PC, the target processor components and the documentation.
Offline tools	refers to the configuration, analysis and build tools that are run on the host PC. These include the RTA-OSEK graphical user interface (GUI) that provides a wrapper around the other offline tools.
RTA-OSEK GUI	refers to the RTA-OSEK graphical user interface (GUI) that provides a wrapper around the other offline tools.
RTA-OSEK Component	refers to the RTA-OSEK Real-Time Operating System kernel that runs on the target processor. Any references to <i>the kernel</i> in this guide refer to RTA-OSEK Component.

In this guide you'll see that program code, header file names, C type names, C functions and RTA-OSEK Component API call names all appear in the

`courier` typeface. When the name of an object is made available to the programmer the name also appears in the `courier` typeface, so, for example, a task named `Task1` appears as a task handle called `Task1`.

## 2 RTA-OSEK Documentation

---

Your RTA-OSEK distribution includes the following manuals:

- *RTA-OSEK Getting Started Guide*  
The manual that you are reading now.
- *RTA-OSEK User Guide*  
This guide describes the underlying principles of RTA-OSEK. You will learn about using the RTA-OSEK graphical user interface (GUI) and you'll find out how to configure, build and analyze applications. When you have read this *Getting Started Guide* you should read the *RTA-OSEK User Guide* next.
- *RTA-OSEK Reference Guide*  
This guide provides a complete reference to the API and programming conventions for RTA-OSEK.
- *RTA-OSEK ORTI User Guide*  
ORTI is the OSEK Run Time Interface. This guide will show you how ORTI can be used for debugging with a compatible debugger.
- *RTA-OSEK Binding Manual*  
Each target that is supported by RTA-OSEK is supplied with a Binding Manual. Binding Manuals tell you specific information about the interaction between RTA-OSEK and your target hardware. They include valid compiler options, register settings for your target hardware and OSEK conformance information.



## 3 System Requirements

---

### 3.1 Host Hardware Requirements

---

You should make sure that you are using at least the following hardware before installing and using RTA-OSEK on a host PC:

- Pentium or higher IBM compatible PC.
- 128Mb RAM.
- 50Mb hard disk space.
- CD-ROM or DVD drive.
- Ethernet card.

The Ethernet card MAC address is used to node lock RTA-OSEK to a nominated host PC, see Chapter 4.1 for further licensing details.

### 3.2 Software Requirements

---

Make sure that your host PC has one of the following installed:

- Windows 98, NT 4.0, ME, 2000 or XP.



## 4 Installation

---

### 4.1 Licensing

---

Each feature of RTA-OSEK is protected by a license key. Licenses are generated and administered using the FLEXlm licensing tools. You must have a valid license key for each RTA-OSEK feature that you want to use. If you require a license, please contact LiveDevices.

Normally, license keys for RTA-OSEK are locked to the MAC address of the Ethernet controller in your host PC. Licenses can, however, be generated and locked to other items of host PC. You should contact LiveDevices to find out more and to discuss your licensing needs.

#### 4.1.1 Installing the License Keys

---

The license keys are supplied in an ASCII text file, which will be sent to you on completion of a valid license agreement.

The file will be named using the format:

```
<License Identifier>-<MAC Address>-<Expiry Date>.lic
```

So, for example, you may have 101-13-000039351C81-12-Jul-2005.lic

You must copy the license file onto your host PC. We recommend that you place the license file in the root folder of your hard disk drive and rename the file `rta.lic`.

During the installation process, RTA-OSEK will ask you for the location of this file. The keys in the license file will be verified each time that RTA-OSEK offline tools are used.

If you are using a **floating** license to support multiple hosts you will be supplied with the appropriate tools for installing your license server.

Further information about FLEX licensing can be found on the RTA-OSEK installation CD in the folder `\Documents\FLEXlm End User Documents`.

#### 4.1.2 Merging License Files

---

If other products installed on your host PC use the same licensing technology, you can merge the license files. Make sure that you copy all the `FEATURE` lines from the RTA-OSEK license file into your existing license file.

#### 4.1.3 Moving the License File after Installation

---

If you wish to move your license file, whether you have merged it or not, you must make sure that RTA-OSEK knows where the file can be found. Use the Licensing settings dialog as explained in Chapter 5.2.4 to inform RTA-OSEK of the new license file location.

## 4.2 Installing RTA-OSEK

---

RTA-OSEK is installed from two CDs. One of these contains the offline tools, the other the target-specific parts of the operating system for your platform.

You must install the Tools CD first – this will create the necessary directory structures, install the offline tools and install the license that will enable the target-specific parts of the system to be used.

- Insert the Tools CD into your CD-ROM or DVD drive and wait for the installation program to start automatically.

If the installation program does not automatically run, you must start the `setup.exe` program from the CD.

- During the install, you will be given the following options:
  - **Full Installation:** All software and documentation will be installed
  - **Install Tools Only:** Only the RTA-OSEK offline tools will be installed.

When you have selected the option that you want, click the **Install** button.

- Read the license agreement before proceeding with installation. You can continue with the installation if you accept the terms of the license. If you decline the terms, the installation will terminate.
- Follow the on-screen instructions to install the components that you selected.
- When asked to select the RTA-OSEK license file, specify the name and location of your installed license file. If you do not have a valid license file, please contact LiveDevices before continuing.
- If you are installing the RTA-OSEK offline tools you will be asked to select a Typical, Minimal or Custom setup type. When you are asked, select **Typical**.

**Important:** If you are installing the RTA-OSEK tools and multiple targets, do not restart your machine until all targets and tools have been installed.

- If you are installing the tools, you can set the RTA-OSEK GUI as the default `.oil` file editor. Set this as the default, if required.

You have now installed the offline tools.

For each target that you have purchased a license for:

- Insert the Target CD into your CD-ROM or DVD drive and wait for the installation program to start automatically. If the installation program does not automatically run, you must start the `setup.exe` program from the CD
- Restart your computer, if you are asked.

When installation of the tools and targets is complete, you can run the RTA-OSEK GUI by launching the **RTA-OSEK** item from your Windows Start menu.



## 4.3 RTA Folder Structure

After installing RTA-OSEK, a number of new folders will be created. The location of these folders depends on where you decided to install the files. The default location is `c:\rta`.

The following table shows the folders that are created. Where you see *instdir* in the table, replace this with the installation folder that you selected (for example, `c:\rta`). Where you see *target* replace this with the name of the specific target platform that you have installed (for instance, *Task167* or *COS12\_16*).

Folder	Contents
<i>instdir</i>	This is the root RTA installation folder. This folder contains <code>rtainit.bat</code> .
<i>instdir</i> \bin	This folder contains the executables for the host PC parts of RTA-OSEK. It also contains the DLLs used by RTA-OSEK for each installed target.
<i>instdir</i> \docs	This folder contains online copies of the RTA-OSEK manuals. You can view the manuals using the Adobe Acrobat Reader.
<i>instdir</i> \packages	This folder contains package definitions for additional 3 <sup>rd</sup> party software where supported. RTA-OSEK installs a package definition for RTA-COM here.
<i>instdir</i> \target	This is the root folder for the specified target. This folder contains the file <code>toolinit.bat</code> . You may need to modify this file to work with your toolchain installation. See Chapter 5 for more information.
<i>instdir</i> \target\drivers	This folder contains example drivers for an RTA-TRACE datalink.
<i>instdir</i> \target\Example	This folder contains all of the files you will need for the example application, which is explained in Chapter 6. The <i>RTA-OSEK Binding Manual</i> for your target contains further information on building and running this application.
<i>instdir</i> \target\Inc	All the C Header files for RTA-OSEK Component can be found here.

Folder	Contents
<code>\instdir\target\Lib</code>	<p>This is the installation folder for the precompiled RTA-OSEK Component libraries.</p> <p><code>rtk_*</code> are the RTA-OSEK Component libraries you will use in your own applications.</p> <p><code>RT*</code> are the RTA-TRACE datalink libraries that you can use when tracing your applications.</p>
<code>\instdir\target\Packages</code>	Build file configuration and stack size/execution time template files for RTA-COM package.

## 4.4 Troubleshooting

---

### 4.4.1 Installing a Target

---

When you are installing a target, make sure that:

- Your license file contains a key for all the features that you want to use.
- You have selected the correct license file during the installation of RTA-OSEK.

If this does not resolve your issues, please contact the LiveDevices for assistance.

### 4.4.2 Using RTA-OSEK

---

RTA-OSEK will report an error if you try to use a feature for which a correct license key cannot be found. If you think that you should have a license for a feature but RTA-OSEK appears not to work then should follow the trouble shooting steps below before contacting LiveDevices:

- Check that your license can be seen by RTA-OSEK

By default, RTA-OSEK looks for the licence file in the location that you specified during the installation process. However, if you have installed other FlexLM licensed products then you may have the environment variable `LM_LICENCE_FILE` set. `LM_LICENCE_FILE` tells FlexLM where to look for licenses. If this variable is set then it overrides the default search path.

To solve this problem you will need to add the path to your RTA-OSEK license file to the `LM_LICENCE_FILE`. You can do this using Windows Control Panel. Open System > Advanced > Environment Variables and look in the User and/or System Variables for `LM_LICENCE_FILE`

- Check that your license keys are valid.  
You can check that the expiry date for your licensed features has not elapsed by opening your license file in a text editor. If you have a current maintenance contract, LiveDevices will automatically provide you with a new license file that extends the license period prior to the license expiring.
  - Check that the Ethernet MAC address matches the one specified in the license file.
  - Check that the MAC address in the license file matches the MAC address of the Ethernet controller on your host PC.  
You can find out the MAC address of your PC by running the Microsoft program `ipconfig /all` at a Command Prompt. The "Physical Address" reported should be identical to the MAC address in your licence file.
  - Check that your Ethernet controller is still enabled when your PC is not physically connected to a network  
If you use a laptop and RTA-OSEK stops working when you disconnect from the network then you should check your hardware settings to ensure that your Ethernet controller is not turned off to save power when a network connection is not present. You can do this using Windows Control Panel. Open System > Hardware > Device manager then select your Network Adapter. Right click to open "Properties" and check that the Ethernet controller is not configured for power saving in "Advanced" and/or "Power management" settings.
- If you have not resolved your issues, after confirming these points above, please contact LiveDevices. You must provide the contents and location of your license file and your Ethernet MAC address.



## 5 Configuration

---

Before you create an application using RTA-OSEK, there are a number of settings that should be modified to suit your development environment. Chapter 5.1 describes how to change the necessary environment variables to suit your compiler toolchain. Optional configuration settings are described in Chapter 5.2.

### 5.1 Updating toolinit.bat

---

You must update the `toolinit.bat` file for each target that you have installed. You will find each file in the target specific RTA root folder (have a look at Chapter 4.3 to find out where files are installed).

The `toolinit.bat` file defines the environment variables used by RTA-OSEK during the build process. The variables in this batch file must be set to the location that your toolchain is installed.

The following environment variables are defined:

Variable	Description
CBASE	Base directory for the toolchain.
CC	Location of the compiler.
AS	Location of the assembler.
LNK	Location of the linker.
AR	Location of the librarian / archiver.
CBASE_INC	Directory containing the C run-time include files.

Other macros may be defined. You should check `toolinit.bat` for further information.

When you build applications, the RTA-OSEK GUI will use the tools that you specify in `toolinit.bat`. Building from within the RTA-OSEK GUI will only work if these tools are configured correctly. Compatibility issues with the compiler toolchain are the most probable causes of build problems.

Make sure that you are using the correct compiler version. The *RTA-OSEK Binding Manual* for your target will tell you which version to use. The compiler options should only be changed after the Binding Manual has been consulted. If problems persist, please contact LiveDevices.

### 5.2 Customizing RTA-OSEK

---

You can change a number of the default RTA-OSEK settings using the information in the following descriptions. All of these settings can be accessed by selecting **Options...** from the **File** menu.

## 5.2.1 Application Settings

The **Application Settings** tab can be used to set application specific settings.

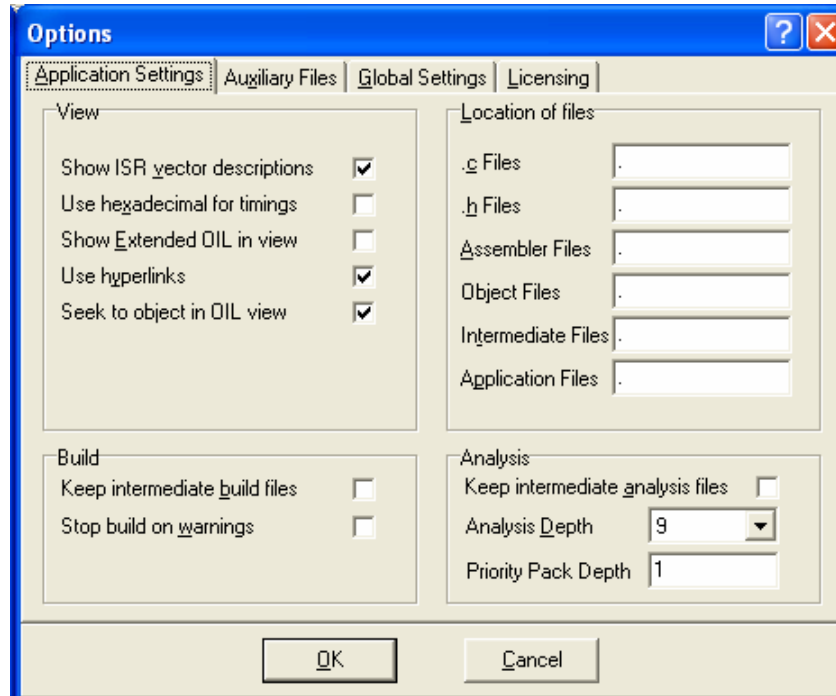


Figure 5:1 – Options for application settings

Each application can change the following settings, which are preserved in structured comments in the application's OIL file:

- View settings

**Show ISR vector descriptions.** Selecting this option will display descriptive names for ISR vectors when these are available. Otherwise the hexadecimal address of the vectors will be used.

**Use hexadecimal for timings.** All timing values will be displayed in hexadecimal when this option is selected. Otherwise decimal values will be used.

**Show Extended OIL in view.** Certain information about an application is captured by RTA-OSEK in smart comments embedded in the OIL file. Selecting this option will enable these comments to be seen in the OIL File view. Otherwise only the standard OIL is displayed.

**Use hyperlinks.** When this option is selected, each object configured in RTA-OSEK will be displayed as a hyperlink in all information screens. Clicking on the hyperlink allows you to access that object's configuration menu. Otherwise object names will be displayed in plain text.

**Seek to object in OIL view.** When this option is selected RTA-OSEK will automatically display the portion of the OIL file that is relevant to the object being configured. Otherwise the portion of the OIL file that is displayed will remain at the last selected position.

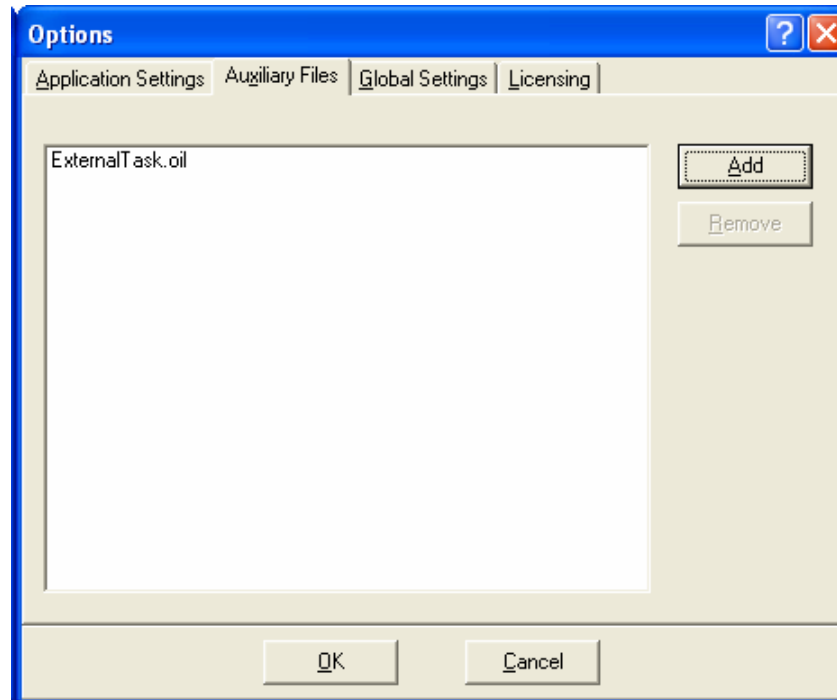
- Build settings
  - Keep intermediate build files.** When this option is selected RTA-OSEK will preserve any intermediate files generated when building the application. Otherwise these files will be automatically deleted.
  - Stop build on warnings.** RTA-OSEK will halt the application build on any warning messages when this option is selected. Otherwise only error messages will stop the build process.
- Location of files settings

If you wish RTA-OSEK to generate files into specific locations for your application then you can enter the paths to the file locations for C source files, header files, assembler files, object files, intermediate files and application files. These may be absolute paths, or paths relative to the locations of the application's OIL file.
- Analysis settings
  - Keep intermediate analysis files.** RTA-OSEK will preserve intermediate files used for analysis when this option is selected. Otherwise these files will be automatically deleted.
  - Analysis depth.** This controls the performance of the schedulability analysis. Values 1 and 9 are supported (2-8 are reserved for future extensions). Value 9 is recommended for most users. If you find that analysis is consuming excessive time then you should experiment with using Value 1.
  - Priority pack depth.** This setting controls how many tasks are considered simultaneously at the same priority level when performing timing analysis.

### 5.2.2 Auxiliary Files Settings

---

Auxiliary OIL files are used to store fragments of OIL, for example, the configuration for a particular module of a larger system. You can add auxiliary OIL files to an application by browsing to them using the Auxiliary Files settings dialog.



**Figure 5:2 – The Auxiliary Files dialog**

When adding an auxiliary file to an application, you can choose to save it by name only, as a relative path to the application OIL file, or as an absolute path.

### **5.2.3 Global Settings**

---

You can change how RTA-OSEK behaves for all applications using the Global Settings dialog.



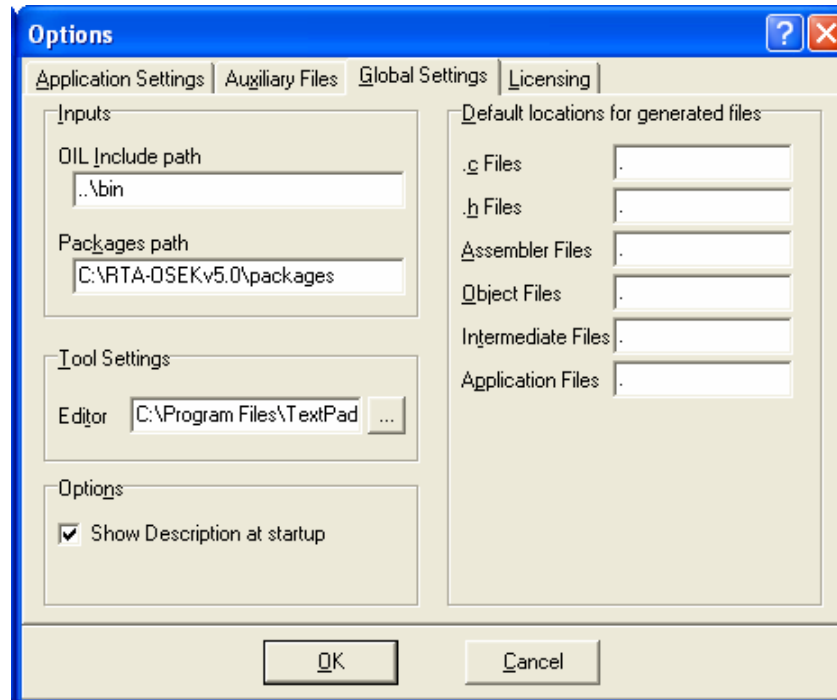


Figure 5:3 – The Global Settings dialog

You can customize the following RTA-OSEK settings:

- Inputs settings
  - OIL Include path.** This setting is used to specify the location for files that are incorporated into an OIL file using the `#include` mechanism.
  - Packages path.** This setting is used to specify the location of files describing packages used by RTA-OSEK.

Both of the above settings can contain multiple paths separated by ';' characters.
- Tool settings
  - Editor.** You can use this setting to ensure that RTA-OSEK will use your preferred text editor program when displaying files.
- Options settings
  - Show Description at startup.** When this option is selected, RTA-OSEK will display a description of the application, if one has been entered, when you open an OIL file. Otherwise no description is displayed.
- Default location for generated files.
 

You can specify the default location for the following types of files that are generated by RTA-OSEK: C source files, header files, assembler files, object files, intermediate files and application files. Each type of file can be generated to a different location if required.

## 5.2.4 Licensing Settings

RTA-OSEK is licensed using the FlexLM licensing tools, as described in Chapter 4.1. You can manage your license files using the Licensing settings dialog.

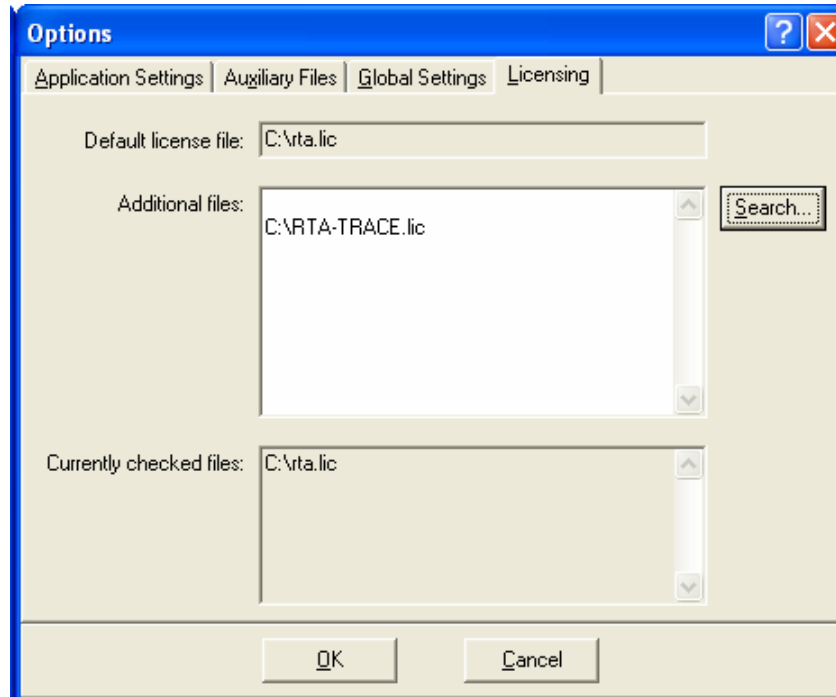


Figure 5:4 – Licensing settings dialog

Use the Search... button to select additional license files that RTA-OSEK should search for valid license keys. This can be useful if you have separate license keys for different RTA-OSEK target ports, for example.

## 6 Building the Example Application

The example application has been supplied so that you can verify that your installation has been successful. Once you have installed RTA-OSEK and configured the toolchain, you can then build and run this program. Building the example application checks that the RTA-OSEK tools, target and appropriate toolchain are configured correctly.

The example application can be found in the `\instdir\target\Example` folder for each target that you have installed (have a look at Chapter 4.3 to find out where files are installed).

**Important:** As well as following the instructions in this chapter, you may also need to configure the example application for your own target. Target specific details can be found in the application description notes in `example.oil`. You can view these notes by opening this file in the RTA-OSEK GUI.

### 6.1 What Does the Example Application Do?

The example application is a very simple program. It shows preemption between two tasks, A and B. Both tasks run for 2ms. Task A is the higher priority task and runs periodically every 50ms. Task B runs periodically every 25ms.

The periodic running of the tasks is achieved using two alarms, `stim1` and `stim2`, which are attached to a counter. The counter is ticked using a 1ms timer interrupt that is handled by the Interrupt Service Routine (ISR) `isr1`. Both alarms are autostarted. `stim1` is offset 1ms relative to `stim2`.

When the application runs, `stim2` will expire after 1ms and activate task B. Task B sets `IO_PIN1` high and runs for 1ms before being preempted by task A. This happens because task A is higher priority than task B.

Task A saves the state of `IO_PIN1` and then sets it low before setting `IO_PIN2` high. It then runs for 2ms before setting `IO_PIN2` low. Finally, it restores the state of `IO_PIN1` and terminates, allowing task B to continue from the point at which it was preempted. Task B then runs for its remaining 1ms.

You will find the following files in `\instdir\target\Example`. You may also find other files (such as files containing code to handle a reset).

File Name	Description
<code>example.oil</code>	This is the configuration file for the application that defines the tasks, interrupts, alarms and counters and also their relationships.
<code>A.c</code>	Source code for task A.
<code>B.c</code>	Source code for task B.

File Name	Description
<code>isr1.c</code>	Source code for the <code>isr1</code> .
<code>main.c</code>	This is the main program. It provides the idle task for the RTA-OSEK kernel.
<code>wait.h</code>	Header file for busy wait loop.
<code>wait.c</code>	Source code for busy wait loop.
<code>target.h</code>	Header file for target specific functions.
<code>target.c</code>	Source code for target specific functions.
<code>&lt;target&gt;def.h</code>	Definitions of target ports and control registers where <code>target</code> is the name of the <i>target</i> microcontroller.

Other files, such as linker scripts, may exist.

## 6.2 Target Specific Configuration

The example application is supplied ready to run on a reference hardware platform. This is usually an evaluation board for the target microcontroller.

When you open `example.oil` in the RTA-OSEK GUI, specific platform documentation can be found in the application description notes that are displayed automatically. If your target hardware differs from the reference platform, you will need to modify the target specific aspects defined in `target.h` and `target.c`.

### 6.2.1 `target.h`

The example application uses two I/O pins configured as outputs to display task activity. You can find out more about the port and pins used on the reference hardware in the application description notes in `example.oil`. The `target.h` file defines macros for configuring and initializing the I/O port and for setting and reading the state of port pins. The pins used on the port are defined by the macros, `IO_PIN1` and `IO_PIN2`.

The macros use logical names for the target hardware ports that are defined in `<target>def.h`. If you change the port or the pins you will need to modify these macros and may have to modify `<target>def.h`.

### 6.2.2 `target.c`

The `target.c` file contains functions to initialize the target hardware and the development environment, as well as stopwatch functions used for timing measurements.

Function	Description
<code>init_environment()</code>	Called from the main program. Contains code for environment initialization, such as setting the bus clock for the development system.
<code>init_target()</code>	Called from the main program. Contains code to initialize the timer hardware on the target.
<code>StartupHook()</code>	Contains code to initialize and enable the periodic interrupt source. Called automatically during the execution of <code>StartOS(OSDEFAULTAPPMODE)</code> in the main program.
<code>GETSTOPWATCH()</code>	Called to make timing measurements in the Timing and Extended builds of the RTA-OSEK kernel.
<code>GetStopwatchUncertainty()</code>	Called to determine the uncertainty in the stopwatch measurements if the stopwatch timer runs at a different frequency to the CPU clock.
<code>osTraceStopwatch()</code>	Called to obtain the current time when using RTA-TRACE.

### 6.3 Building the Example Application

To build the example application

- Launch **RTA-OSEK** from the PC's start menu. This will start the RTA-OSEK GUI. From the RTA-OSEK GUI open the file `Vinstdir\target\Example\example.oil`. This file contains the operating system configuration information for the application.
- From the RTA-OSEK GUI navigation bar, select the **Builder** tab (Figure 6:1) and then select the **Custom Build** subgroup, as shown in Figure 6:2.

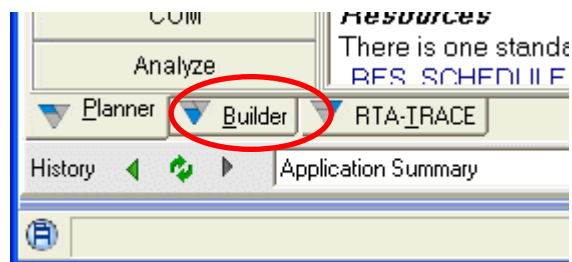


Figure 6:1 The Builder Tab

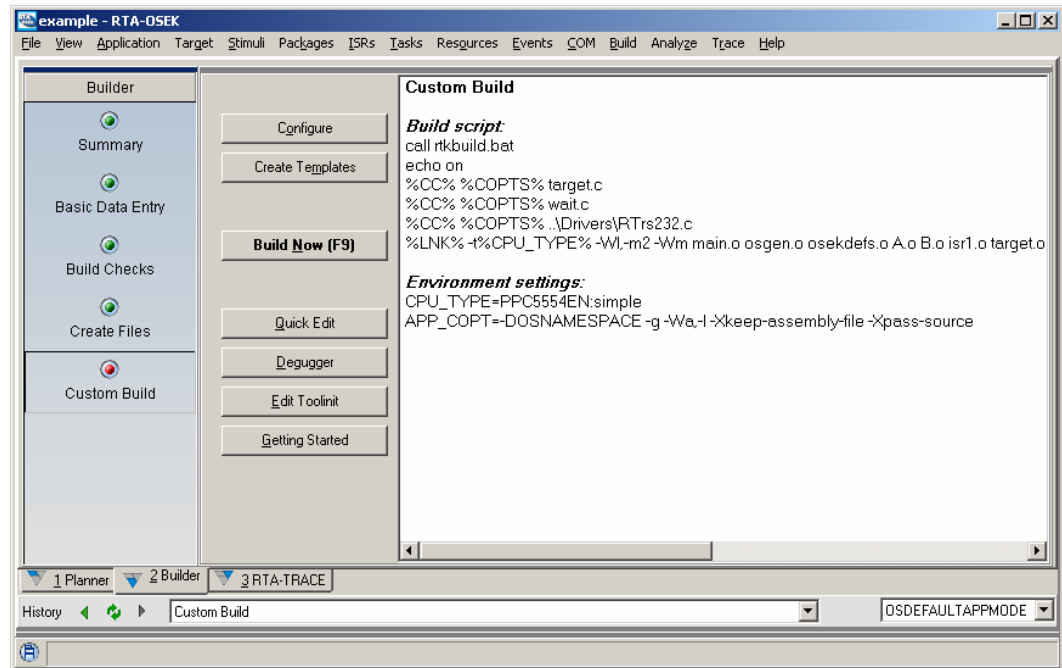


Figure 6:2 - Using the RTA-OSEK GUI Custom Build

- Click the '**Build Now (F9)**' button. This makes RTA-OSEK do two things:
  1. Create an `rtkbuild.bat` file is that contains compiler and assembler commands to build the declared objects (task A, task B and ISR `isr1`).
  2. Call the the build script that you see to perform the following actions:
    - Calls `rtkbuild.bat` to compile and/or assemble the OS objects that the tool is aware of.
    - Compiles and assembles any other code required by your application (such as resetting code and defining the target hardware).
    - Links the generated object files with the RTA-OSEK Component library.
    - Generates a binary file for downloading to your target hardware.

## 6.4 Verifying Program Execution

You can monitor task activation by connecting oscilloscope probes to the IO pins defined by `IO_PIN1` and `IO_PIN2`.

Figure 6:3 shows an oscilloscope trace of the state of the IO pins once the program is running. Each vertical gridline represents 10ms of time.

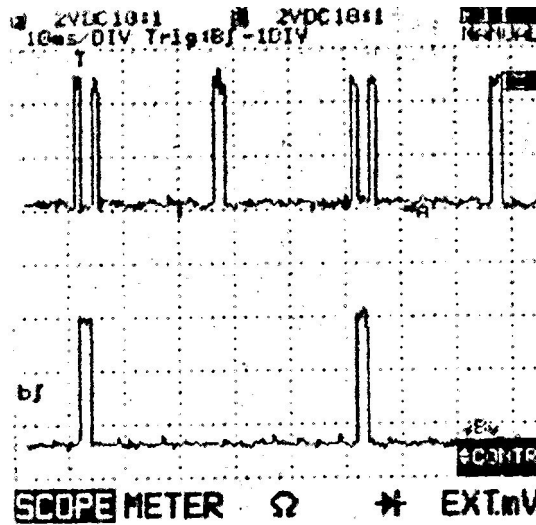


Figure 6:3 - Oscilloscope Trace

## 6.5 Troubleshooting

If your application doesn't appear to be running, you can use a debugger to verify that the ISR is being called. Place a breakpoint on the first instruction of the ISR (`osek_isr_e_isr1`) and see if your application reaches it.

If the ISR runs this means that the counter is being ticked. You can then set breakpoints on the two tasks (`osek_task_e_A` and `osek_task_e_B`) to see whether or not they run. If the tasks run, but you do not see an output, check your hardware initialization.

If your oscilloscope does not show a trace, check the settings for `IO_PIN1` and `IO_PIN2` in `target.h`. You should also check the `IO_PIN` control macros to ensure that they all reference the same I/O port on your target hardware.

If the trace shows different timing behavior, check that your timer hardware is configured correctly and that instruction rate on the target hardware matches that specified in the OIL file.

You can change the specified instruction rate in the **Target** group on the RTA-OSEK GUI navigation bar by selecting the **Timing Data** subgroup and clicking the **Change Clocks** button. You can see this in Figure 6:4 below.

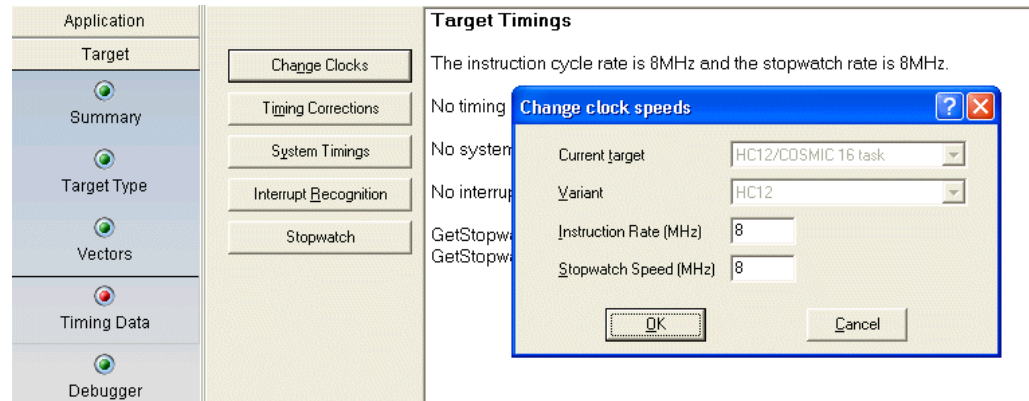


Figure 6:4 - Changing the Instruction Rate



## 7 Support

---

For product support, please contact your local ETAS representative. Office locations and contact details can be found on the ETAS Group website [www.etasgroup.com](http://www.etasgroup.com).

### 7.1 Getting Help

---

There are a number of ways to contact LiveDevices for technical support. When you contact our support team, please provide your customer number.

### 7.2 Email

---

The preferred method for dealing with support inquiries is via email. Any issues should be sent to [support@livedevices.com](mailto:support@livedevices.com)

### 7.3 Telephone

---

You can contact us by telephone during our normal office hours (0900-1730 GMT/BST). Our telephone number is +44 (0) 19 04 56 26 24

### 7.4 Fax

---

Our Fax number is +44 (0) 19 04 56 25 81

