radCASE
model-based tools

Quick Start Guide

IMACS GmbH
Mittelfeldstrasse 25
D – 70806 Kornwestheim
www.radcase.com

support@radcase.com
Tel.: +49 (0) 7154 80 83 - 0

IMACS GmbH reserves the right to make changes without further notice to any products herein. IMACS GmbH makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does IMACS GmbH assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in IMACS GmbH data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. IMACS GmbH does not convey any license under its patent rights nor the rights of others.

Copyright © IMACS GmbH 2021. All rights reserved.
Reproduction, in part or whole, without the prior written consent of IMACS GmbH is prohibited.

1 Contents

2 About This Document

The radCASE Quick Start Guide provides you with a quick summary of the steps necessary to develop an embedded system application within radCASE. The goal is to teach the reader the fundamental workflow with radCASE.

The guide has been written with the assumption that radCASE has already been installed and configured according to the Installation Manual.

This guide is not an exhaustive discussion of all radCASE concepts. For this type of information, refer to the Guidelines.
For a complete guide to radCASE modelling and design items, refer to the Reference Manual.
For a user guide of the radCASE Editor, refer to the IDE-CC Manual.
For instructions about working with the Simulation and Visualization environments, refer to the Monitoring Manual.
For information about the integration of hardware platforms, refer to the Integration Manual.

2.1 Conventions

This document uses the following conventions

Example	Usage
Open	Dialog button or menu entries
refer to *Conventions*	cross reference in document
module.c	filename or path

Table 1, Typographical conventions

Symbol	Usage
	Used for providing hints and suggestions
	Used for calling attention to specific issues and supply important information
	Used to alert for particular issues to avoid a hard time

Table 2, Symbol conventions

2.2 Acronyms

	Refers to
radCASE	Rapid Application Development Computer Aided Software Engineering
HTML	Hypertext markup language

Table 3, Acronyms

3 Workflow Overview

1. Start radCASE by using the Desktop icon or directly opening an existing project by double clicking the .rad-File

2. First of all you have to select the project to work on. You can start by *Creating A New Project* or *Opening An Existing Project*. We recommend using the Heating example for Windows PC, because you can do all described steps of this guide, without the need of any special hardware. All you need for this is Visual Studio / Visual C++ Express installed (for supported versions refer to the Installation Manual).

3. Once you opened a project, you can modify it, which is not part of this guide and is described in the Reference-Manual and the IDE-CC-Manual. For this guide just leave the content of the model as is.

4. After you finished modifying the project just proceed with the *Model Compilation*. The model compilation is the first and fundamental step of the generation process. The model is checked and all the C code, modelling the application, is generated.

5. After model compilation, by *Building The Simulation* the executable program for the simulation is created.

6. Continue by *Starting The Simulation* to simulate the model and verify the behaviour of your application.

7. Once you have verified that the model is working correctly, it is time to *Generate And Run Embedded Code* for the target.

8. When the target is up and running, it is possible to use the *Target Monitoring Software* for design level debugging of the application. At the same time it is possible to visualize the application behaviour and element values in real-time and also modify those values.

9. radCASE also provides the *Generation Of Standalone Applications*. This functionality allows saving all the PC Simulation/Visualization code in a compact form that can be supplied to customers also without a radCASE licence.

10. Finally, you can produce the full technical *Documentation* of your project in radCASE.

4 Opening The Project

4.1 Creating A New Project

If you want to start from scratch creating a new project for the first time, we recommend to check out the Tutorial.

If you are starting a new project you should use one of the prebuild templates. The usage of a new file is mainly meant for building libraries, and even though it is possible to create a new project this way, it requires expert knowledge of radCASE.

The templates provide working implementations of a nearly empty project for different screen resolutions with sensible default settings. So it is always best to use one of those templates and if necessary adjust the few settings which have to be changed for a specific project, instead of creating all settings from scratch.

1. To create a project from a template select **File > New > Project <Ctrl+Shift+N>**.

2. A dialog box opens showing the templates for different screen resolutions. The name is related to the applications display size.

   Select a template and click **OK**.

3. Select a folder where to place the project and enter a project name.
   Click **Open**.
   This will create the following directory structure:

   <Selected directory>
   Common
4. rc_lib
   <Project name>
   CTR
   Develop
   Osdl
   <Project name>.rad

   For more information on the directory structure refer to the Integration manual. For now the only thing you need to know is the .rad-File is the project file containing the project.

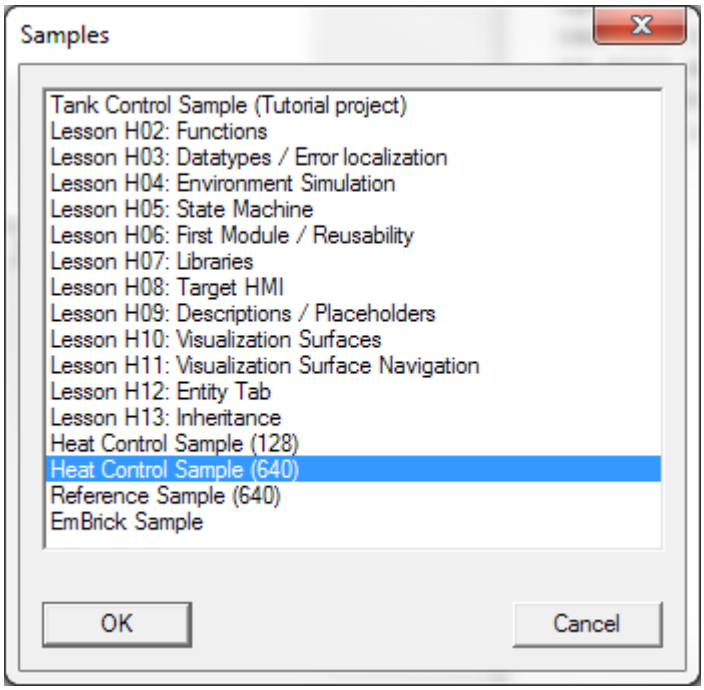
5. The project is created and is opened for editing in the radEDIT.

4.2 Opening An Existing Project

There are two options for opening a file in radCASE:

1. Double-click the project file
2. Open it from the editor (radEDIT) with the command **File > Open...** or using the icon: 

The provided samples can most easily be opened through the radEDIT menu: Help – Samples. Let's open the heating example for Windows PC by selecting "Heat Control Sample (640)":



You can also find the project in your project directory:
<project directory>\radCASESamples\Heat640\Osdl\Heat_640_WinPC.rad

If you have trouble finding your project directory, there are by default links to your project directory on the desktop and in the start menu named *radCASE Projects*.

5 Model Compilation

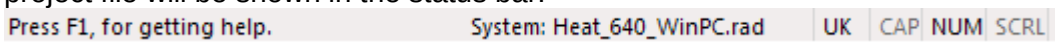
radCASE can produce C code out of the model. The generated C code will be linked against some libraries and a hardware abstraction layer (refer to Integration manual) to produce the application for the embedded system or for the simulation. The C code contains all information from the model: the static structure, dynamic behaviour, I/O mapping, user interfaces, data storage, communication, etc.

To generate the code the following steps are required:

1. Select the project file to compile

   The selected project file is not the currently active tab in the editor. The currently selected project file will be shown in the status bar:

   If there is more than one opened project, you can select the project file to compile by opening the **View > File List** and selecting the correct **Main designfile**.

2. Run the model compilation: **Build > Compile Design <F2>** or use the icon: 

3. While the model compilation is running the editor will display an animated icon showing there is still some activity: 

4. After the model compilation, if any errors occurred, the editor will show a list of errors. Most errors in that list will directly redirect to the source of error in the model, when double clicking the error message in the list.

6 Simulation

6.1 Building The Simulation

For the simulation the generated C code (refer to *Model Compilation*) is translated within a Visual Studio project and is combined with the same Target Monitoring software you will use later in this guide to connect and visualize the embedded system (refer to *Target Monitoring Software*).

1. To create the simulation you have to compile the Visual Studio project. To do this, simply select **Build > Rebuild all simulation code** or use the icon: 

2. In the same way like in the Model compilation as long as the compilation is running the activity is shown with the animated icon: 
3. If errors occurred a list of all errors will be displayed as soon as the creation finishes. In this list most errors can be double clicked to jump to the source of the error in the model.

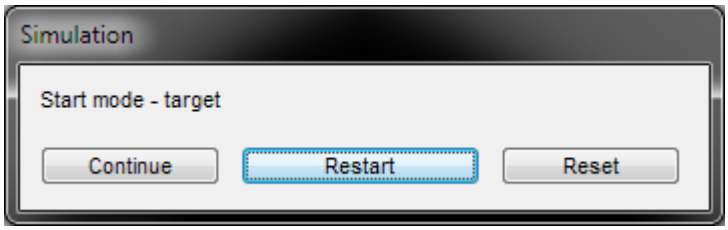
It is also possible to only translate all sources that were changed since the last simulation creation by using **Build > Create simulation** or with the icon: 

Because the simulation is a Visual Studio project, it is possible to use Visual Studio for debugging on the C source code level by using **Build > Debug Simulation** or the icon: 

6.2 Starting The Simulation

After the simulation is created you can simply start the simulation with **Build > Start simulation** or the icon: 

On start of the simulation the following dialog will appear:



Just select the preselected **Restart**.

The three buttons have the following meanings:

Reset	Will reset all permanently saved data to factory defaults
Restart	Will act like starting an embedded controller after a power loss. All permanently saved data will remain the way they were the last time the application ran.
Continue	Will act like the embedded controller was never turned off, even volatile radCASE elements will have the values of the last time the application ran.

On more information on how to work with the simulation please refer to the Monitoring Manual.

The steps until now, including the Model compilation, the simulation creation and the Start of the simulation can be done with one command by using **Build > Rebuild and start simulation <F4>** or using the icon: 

7 Generate And Run Embedded Code

7.1 Tool chain installation

First to get the model running on an embedded system, the correct toolchain for the embedded system has to be installed. This is highly hardware dependent and the information must be provided by the one who did the target integration (which is described in the Integration manual).

For our example the target embedded system is the same windows PC radCASE is running on. For this target Visual Studio 2010 (at least the cost free Express version) is needed on the PC as toolchain.

7.2 Embedded code generation

Normally the embedded code generation from the generated C source codes to a executable that can be run on the embedded system can be started out of radCASE. However this depends on the target integration and deviations from this standard should be prominently documented.

1. In a standard conform way of target integration and of course also in the mentioned example project the embedded code is generated by using **Build > Rebuild all embedded code** or use the icon: 
2. In the same way like for Model compilation and simulation creation as long as the target compilation is running the activity is shown with the animated icon: 
3. If errors occurred again a list of all errors will be displayed as soon as the creation finishes. In this list most errors can be double clicked to jump to the source of the error in the model.

It is also possible to only translate all sources that were changed since the last embedded code generation by using **Build > Create embedded code** or with the icon: 

It is possible to chain the model creation, simulation creation and embedded code generation with **Build > Complete generation** or with the icon: 

7.3 Start embedded code

To start the embedded code, the embedded code must be put on the target embedded system and be started there. This is a very hardware dependant step and it ranges from the user having to put the application on a SD-Card inserting it on the target system and setting some jumper before starting the controller and restarting it without the jumper to get the application running to being completely automatic from the editor.

As rule of thumb the embedded controller should do as much as possible to get the application to the board and running when selecting **Build > Transfer to controller** or using the icon: 

What exactly this command does and which manual steps are necessary is information which has to be provided by the one who did the target integration.

In our example after creating the embedded code it is already on the target hardware and just needs to be started, which can be done fully automatic with Build > Transfer to controller or using the icon: 

7.4 Dubgging embedded code

Debugging on the embedded controller is even more hardware dependent and in many cases not possible at all. So only in very few cases using **Build > Debug target** or using the icon  will start a debugging environment for debugging the embedded controller.

In our example however this is possible without problems, so the command **Build > Debug target** or the icon  will open the underlying Visual Studio project for debugging on C code level.

8 Target Monitoring Software

The Target Monitoring Software can be used for design level debugging and to monitor the current state of the controller. This can also be done over a modem with a controller in the field and allows in most cases for more information than the display on the embedded controller.

You can start the Target Monitoring Software to connect to your running and connected target with the command **Build > Start visualization** or the icon: 

Because in our example the embedded controller is already connected to the PC (over localhost) the visualization will directly connect and is ready to use. On how to use the Target Monitoring Software please refer to the Monitoring Manual.

Depending on the connection to the PC the connection has to be configured; this can be done from within the Target Monitoring software. For a serial connection if no configuration is found, the configuration dialog will open directly when starting the Target monitoring software.

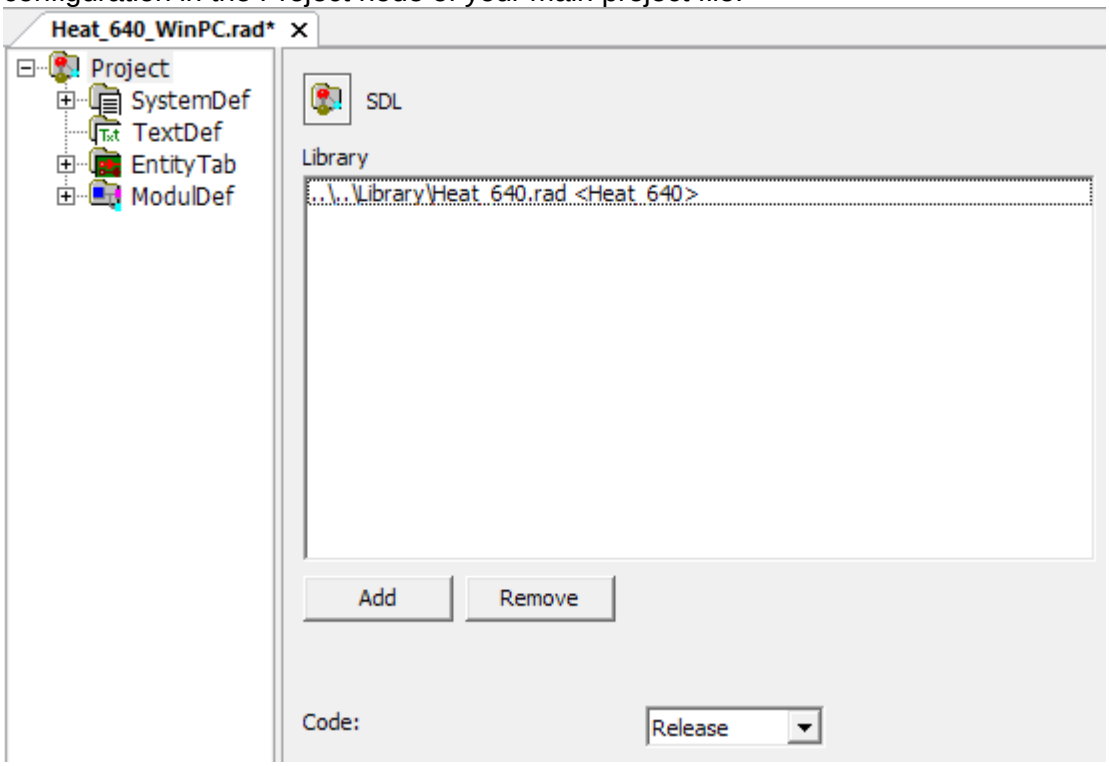
9 Generation Of Standalone Applications

The *Target Monitoring Software* including the *Simulation* can be converted to a standalone application, which will be copied into a separate directory. This standalone application can be passed to end customers who don't have a radCASE license, to connect to the embedded controller or even in the early stage of designing a project as a prototype on which to show a special functionality in the simulation.

To create the Standalone just call **Build > Create standalone** version or use the icon: 

This will create the directory *standalone* in the directory located two layers above the .rad-File of the project, so in our example in the directory *<project directory>\radCASESamples\standalone*. In this directory the simulation and monitoring tool can be started with the batch files *sim.bat* and *vis.bat*.

Only for radCASE Versions prior to 4.11:
The Standalone can only be created if the whole project was built in the Release project configuration (at least Model compilation and Simulation creation). You can find the project configuration in the Project node of your main project file:



The standalone directory will be overwritten the next time you make a standalone of a project in the project directory. If some version of the standalone has to be used again, the directory should be renamed and/or copied to another place.

10 Documentation

radCASE generates the complete technical documentation of the project. The generated documentation contains all the defined information on the design model, including the implemented **HMI** and data.
radCASE supports the generation of multilingual documentation according to the texts provided in the **TextDef** section.

The following sections describe how to generate the project documentation and open it directly from radEDIT.

10.1 Generating the documentation

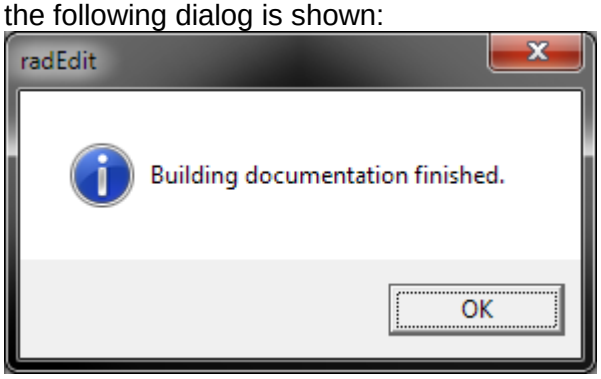
To generate the documentation do the following steps:

1. Select a language to create. This language has to be supported in the application. In our case please select UK (Englisch UK) by pressing CTRL+L in the editor and selecting that language. You can see the selected language in the status bar.

2. If you have changed the language, you have to do the *Model Compilation* and *Building The Simulation* again.

3. After this select **Build > Generate documentation** or use the icon: 

4. Wait for the documentation generation to end. When the documentation generation is finished, the following dialog is shown:



10.2 Opening the documentation

radCASE allows to open the documentation directly from radEDIT. To see the generated documentation just select **Build > Open documentation** or use the icon: 

The generated documentation is stored into the folders *.\doc* and *.\doc2*, created in the design folder (parallel to the *.\OSDL* folder in our example *<project directory>\radCASESamples\Heat640*).
The documentation is created in **HTML** and can be visualized in a browser by opening the *printall.htm* file.