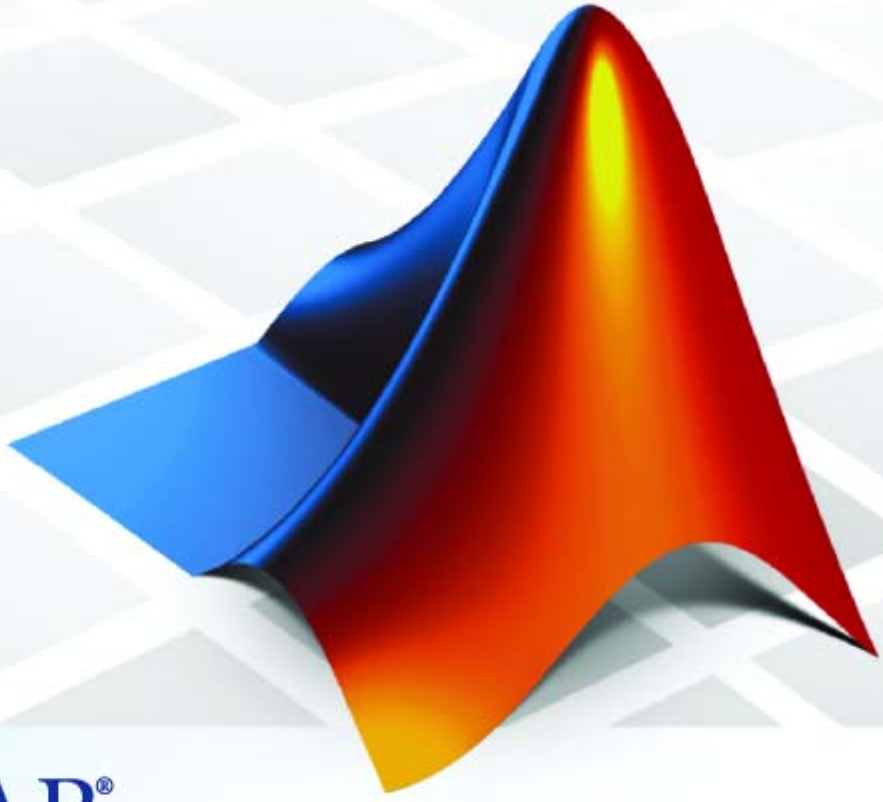
Getting Started with
Real-Time Workshop® Embedded Coder 5



MATLAB®
&SIMULINK®

The MathWorks™
Accelerating the pace of engineering and science

How to Contact The MathWorks

`www.mathworks.com`	Web
`comp.soft-sys.matlab`	Newsgroup
`www.mathworks.com/contact_TS.html`	Technical Support

`suggest@mathworks.com`	Product enhancement suggestions
`bugs@mathworks.com`	Bug reports
`doc@mathworks.com`	Documentation error reports
`service@mathworks.com`	Order status, license renewals, passcodes
`info@mathworks.com`	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

For contact information about worldwide offices, see the MathWorks Web site.

Getting Started with Real-Time Workshop Embedded Coder

© COPYRIGHT 2007 by The MathWorks, Inc.

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

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB, Simulink, Stateflow, Handle Graphics, Real-Time Workshop, SimBiology, SimHydraulics, SimEvents, and xPC TargetBox are registered trademarks and The MathWorks, the L-shaped membrane logo, Embedded MATLAB, and PolySpace are trademarks of The MathWorks, Inc.

Other product or brand names are trademarks or registered trademarks of their respective holders.

Patents

The MathWorks products are protected by one or more U.S. patents. Please see `www.mathworks.com/patents` for more information.

Revision History

September 2007 First printing New for Version 5.0 (Release 2007b)

Contents

1 Introduction to Real-Time Workshop Embedded Coder

What Is Real-Time Workshop Embedded Coder? **1-2**

Real-Time Workshop Embedded Coder Feature Summary **1-3**

What You Need to Know to Use This Product **1-5**
Prerequisites **1-5**
Real-Time Workshop Embedded Coder Documentation Collection **1-5**
Related Documentation **1-7**

Installing Real-Time Workshop Embedded Coder **1-8**

Real-Time Workshop Embedded Coder Demos **1-9**

Index

1

Introduction to Real-Time Workshop Embedded Coder

What Is Real-Time Workshop Embedded Coder? (p. 1-2)	Describes the features of Real-Time Workshop® Embedded Coder.
Real-Time Workshop Embedded Coder Feature Summary (p. 1-3)	Summary of the features of Real-Time Workshop Embedded Coder.
What You Need to Know to Use This Product (p. 1-5)	Prerequisite experience for use of Real-Time Workshop Embedded Coder; summary of related documentation.
Installing Real-Time Workshop Embedded Coder (p. 1-8)	Installation instructions.
Real-Time Workshop Embedded Coder Demos (p. 1-9)	Information on interactive demos and example code provided to help you learn about Real-Time Workshop Embedded Coder.

What Is Real-Time Workshop Embedded Coder?

Real-Time Workshop Embedded Coder is a separate, add-on product for use with Real-Time Workshop. It is intended for use in embedded systems development. Real-Time Workshop Embedded Coder generates code that is easy to read, trace, and customize for your production environment.

Real-Time Workshop Embedded Coder provides a framework for the development of production code that is optimized for speed, memory usage, and simplicity. Real-Time Workshop Embedded Coder generates optimized ANSI-C or ISO-C code for fixed-point and floating-point microprocessors. It extends the capabilities provided by Real-Time Workshop to support specification, integration, deployment, and testing of production applications on embedded targets. Real-Time Workshop Embedded Coder addresses targeting considerations such as RAM, ROM, and CPU constraints, code configuration, and code verification.

The Embedded Real-Time (ERT) target provided by Real-Time Workshop Embedded Coder is designed for customization. Most users want to generate code for a particular microprocessor or development board, and to deploy the code on target hardware with a cross-development system. To do this, some modifications to the ERT target files are required. This document and its companion, the Developing Embedded Targets document, describe how to customize the ERT target for your production requirements.

For large-scale, multi-model projects involving teams of engineers, Real-Time Workshop Embedded Coder offers Module Packaging Features (MPF) you can use to control the number and organization of files generated, the location of global identifiers, registration of user-defined data types, customized comments, and the location of target variables.

Real-Time Workshop Embedded Coder Feature Summary

In addition to supporting the features of Real-Time Workshop, Real-Time Workshop Embedded Coder:

- Generates ANSI/ISO C or C++ code and executables from Simulink® and Stateflow® models with memory usage, execution speed, and readability comparable to handwritten code
- Extends Real-Time Workshop and Stateflow Coder with the optimizations and code configuration features essential for production deployment
- Supports all Simulink data objects and data dictionary capabilities, including user-defined storage classes, types, and aliases
- Provides an intuitive graphical user interface for creating custom data
- Concisely partitions multirate code for efficient scheduling with or without a real-time operating system (RTOS)
- Provides a rich set of commenting capabilities to trace code to models and requirements
- Verifies code by automatically importing it into Simulink for software-in-the-loop testing
- Generates code documentation that is integrated with the Simulink Model Explorer and hyperlinked to the model
- Provides a Model Advisor that checks your model configuration and offers advice on how to optimize or tune a configuration set based on your stated goals or style.
- Generates an extensible main program based on information you provide for deploying the code in your real-time environment
- Generates single-rate or multirate code using periodic sample times specified in a model
- Applies a strategy called rate grouping for multirate, multitasking models, which generates separate functions for the base rate task and for each subrate task in the model
- Provides an option to easily transition between the Real-Time Workshop generic real-time (GRT) target and the Real-Time Workshop Embedded Coder embedded real-time (ERT) target

- Provides extensible module packaging features that let you package generated code to comply with specific software styles and standards
- Provides capabilities for verifying generated code, including the ability to import generated code back into Simulink as an S-function for software-in-the-loop testing with a plant model, generation of user-controlled comments and descriptions to improve code readability and traceability, inclusion of requirements in generated code, and persistent identifier names for minimizing code differences between model revisions
- Documents generated code in an HTML report that comprehensively describes code modules and model configuration settings applied during code generation
- Supports international (non-US-ASCII) characters encountered during code generation when found in Simulink block names and block descriptions, user comments on Stateflow diagrams, Stateflow object descriptions, custom TLC files, and code generation template files. For details about international character support, see Support for International (Non-US-ASCII) Characters in the Real-Time Workshop documentation.

What You Need to Know to Use This Product

In this section...
"Prerequisites" on page 1-5 "Real-Time Workshop Embedded Coder Documentation Collection" on page 1-5 "Related Documentation" on page 1-7

Prerequisites

To use Real-Time Workshop Embedded Coder, you should have basic familiarity with MATLAB®, Simulink, and Real-Time Workshop. If you have not done so, you should read:

- The tutorials in the document Getting Started with Real-Time Workshop. The tutorials provide hands-on experience in configuring models for code generation and generating code.
- The "Program Architecture" and "Models with Multiple Sample Rates" chapters of the Real-Time Workshop documentation. These sections give a general overview of the architecture and execution of programs generated by Real-Time Workshop.

Real-Time Workshop Embedded Coder Documentation Collection

The Real-Time Workshop Embedded Coder documentation collection consists of the following:

Document	Description
User's Guide	Describes ERT model execution, timing, and task management; explains how to interface to and call model code; describes default ERT code generation options; and discusses advanced configuration options.
Module Packaging Features	Explains how to use the Module Packaging Features.

Document	Description
Reference	Provides reference descriptions of Real-Time Workshop Embedded Coder functions and blocks.
Developing Embedded Targets	Describes requirements and implementation details for creating custom embedded targets based on the supplied ERT.

Related Documentation

You may be interested in the following documentation, especially if you are planning to implement custom embedded targets:

Document	Description
Real-Time Workshop User's Guide: Writing S-Functions for Real-Time Workshop	Discusses inlining and code generation issues relevant to device drivers and other S-functions
Real-Time Workshop User's Guide: Data Exchange APIs	Explains how to interface signals and parameters within generated code to your own code; combine code generated from multiple models into a single system; and implement external mode communication with your own low-level protocol layer.
Target Language Compiler	Provides details about the Target Language Compiler (TLC) needed to make nontrivial modifications to the ERT system target file, use TLC hooks into the build process, utilize information from the *model*`.rtw` file, implement inlined device drivers, or pass information into or out of the TLC phase of the build process.
Writing S-Functions	Explains how to write fully inlined S-functions. This information is necessary for developing device driver blocks for a target.

Installing Real-Time Workshop Embedded Coder

Your platform-specific MATLAB installation documentation provides all of the information you need to install Real-Time Workshop Embedded Coder.

Prior to installing Real-Time Workshop Embedded Coder, you must obtain a License File or Personal License Password (PLP) from The MathWorks. The License File or PLP identifies the products you are permitted to install and use.

If you customize your installation, the installer displays a dialog box that lets you select the MATLAB products to install. You can select and install only products for which you are licensed.

Real-Time Workshop Embedded Coder has product prerequisites, described in the following table, that must be met for proper installation and execution.

Licensed Product	Prerequisite Products	Additional Information
Simulink	MATLAB	—
Real-Time Workshop	Simulink	Requires host platform C compiler (Microsoft Visual C/C++ or Watcom C/C++). Stateflow Coder is also required when generating code for Simulink models containing Stateflow charts.
Real-Time Workshop Embedded Coder	Real-Time Workshop	Requires a cross-compiler for the target processor.

If you experience installation difficulties and have Web access, use the resources available on the MathWorks Web site Installation and Licensing page at `http://www.mathworks.com/support/install.html`.

Real-Time Workshop Embedded Coder Demos

The Real-Time Workshop demo suite contains many demos that can help you become familiar with features of Real-Time Workshop Embedded Coder and to inspect generated code. These demos illustrate features specific to Real-Time Workshop Embedded Coder and also general Real-Time Workshop features as used with Real-Time Workshop Embedded Coder.

If you are reading this document online in the MATLAB Help browser, you can open the demo suite by clicking on this link: `rtwdemos`

Alternatively, you can access the demo suite by typing the name of the demo library at the MATLAB command prompt:

```
rtwdemos
```

Most of the demos provide a button titled **Generate Code Using Real-Time Workshop Embedded Coder**. When you click this button, the demo auto-configures itself for code generation using the ERT target, and then initiates the code generation process. If your installation is licensed for Real-Time Workshop Embedded Coder, use this button.

If your installation is not licensed for Real-Time Workshop Embedded Coder, you can run most of the demos by clicking on the **Generate Code Using Real-Time Workshop** button. When you click this button, the demo auto-configures itself for code generation using the GRT target, and then initiates the code generation process. Note that the GRT target provides a subset of the capabilities of the ERT target.

Index

D

demos for Real-Time Workshop Embedded Coder 1-9

I

installation of Real-Time Workshop Embedded Coder 1-8