Embedded Target for Motorola® HC12 Release Notes

Chapter 1, "Embedded Target for Motorola HC12 1.1.3 Release Notes" describes the changes introduced in the Embedded Target for Motorola HC12 since Version 1.1.2. The following topics are discussed in these Release Notes:

- "New Features" on page 1-2
- "Major Bug Fixes" on page 1-3
- "Known Software and Documentation Problems" on page 1-4

The Embedded Target for Motorola HC12 Release Notes also provide information about recent versions of the product, in case you are upgrading from an earlier version:

- Chapter 2, "Embedded Target for Motorola HC12 1.1.2 Release Notes"
- Chapter 3, "Embedded Target for Motorola HC12 1.1.1 Release Notes"
- Chapter 4, "Embedded Target for Motorola HC12 1.1 Release Notes"
- Chapter 5, "Embedded Target for Motorola HC12 1.0 Release Notes"

Printing the Release Notes. If you would like to print the Release Notes, you can link to a PDF version.

Embedded Target for Motorola HC12 1.1.3 Release Notes

New Features

No significant new features have been introduced for Version 1.1.3 of the Embedded Target for Motorola HC12. Except for minor bug fixes, the product is essentially unchanged from Version 1.1.2.

Major Bug Fixes

To view major bug fixes made in R14SP3 for the Embedded Target for Motorola HC12, use the Bug Reports interface on the MathWorks Web site.

Note that if you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this link. You will see the bug report for the Embedded Target for Motorola HC12. The report is sorted with fixed bugs listed first, and then open bugs.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

For bug fixes added prior to R14SP2, see "Major Bug Fixes" on page 3-3 in the Embedded Target for Motorola HC12 1.1.1 release notes.

Known Software and Documentation Problems

To view important open bugs in R14SP3 for the Embedded Target for Motorola HC12, use the Bug Reports interface on the MathWorks Web site.

Note that if you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this link. You will see the bug report for the Embedded Target for Motorola HC12. The report is sorted with fixed bugs listed first, and then open bugs. You can select the Status column to list the open bugs first.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

Embedded Target for Motorola HC12 1.1.2 Release Notes

New Features

No significant new features have been introduced for Version 1.1.2 of the Embedded Target for Motorola HC12. Except for minor bug fixes, the product is essentially unchanged from Version 1.1.1.

Known Software and Documentation Problems

Model Referencing Not Supported

This release of the Embedded Target for Motorola HC12 does not support the new Simulink model referencing feature.

Blocksets Not Supported

The Embedded Target for Motorola HC12 does not support any of the Simulink blockset products.

Note on Resource Management Mechanism for Device Drivers

The Embedded Target for Motorola HC12 provides a resource management and resource collision avoidance mechanism that is subject to change in future releases. This feature is described in the "Creating Device Drivers for the Embedded Target for Motorola HC12" section of the Embedded Target for Motorola HC12 User's Guide documentation.

This feature is user extensible, allowing your custom device drivers to use the same resource management mechanisms used by the drivers provided with the target. However, note that in a future release of the Embedded Target for Motorola HC12, we will provide an resource management alternate mechanism that will obsolete the current feature. If you develop device drivers that depend on the current resource configuration and resource collision avoidance mechanisms, you will need to modify such drivers for compatibility with future releases.

Embedded Target for Motorola HC12 1.1.1 Release Notes

New Features

This section introduces the new features and enhancements added in the Embedded Target for Motorola® HC12 1.1.1.

CodeWarrior Version 3.1 for HC12 Supported

The Embedded Target for Motorola HC12 now supports Metrowerks CodeWarrior for HC12 version 3.1.

Major Bug Fixes

The Embedded Target for Motorola HC12 1.1.1 includes several bug fixes made since Version 1.1. This section describes the particularly important bug fixes.

Code Generation for the ADC Block

Two issues concerning code generation for the ADC block are corrected:

- Previously, generated code for the microcontroller would never enter the conversion wait loop, thus making the program not wait for the conversion to complete. The conversion will now run to completion.

- Previously, generated code for the microcontroller would set the ATD0CTL4/ATD1CTL4 register to 0x85 regardless of the resolution chosen by the user. ATD0CTL4/ATD1CTL4 is now set properly to the correct value: 0x85 = 8-bit resolution 0x05 = 10-bit resolution

Known Software and Documentation Problems

Model Referencing Not Supported

This release of the Embedded Target for Motorola HC12 does not support the new Simulink model referencing feature.

Blocksets Not Supported

The Embedded Target for Motorola HC12 does not support any of the Simulink blockset products.

Note on Resource Management Mechanism for Device Drivers

The Embedded Target for Motorola HC12 provides a resource management and resource collision avoidance mechanism that is subject to change in future releases. This feature is described in the "Creating Device Drivers for the Embedded Target for Motorola HC12" section of the Embedded Target for Motorola HC12 User's Guide documentation.

This feature is user extensible, allowing your custom device drivers to use the same resource management mechanisms used by the drivers provided with the target. However, note that in a future release of the Embedded Target for Motorola HC12, we will provide an resource management alternate mechanism that will obsolete the current feature. If you develop device drivers that depend on the current resource configuration and resource collision avoidance mechanisms, you will need to modify such drivers for compatibility with future releases.

Embedded Target for Motorola HC12 1.1 Release Notes

New Features

This section introduces the new features and enhancements added in the Embedded Target for Motorola® HC12 1.1.

CodeWarrior Version 3.0 for HC12 Supported

The Embedded Target for Motorola HC12 now supports Metrowerks CodeWarrior for HC12 version 3.0.

New Target Options User Interface

You can now view and edit code generation options for the Embedded Target for Motorola HC12 via the Simulink Model Explorer or the **Configuration Parameters** dialog box. The Model Explorer provides centralized access to all elements of a Simulink model or Stateflow chart. These elements include *configuration sets* that specify model parameters such as solver options, code generation options, and other choices. The **Configuration Parameters** dialog box provides a compact view of the model's active configuration set.

A configuration set includes code generation options that affect the Real-Time Workshop in general, and options that are specific to a given target, such as the Embedded Target for Motorola HC12. Configuration sets can be particularly useful in embedded systems development. By defining multiple configuration sets in a model, you can easily re-target code generation from that model. For example, one configuration set might specify the ERT target with external mode support enabled for rapid prototyping, while another configuration set might specify the Embedded Target for Motorola HC12 to generate production code for deployment of the application. Activation of either configuration set fully reconfigures the model for the appropriate type of code generation.

Before you work specifically with the Embedded Target for Motorola HC12 in the Model Explorer or **Configuration Parameters** views, you should become acquainted with configuration sets, the Model Explorer, and the revised layout of general Real-Time Workshop options. To do so, see:

• The Simulink 6.0 Release Notes for information on how to use configuration sets and the Model Explorer.

- The Real-Time Workshop 6.0 Release Notes for information on revisions and additions to the general Real-Time Workshop code generation options.

- The Real-Time Workshop Embedded Coder 4.0 Release Notes for information on revisions and additions to the general Real-Time Workshop Embedded Coder code generation options.

Only the layout and appearance of target options for the Embedded Target for Motorola HC12 have changed in this release. Functionally, these options are the same as in the previous release; see the Embedded Target for Motorola HC12 User's Guide for a detailed description of the target options.

Appearance of Target Options

In previous releases, target options were displayed in the **Simulation Parameters** dialog. Target options, organized into functional groups, were displayed under control of the **Category** menu in the Real-Time Workshop pane.

In this release, Model Explorer and **Configuration Parameters** dialog views of target options preserves the previous organization of options into categories. However, the **Category** menu has been replaced by tabbed and list selection mechanisms.

The figure below shows the Embedded Target for Motorola HC12
options displayed in the **Configuration Parameters** dialog
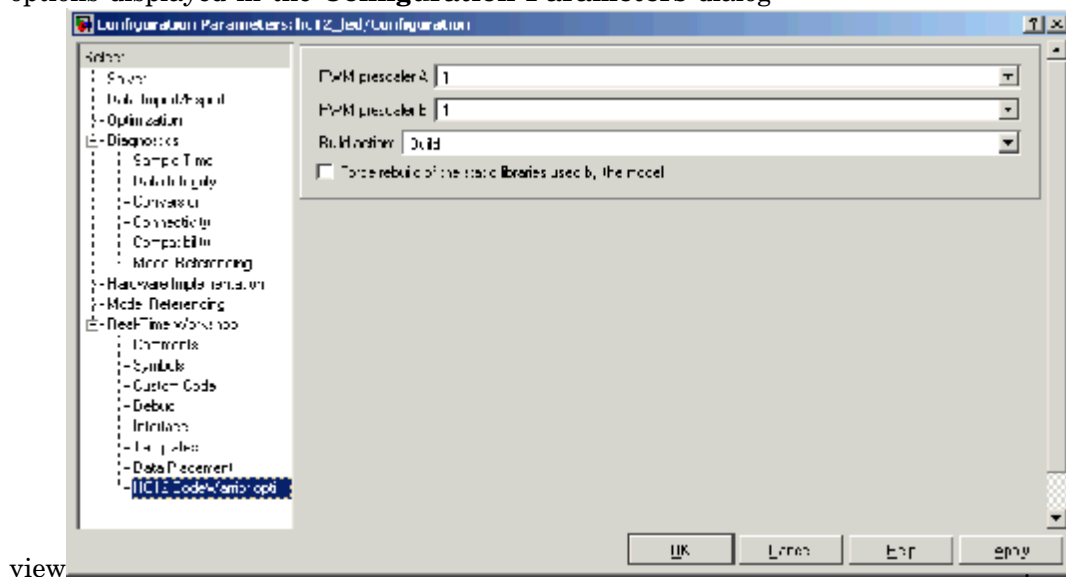
view

Major Bug Fixes

The Embedded Target for Motorola HC12 1.1 includes several bug fixes made since Version 1.0. This section describes these bug fixes.

Incorrect Library Reuse Message During Build Process Fixed

When the **Force rebuild of the static libraries used by this model option** is selected, the build process rebuilds libraries such as the Real-Time Workshop library, rtwlib.lib. However, an erroneous message was displayed that indicated that the build process was reusing an existing library, for example:

```
### Reusing stored rtw library file
C:\WINNT\Profiles\XXXXX~1.OOO\LOCALS~1\Temp\hc12\rtwintlib.lib
1 file(s) copied.
```

This problem has been fixed.

Upgrading from an Earlier Release

The issues involved in upgrading from the The Embedded Target for Motorola HC12 Version 1.0 to Version 1.1 are described below.

## Change in Options Settings for Existing Models

When you open an existing model saved in a previous release, the build process may change the values of some of the code generation options. These changes occur only in cases where an option that is not supported by the target was configured incorrectly.

The following options are forced to the values specified below:

- **Generate an example main program**: deselected
- **Suppress error status in real-time model data structure**: selected
- **Create Simulink (S-Function) block**: deselected
- **MAT-file logging**: deselected

Icon Change for Master Block

The icon for the Master block has changed, as shown below.
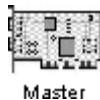
Master

Existing models will reflect this change of appearance, but will continue to operate as they did previously. there has been no functional change in the operation of the Master block.

Known Software and Documentation Problems

Model Referencing Not Supported

This release of the Embedded Target for Motorola HC12 does not support the new Simulink model referencing feature.

Blocksets Not Supported

The Embedded Target for Motorola HC12 does not support any of the Simulink blockset products.

Note on Resource Management Mechanism for Device Drivers

The Embedded Target for Motorola HC12 provides a resource management and resource collision avoidance mechanism that is subject to change in future releases. This feature is described in the "Creating Device Drivers for the Embedded Target for Motorola HC12" section of the Embedded Target for Motorola HC12 User's Guide documentation.

This feature is user extensible, allowing your custom device drivers to use the same resource management mechanisms used by the drivers provided with the target. However, note that in a future release of the Embedded Target for Motorola HC12, we will provide an resource management alternate mechanism that will obsolete the current feature. If you develop device drivers that depend on the current resource configuration and resource collision avoidance mechanisms, you will need to modify such drivers for compatibility with future releases.

Embedded Target for Motorola HC12 1.0 Release Notes

Introduction to the Embedded Target for Motorola HC12

The Embedded Target for Motorola HC12 is an add-on product for use with the Real-Time Workshop Embedded Coder. It provides a unified set of tools for developing real-time applications for the Motorola HC12/HS12 processor. This product was developed and tested with the Motorola EVB912DP256 board, which employs an MC9S12DP256 microcontroller.

The Embedded Target for Motorola HC12 generates code for the Motorola HC12 processor from Simulink block diagrams. Code generation uses the highly efficient Real-Time Workshop Embedded Coder format. Used in conjunction with Simulink, Stateflow, and the Real-Time Workshop Embedded Coder, the Embedded Target for Motorola HC12 lets you

- Design and model your system and algorithms.
- Compile, download, run and debug generated code on the target hardware, seamlessly integrating with the CodeWarrior for Motorola HC12 development environment.

The next section outlines the major features of the Embedded Target for Motorola HC12.

Feature Summary

Support for RAM and Flash Memory Models
The EVB912DP256 provides 12K RAM and 256K flash memories. The 16-bit addressing architecture allows direct access to only 64K of contiguous memory. The memory segmentation model of the EVB912DP256 uses a PPAGE (byte) register to page in 16k memory pages.

The Embedded Target for Motorola HC12 supports several memory models by providing special CodeWarrior project stationery. You can select the desired memory model from an option menu before generating code. The following memory models are supported:

- Small memory model for RAM allows access to the entire 12K of RAM.

- Small memory model for flash allows direct access of up to 64K bytes of flash memory.

- Banked memory model supports access of the entire 256K bytes of flash on the EVB912DP256board. The banked memory model can extend beyond 256K bytes for boards that include additional memory.

Modifiable Project Stationery

The Embedded Target for Motorola HC12 uses special CodeWarrior project stationery. This project stationery can be modified, extended, or replaced with your own CodeWarrior project.

Using the CodeWarrior IDE, you can modify settings of the project stationery. Project settings allow you to customize instructions provided to the assembler, compiler, linker, and debugger. You can add your own custom C code to the project stationery.

During code generation, project stationery is replicated and placed beneath your current working directory. The build process automatically invokes CodeWarrior and opens the project stationery. After the project is compiled and linked, a click on the CodeWarrior IDE start button lets you download and run your application to the EVB912DP256, via the ICD12 Background Debugging Module (BDM).

Extensible Device Driver Library

The Embedded Target for Motorola HC12 provides a library of sample I/O device drivers. The library includes the following blocks:

- ADC Input (analog-to-digital converter)

- Digital Input

- Digital Output

- Master Block

- PWM Output (Pulse Width Modulation)

You can use these device drivers as is, or modify, extend, or replace them for your needs. The drivers are fully documented to help you modify them to your requirements. The driver blocks are based on a driver paradigm that is easy

to understand and replicate. With minimum effort, you can modify device drivers and extend the device driver block library to support drivers that you may have already implemented in production applications.

The Master block is a special block that is required in every model used for code generation with the Embedded Target for Motorola HC12. The functions of the Master block are summarized in the next two sections.

Computation of Accurate, Hardware-Achievable Sample Time

The hc12_closest_st function is provided to help you set a sample time for your model that is achievable via the on-chip Clock Reset Generator (CRG). This feature lets you simulate your model using the same step size that will be used when generated code is deployed on the target hardware.

Resource Management via the Master Block

A *resource collision* in a model is a circumstance in which two or more device driver blocks are configured to use the same hardware resource. For example, a Digital Output device driver block might be configured to use PORTB as the output channel. If copies of this block are added to the model, each copy would be contending for use of PORTB, resulting in a resource collision.

The Master block maintains a *resource database* that helps guard against resource collisions. As device driver blocks are added to a model, they register resource usage with the Master block. If a resource that is already in use is requested, a resource collision error is reported via a dialog box, and the conflicting blocks are highlighted in the model. You can then correct this error by selecting an alternate resource for the block, or by eliminating one of the conflicting blocks.

Report Generation

The Embedded Target for Motorola HC12 generates an extended version of the HTML code generation report supported by the Real-Time Workshop Embedded Coder. The HTML report includes detailed information on code size for RAM and ROM. This information is obtained by post-processing the map file generated by CodeWarrior during compilation.