

CodeWarrior Development Studio for Power Architecture Processors FAQ Guide

Document Number: CWPAFAQUG
Rev. 10.x, 06/2015

Contents

Section number	Title	Page
Chapter 1		
Introduction		
1.1	Contents of this Manual.....	13
1.2	New FAQs.....	14
1.3	Accompanying Documentation.....	15
Chapter 2		
IDE and Installation		
2.1	Updating and Installing Software.....	17
2.1.1	I want to install updates for the CodeWarrior software. However, when I check for updates, no updates are found. What do I do?.....	17
2.1.2	I am facing issues in accessing the updater sites. How do I resolve this?.....	18
2.1.3	My attempt to install the CodeWarrior tool fails with the error "Another version of this product is already installed". What do I do?.....	19
2.1.4	How do I uninstall the third party plugins?.....	19
2.2	Editor.....	20
2.2.1	How can I quickly open declaration of any variable, function name, macro, or header file from within Editor?.....	21
2.2.2	How can I open definition of a function or header file from within the editor?.....	22
2.2.3	If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations?.....	23
2.2.4	How can I change the color that highlights the current line in a source code file?.....	25
2.2.5	Is it possible to view definition of a macro or a variable in the source code file?.....	26
2.2.6	Can the CodeWarrior IDE assist me in writing the source code?	27
2.2.7	How can I change the tab width/size?.....	29
2.2.8	Is it possible to display line numbers in Editor?.....	30
2.2.9	Is it possible to change the default code format settings?.....	30
2.2.10	How can I quickly switch to a header file from within the source code file?.....	32
2.2.11	How does the CodeWarrior IDE differentiate between enabled and disabled macros?.....	32
2.2.12	How can I configure predefined macros in the CodeWarrior IDE?.....	32

Section number	Title	Page
2.2.13	Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE?.....	33
2.2.14	Is it possible to roll back the changes I did to my source code?.....	34
2.2.15	Is there a way to change the default color of comments that span across multiple lines?.....	35
2.2.16	Is there a way to revert to the original contents of a source code file?.....	36
2.2.17	How do I convert Line Delimiters to the Unix format?.....	36
2.2.18	Why do I get the following error message when I try to edit a source code file?.....	37
2.2.19	What does an asterisk on the Editor title bar mean?.....	37
2.2.20	How can I find out where a function is declared in my source code?.....	37
2.2.21	How can I open .tcl extension files in the Editor window?.....	38
2.2.22	Is it possible to condense a large source file by folding/unfolding different portions of code?.....	40
2.2.23	Is there a way to determine in which workspace I am working currently?.....	41
2.2.24	How can I share same installation of the CodeWarrior software with multiple users?.....	42
2.2.25	How can I comment/uncomment a block of code in the editor?.....	42
2.2.26	How can I compare files in CodeWarrior IDE?.....	45
2.2.27	Is there a way to compare files that are not in project?.....	47
2.2.28	How do I switch between the source files in the Editor view?.....	48
2.2.29	I want to edit my source files, but all the other views are using the space as well. How do I maximize the editor view to have better working area?.....	48
2.2.30	How do I remove blank lines in the Editor view?.....	49
2.3	Workbench Window.....	49
2.3.1	What controls the initial layout of a view in a Workbench window?.....	50
2.3.2	Is it possible to restrict the number of resources that appear on my Workbench window?.....	50
2.3.3	How do I minimize a Workbench window?.....	52
2.3.4	How do I customize my Workspace perspectives?.....	52
2.3.5	How do I enable full screen mode in my CodeWarrior IDE?.....	53
2.3.6	How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with?.....	55
2.3.7	How can I view all keyboard shortcuts available in the CodeWarrior IDE?.....	56
2.3.8	How do I edit the keyboard shortcuts in the CodeWarrior IDE?.....	57
2.4	CVS.....	58

Section number	Title	Page
2.4.1	Does the CodeWarrior IDE support version control systems?.....	58
2.4.2	How can I add my project to CVS from within the CodeWarrior IDE?.....	60
2.5	Search and Replace Action.....	64
2.5.1	How do I perform a simple search in the CodeWarrior IDE?.....	65
2.5.2	Is it possible to preview the changes of the Replace action?.....	65
2.5.3	Is there a way to change the variable or the function name in a CodeWarrior project?.....	67
2.5.4	How can I search for specific files in a workspace?.....	69
2.5.5	Is there a way to replace a variable easily?.....	70
2.5.6	How can I go back to the previous location if the Editor takes me to another source or header file when, for example, I select the Open Declaration option?.....	72
2.5.7	Is there a shortcut to open the search window for a specified text that is equivalent to the Ctrl + Shift + M shortcut key available in the Classic CodeWarrior IDE?.....	72
2.5.8	Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings?	73
2.6	Miscellaneous.....	74
2.6.1	How can I improve the performance of the CodeWarrior IDE I am working with?.....	75
2.6.2	What is the first thing that I see when I start the CodeWarrior IDE?.....	76
2.6.3	What is a perspective?.....	77
2.6.4	How can I find the version of the CodeWarrior that I am using?.....	77
2.6.5	Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project?.....	77
2.6.6	How do I update my local history settings?.....	78
2.6.7	Is there a way to filter the settings in the Preferences window to find a particular setting?.....	80
2.6.8	How can I change a key binding?.....	80
2.6.9	Is it possible to view the change history of a source code file?.....	82
2.6.10	How can I view a graphical representation of source code in the CodeWarrior IDE using Dot and Doxygen?.	83
2.6.11	How can I determine which header files are associated with my source code file?.....	85
2.6.12	How do I view call hierarchy in my source code?.....	86
2.6.13	Why the project that I just created is not visible in the CodeWarrior Projects view?.....	86
2.6.14	I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why?.....	87

Section number	Title	Page
2.6.15	Why does the Rename option appear grayed out?.....	87
2.6.16	Why duplicating a configuration in the debugger perspective does not duplicate the run configuration?.....	87
2.6.17	Can I still use Ctrl+Tab keys to navigate between open windows?.....	87
2.6.18	How can I change the debugger key bindings to the ones that I used to have in the Classic CodeWarrior IDE?.....	88
2.6.19	How can I change the name of the executable that is generated when I build my project?.....	90
2.6.20	Why the console view does not display all the warning and error messages by default?.....	90
2.6.21	Is there a file that contains all the launch configuration settings?.....	91
2.6.22	How the CodeWarrior Project Importer handles recursive access paths?.....	91
2.6.23	Can I disassemble my source code file?.....	92
2.6.24	Where exactly the disassemble file gets created?.....	92
2.6.25	Why the.metadata folder in my workspace stores a huge history?.....	92
2.6.26	How to have projects from different workspaces or locations in a workspace?.....	93
2.6.27	How can I switch to another workspace?.....	93
2.6.28	What is the purpose of the Tasks view?.....	93
2.6.29	What is the purpose of the Properties view?.....	94
2.6.30	What is the purpose of the Outline view?.....	95
2.6.31	How can I find out if certain files contain debug information?.....	95
2.6.32	How can I resolve the following error message that I get when I start the CodeWarrior IDE?.....	95
2.6.33	How can I open an existing project in the CodeWarrior IDE?.....	96
2.6.34	What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button?.....	96
2.6.35	How can I start the post-build steps in the CodeWarrior IDE?.....	97
2.6.36	Is it necessary to have the project name identical to the name of the directory that contains the .project file?..	97
2.6.37	Where does the CodeWarrior IDE save the debug configuration as a local file by default?.....	97
2.6.38	When I save my launch configurations as a local file and then delete the project, all the local configurations get deleted as well. How can I resolve this issue?.....	97
2.6.39	How can I modify and save the files in GBK encoding using the CodeWarrior IDE?.....	98
2.6.40	How are the PARENT-COUNT-MyVariable definitions defined in the CodeWarrior IDE?.....	99

Section number	Title	Page
2.6.41	Is there a way to instruct the CodeWarrior IDE to use relative paths instead of absolute ones to store the project file location in a workspace?.....	99
2.6.42	What is a Working Set? How do I build/compile a Working Set?.....	99
2.6.43	How do I search within the Working Set?.....	101
2.6.44	How do I manage the Working Sets?.....	102
2.6.45	How can I can filter the projects listed in the CodeWarrior Projects view based on Working Sets?.....	103
2.6.46	How can I import/export a Working Set in the CodeWarrior eclipse IDE?.....	104
2.6.47	What are Icon decorators?.....	106
2.6.48	How do I enable or disable an icon decorator?.....	107
2.6.49	Can the icon decorators be combined?.....	108
2.6.50	When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, "You are opening a large file. Scalability mode has been turned on for this editor to help improve performance by disabling features requiring to parse the source file language." What does it mean?.....	108
2.6.51	How do I configure the Scalability settings?.....	109

Chapter 3 Project Management

3.1	FAQs - Project Management.....	111
3.1.1	How can I find out where the source files related to a CodeWarrior project are saved?.....	111
3.1.2	How can I add another source or header file to my project?.....	112
3.1.3	What is a workspace?.....	113
3.1.4	What is a project?.....	113
3.1.5	How can I view the contents of various types of source code files that appear in my project folder?.....	113
3.1.6	Is there a way to compare the Property Settings of two different projects?.....	113
3.1.7	How do I resize the various views in a Perspective?.....	114
3.1.8	How can I add a project from some other workspace to my current workspace?.....	114
3.1.9	How do I ensure that the CodeWarrior IDE saves all modifications that I make to the project properties?.....	115
3.1.10	While working with the CodeWarrior IDE, the Remote System Changed dialog box appears. Why does it appear and how can I resolve the issue?.....	115

Chapter 4 Project

Section number	Title	Page
4.1	FAQs - Project.....	119
4.1.1	How do I add a new Source Folder to my project?.....	119
4.1.2	How can I add to my project a folder that exists outside the current workspace?.....	120
4.1.3	How do I add to my project a new file that exists outside the current workspace?.....	122
4.1.4	How can I export a project that I could send to someone else?.....	122
4.1.5	How do I instruct the CodeWarrior IDE to save changes to my source code files automatically before building the project?.....	123
4.1.6	How do I compare two source code files?.....	124
4.1.7	How do I compare two directories?.....	124
4.1.8	How can I specify the number of days to keep local history for source code files?.....	125
4.1.9	Is there a shortcut to remove object code that is equivalent to the Ctrl + - shortcut key available in the Classic CodeWarrior IDE?.....	125
4.1.10	While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?.....	126
4.1.11	While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?.....	127
4.1.12	How do I enable the incremental build on my project?.....	127
4.1.13	How do I filter the items listed in the Problems view based on the project selected?.....	129

Chapter 5 Compiler

5.1	FAQs - Compiler.....	131
5.1.1	How do I compile my project?.....	131
5.1.2	Where do I see the warning and error messages?.....	132
5.1.3	When I compile my project, where can I see the commands that are being executed?.....	132

Chapter 6 Linker

6.1	FAQs - Linker.....	133
6.1.1	Can I use a different linker command file than the one being used in my project?.....	133

Chapter 7 Debugger

7.1	Breakpoints.....	135
-----	------------------	-----

Section number	Title	Page
7.1.1	How do I set breakpoints?.....	136
7.1.2	How do I view the breakpoints that have been set?.....	136
7.1.3	How do I disable existing breakpoints?.....	137
7.1.4	What does the checkmark next to the breakpoint represent?.....	137
7.1.5	What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options?.....	138
7.1.6	Which breakpoint option do I select if I want to add a breakpoint?.....	138
7.1.7	Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list?.....	138
7.1.8	What if I want to delete a breakpoint?.....	138
7.1.9	How do I place a breakpoint on a given physical address?.....	138
7.1.10	How do I add hardware breakpoints? Can I add them through the GUI?.....	139
7.1.11	How can I view the hardware breakpoints that have been set?.....	139
7.1.12	What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other?.....	140
7.1.13	What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window?.....	140
7.1.14	Is there a way to add breakpoints to C and Assembly files that exist in a library?.....	140
7.1.15	How do I set a breakpoint at a particular function?.....	141
7.1.16	What is the difference between a breakpoint and a line breakpoint?.....	141
7.1.17	The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing?.....	141
7.1.18	While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do?.....	142
7.1.19	Can I use software breakpoints on code running in flash memory?.....	142
7.1.20	What is a hardware breakpoint?.....	143
7.1.21	How do I select the type of breakpoint for my application in the Editor view?.....	143
7.2	Variables.....	143
7.2.1	By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?.....	144
7.2.2	How do I debug my application using global variables?.....	144
7.2.3	How do I filter variables in files?.....	145

Section number	Title	Page
7.2.4	How do I enable watch on a variable?.....	145
7.2.5	What is a static local variable?.....	146
7.2.6	When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do?.....	146
7.3	Build Configuration.....	146
7.3.1	How do I create a new build configuration for my project?.....	147
7.3.2	Is it possible to exclude a single or multiple files from the project while building it?.....	148
7.3.3	Is it possible to configure build settings for a particular source code file?.....	148
7.3.4	Is it possible to store output of the build configuration of my project in a separate directory?.....	149
7.3.5	How can I disable the option of having the CodeWarrior IDE build my project before launching it?.....	150
7.3.6	How can I export the launch configurations settings and then import them back into the project?.....	150
7.3.7	How do I ensure that the launch configurations are a part of the project when it is exported?.....	151
7.3.8	I am unable to specify a JTAG configuration file because the option is disabled. Is there a way to specify it?.	152
7.4	Flash Programmer.....	152
7.4.1	How can I create the flash programmer, hardware diagnostics, or import/export fill memory tasks in the CodeWarrior IDE?.....	153
7.4.2	I need to flash a large size application (more than 64 KB) , and I also need to perform target tasks, such as secure/unsecure the device and additional verify or erase. How do I do it?.....	153
7.4.3	Is there a way to quickly flash multiple boards with different S-Records?.....	155
7.4.4	Is it possible to debug the target multiple times with the same binary application, without flashing the device?.....	155
7.4.5	I need to enable logging for flash programming. How do I do it?.....	156
7.4.6	How can I program my boards without a project?.....	157
7.5	Miscellaneous.....	158
7.5.1	How can I view only certain registers?.....	159
7.5.2	How can I view addresses in memory?.....	160
7.5.3	How do I open up multiple memory locations in the memory view?.....	161
7.5.4	How can I view variables that I have set in my project?.....	162
7.5.5	How can I change the values of variables?.....	163
7.5.6	How do I view global variables that have been accessed in my project?.....	163

Section number	Title	Page
7.5.7	Is the command-line debugger available in the Eclipse-based CodeWarrior IDE as well?.....	164
7.5.8	Is it possible to specify a different name for the binary file (.elf)?.....	164
7.5.9	Can I debug the code from the generated assembly file step-by-step?.....	165
7.5.10	If I have a debugger session running on my application, how can I halt the application?.....	165
7.5.11	Why do I get the following error message when I attempt to debug my project?.....	166
7.5.12	Can I change the format of the Memory view?.....	166
7.5.13	Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective?.....	168
7.5.14	Is there a way to display the variables always in the Hexadecimal format?.....	168
7.5.15	Can I change the endianness that is displayed in the Memory view?.....	169
7.5.16	In the Memory view, is it possible to go back to the address that the rendering was created for?.....	171
7.5.17	Is there a way to view disassembly at any location?.....	172
7.5.18	Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format?	172
7.5.19	What does the Reset on Connect option do?.....	174
7.5.20	Can I change the Program Counter (pc) value while in the debugger without having to restart the debugger?..	174
7.5.21	How do I specify the program entry point?.....	174
7.5.22	How can I run a debug session repeatedly?.....	175
7.5.23	Can I direct console output to a file instead of the console window?.....	175
7.5.24	How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration?..	176
7.5.25	How can I inspect global variables in the variables view?.....	177

Chapter 8 Debugger Shell

8.1	FAQs - Debugger Shell.....	179
8.1.1	How can I find the default value of memory width? How can I change the value if I want to?.....	179
8.1.2	Can I change the Program Counter (pc) value?.....	180
8.1.3	Does the config page off option still available in the Eclipse-based CodeWarrior IDE?.....	180
8.1.4	Is there a way to accelerate the execution speed and turn off the printing when I use a lot of mem commands in the debugger shell?.....	180

Chapter 9

Section number	Title	Page
Profiling and Analysis Tool		
9.1	FAQs -Tracing and Analysis.....	183
9.1.1	How is tracing useful for the user?.....	184
9.1.2	How do I enable trace and profiling?.....	184
9.1.3	How do I collect trace data?.....	184
9.1.4	How do I view trace data?.....	185
9.1.5	What are the different types of fields displayed in the Trace viewer and what type of information is provided in them?.....	185
9.1.6	Where does the trace data get saved?.....	186
9.1.7	How do I import a trace data file?.....	186
9.1.8	How do I configure Trace profile(s)?.....	187
9.1.9	What are the various Trace debug connection types I can choose from?.....	187
9.1.10	What are tracepoints?.....	188
9.1.11	What are software and hardware tracepoints?.....	188
9.1.12	What are start and stop tracepoints?.....	188
9.1.13	How do I view the tracepoints that I have set?.....	188
9.1.14	What is Performance Analysis?.....	189
9.1.15	How do I create working sets?.....	189
9.1.16	How do we add scenarios to a Performance Analysis configuration?.....	190
9.1.17	What is Performance Analysis perspective used for?.....	190
9.1.18	What are scenarios?.....	190
9.1.19	What are Nexus Messages?.....	190
9.1.20	What is Exclude Symbols feature used for?.....	191
9.1.21	What are Custom Scenarios?.....	191

Chapter 1

Introduction

This FAQ Guide lists most frequently asked or most anticipated questions about Power Architecture V10.x.

In this document, the FAQs are divided into various categories based on the feature they are associated with, such as Installation, IDE, Editor, Project Management, Project, Compiler, Assembler, Linker, Debugger, Debugger Shell, USBTAP, Ethernet TAP, and Profiling and Analysis Tools.

In this chapter:

- [Contents of this Manual](#) - Describes the contents of this manual.
- [What's New](#) - Lists new FAQs in this manual.
- [Accompanying Documentation](#) - Describes supplementary CodeWarrior documentation, third-party documentation, and references.

1.1 Contents of this Manual

This chapter describes the contents of this manual.

The following table lists and describes each chapter in this manual. Each chapter lists a specific category of FAQs.

Table 1-1. Manual Contents

Chapter	Description
IDE and Installation	Lists IDE related FAQs
Project Management	Lists project management related FAQs
Project	Lists project related FAQs
Compiler	Lists compiler related FAQs
Linker	Lists linker related FAQs
Debugger	Lists debugger related FAQs
Debugger Shell	Lists debugger shell related FAQs
Profiling and Analysis Tool	Lists profiler related FAQs

1.2 New FAQs

The manual includes these new FAQs for the current release.

- **NEW** How do I switch between the source files in the Editor view?
- **NEW** I want to edit my source files, but all the other views are using the space as well. How do I maximize the editor view to have better working area?
- **NEW** How do I remove blank lines in the Editor view?
- **NEW** How do I customize my Workspace perspectives?
- **NEW** How do I enable full screen mode in my CodeWarrior IDE?
- **NEW** How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with?
- **NEW** How can I view all keyboard shortcuts available in the CodeWarrior IDE?
- **NEW** How do I edit the keyboard shortcuts in the CodeWarrior IDE?
- **NEW** What is a Working Set? How do I build/compile a Working Set?
- **NEW** How do I search within the Working Set?
- **NEW** How do I manage the Working Sets?
- **NEW** How can I filter the projects listed in the CodeWarrior Projects view based on Working Sets?
- **NEW** How can I import/export a Working Set in the CodeWarrior eclipse IDE?
- **NEW** What are Icon decorators?
- **NEW** How do I enable or disable an icon decorator?
- **NEW** Can the icon decorators be combined?
- **NEW** When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, "You are opening a large file. Scalability mode has been turned on for this editor to help improve performance by disabling features requiring to parse the source file language." What does it mean?
- **NEW** How do I configure the Scalability settings?
- **NEW** While working with the CodeWarrior IDE, the Remote System Changed dialog box appears. Why does it appear and how can I resolve the issue?
- **NEW** While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?
- **NEW** While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?
- **NEW** How do I enable the incremental build on my project?
- **NEW** While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do?
- **NEW** Can I use software breakpoints on code running in flash memory?

- **NEW** What is a hardware breakpoint?
- **NEW** How do I select the type of breakpoint for my application in the Editor view?
- **NEW** By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?
- **NEW** How do I debug my application using global variables?
- **NEW** How do I filter variables in files?
- **NEW** How do I enable watch on a variable?
- **NEW** What is a static local variable?
- **NEW** When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do?
- **NEW** Is it possible to configure build settings for a particular source code file?
- **NEW** How can I program my boards without a project?
- **NEW** What is Performance Analysis perspective used for?
- **NEW** What are scenarios?
- **NEW** What are Nexus Messages?
- **NEW** What is Exclude Symbols feature used for?
- **NEW** What are Custom Scenarios?

1.3 Accompanying Documentation

The **Documentation** page describes the documentation included in this version of CodeWarrior Development Studio for Power Architecture Processors.

You can access the **Documentation** by any of the following methods:

- a shortcut link on the Desktop that the installer creates by default.
- opening the `START_HERE.html` in `CWInstallDir\PA\Help` folder.



Chapter 2

IDE and Installation

In this chapter, you find Frequently Asked Questions (FAQs) related to the CodeWarrior IDE.

The FAQs listed in this chapter are divided into following categories:

- [Updating and Installing Software](#)
- [Editor](#)
- [Workbench Window](#)
- [CVS](#)
- [Search and Replace Action](#)
- [Miscellaneous](#)

2.1 Updating and Installing Software

In this topic, installation related FAQs are listed.

- [I want to install updates for the CodeWarrior software. However, when I check for updates, no updates are found. What do I do?](#)
- [I am facing issues in accessing the updater sites. How do I resolve this?](#)
- [My attempt to install the CodeWarrior tool fails with the error "Another version of this product is already installed". What do I do?](#)

2.1.1 I want to install updates for the CodeWarrior software. However, when I check for updates, no updates are found. What do I do?

If you are using **Help > Check for Updates** option to check for new service packs or patches, you may get the message "There is nothing to update", even when patches and service packs are available.

This happens because the **Check for Updates** option, checks for updates available for already installed packages or features. The **Check for Updates** option scans through the list of the installed plugins and contacts the configured and enabled update sites (for your reference, see the image listed below).

To install new service packs or patches, use the **Help > Install New Software** option. While installing new software, make sure that you clear the **Contact all update sites during install to find required software** checkbox.

Also, note that if the CodeWarrior software needs to access updates through proxy, you need to configure the network connections.

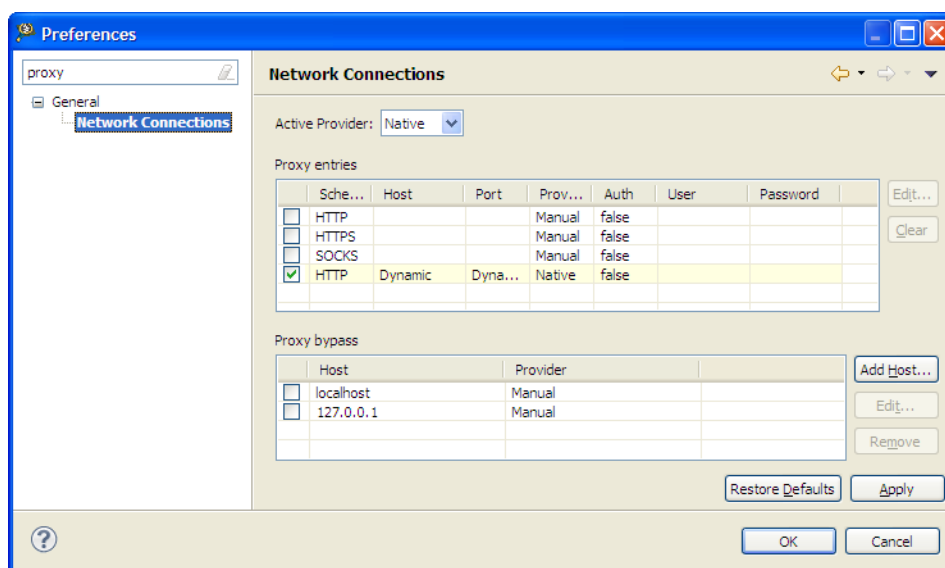


Figure 2-1. Network Connections

2.1.2 I am facing issues in accessing the updater sites. How do I resolve this?

The **Install New Software** option requires minor modification to the CodeWarrior V10.x site pointers to enable correct behavior.

If you have problem to access an updater site, add a backslash to the end.

For example, add a backslash at the end of the Freescale update site:

http://freescale.com/lgfiles/updates/Eclipse/PA10_0_2/com.freescale.pa.update.site_win/

http://freescale.com/lgfiles/updates/Eclipse/PA10_0_2/com.freescale.pa.update.site_lin/

2.1.3 My attempt to install the CodeWarrior tool fails with the error "Another version of this product is already installed". What do I do?

This happens when while un-installing the CodeWarrior tool, the installer crashes in between. The files are removed from the hard drive and the icons are removed from the Start menu. Also, the CodeWarrior tool does not appear in the Add/Remove Programs panel. But, your system will not let you install another version of the CodeWarrior tool, as there are still some files remaining.

To remove the remaining CodeWarrior files:

1. Open a command prompt and go to the folder containing the `setup.exe` of the CodeWarrior tool you want to uninstall.
2. In this folder, edit the `setup.ini` file and search for the value, `ProductCode`.
3. Enter the following command at the command prompt:

```
msiexec /i[ProductCode]
```

The CodeWarrior setup dialog box appears.

4. Repair or remove the current installation.
5. Remove the remaining files.
6. Re-launch the setup of the CodeWarrior tool you want to install.

You can now successfully install another version of the CodeWarrior tool.

2.1.4 How do I uninstall the third party plugins?

To uninstall a third party plugin, follow the steps listed below:

1. Select **Help > Install New Software** from the IDE menu bar.

The **Install** dialog box appears.

2. Click the **What is already installed?** option.

The **CodeWarrior Development Studio Installation Details** dialog box appears.

3. Select the desired plugin and click **Uninstall**.

The **Uninstall** dialog box appears.

4. Click **Finish**.

The IDE uninstalls the selected plugin.

2.2 Editor

In this topic, Editor related FAQs are listed.

- [How can I quickly open declaration of any variable, function name, macro, or header file from within Editor?](#)
- [How can I open definition of a function or header file from within the editor?](#)
- [If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations?](#)
- [How can I change the color that highlights the current line in a source code file?](#)
- [Is it possible to view definition of a macro or a variable in the source code file?](#)
- [Can the CodeWarrior IDE assist me in writing the source code?](#)
- [How can I change the tab width/size?](#)
- [Is it possible to display line numbers in Editor?](#)
- [Is it possible to change the default code format settings?](#)
- [How can I quickly switch to a header file from within the source code file?](#)
- [How does the CodeWarrior IDE differentiate between enabled and disabled macros?](#)
- [How can I configure predefined macros in the CodeWarrior IDE?](#)
- [Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE?](#)
- [Is it possible to roll back the changes I did to my source code?](#)
- [Is there a way to change the default color of comments that span across multiple lines?](#)
- [Is there a way to revert to the original contents of a source code file?](#)
- [How do I convert Line Delimiters to the Unix format?](#)
- [Why do I get the following error message when I try to edit a source code file?](#)
- [What does an asterisk on the Editor title bar mean?](#)
- [How can I find out where a function is declared in my source code?](#)
- [How can I open .tcl extension files in the Editor window?](#)
- [Is it possible to condense a large source file by folding/unfolding different portions of code?](#)
- [Is there a way to determine in which workspace I am working currently?](#)

- How can I share same installation of the CodeWarrior software with multiple users?
- How can I comment/uncomment a block of code in the editor?
- How can I compare files in CodeWarrior IDE?
- Is there a way to compare files that are not in project?
- **NEW** How do I switch between the source files in the Editor view?
- **NEW** I want to edit my source files, but all the other views are using the space as well. How do I maximize the editor view to have better working area?
- **NEW** How do I remove blank lines in the Editor view?

2.2.1 How can I quickly open declaration of any variable, function name, macro, or header file from within Editor?

To quickly open declaration of any variable, function name, macro or header file from within Editor, press the **Ctrl** key and click the text. Alternatively, place the mouse cursor on the text and perform either of the following:

- press F3,
- right-click and select **Open Declaration** (as the figure, *Open Declaration* listed below shows), or
- select **Navigate > Open Declaration** from the IDE menu bar (as the figure, *Navigate Menu* listed below shows).

This will open the logical file associated with the selected text. For example, for a header file include, the header file will open. For variables, the file containing the definition or declaration of the variable will open.

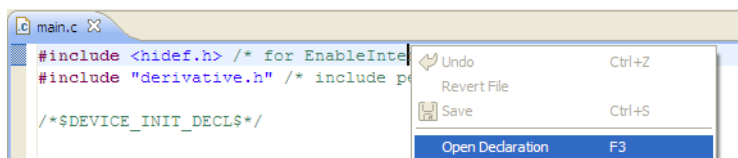


Figure 2-2. Open Declaration

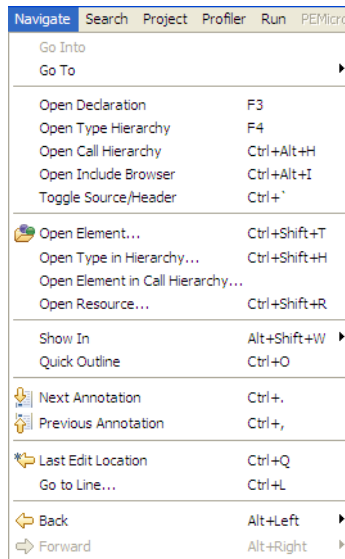


Figure 2-3. Navigate Menu

The **Navigate** menu lets you perform various navigation actions in the source file, such as opening header files, viewing macro or variable definitions, navigating to the last edit location or previous source file. You can also perform these actions using the keyboard shortcuts.

2.2.2 How can I open definition of a function or header file from within the editor?

The CodeWarrior Eclipse IDE uses an indexer to build index of all sources in the background. To open definition, you need to configure indexer preferences. The indexer can be configured per project or per workspace basis.

1. Select **Window > Preferences** .

The Preferences dialog box appears.

2. Type **indexer** as the filter text to narrow down the list of preferences.
3. Select **C/C++ > Indexer** .

The **Indexer** page appears in the right panel.

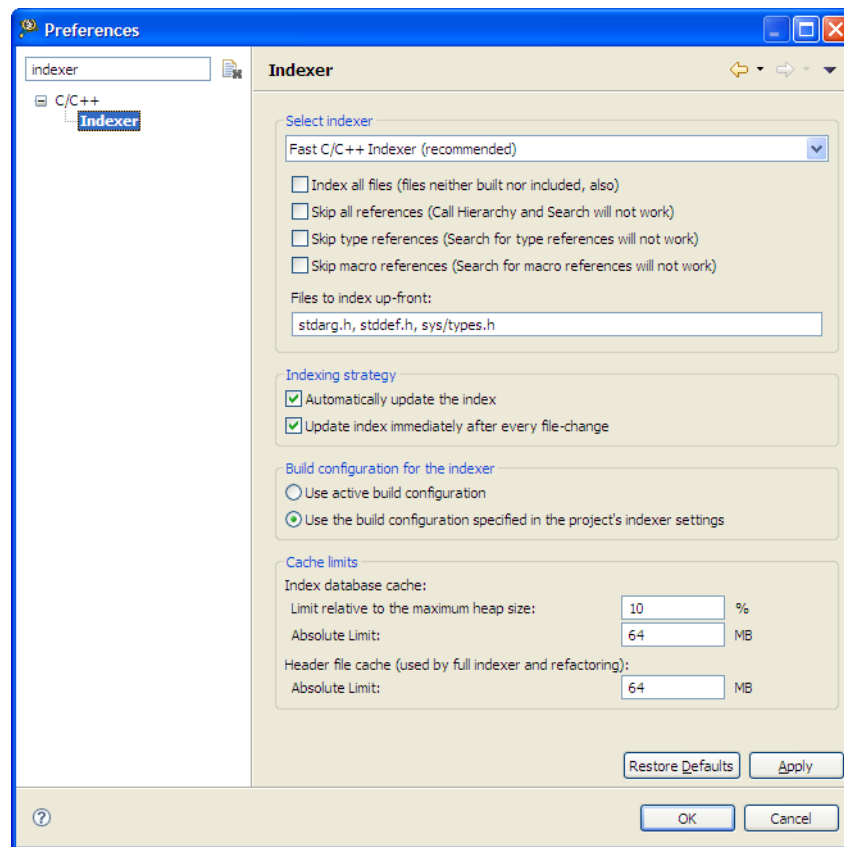


Figure 2-4. Indexer Page

4. From the **Select indexer** drop-down list, select the **Full C/C++ Indexer** option.
5. Click **OK**.
6. Right-click the project with which the source file is associated in the **CodeWarrior Projects** view and select **Index > Rebuild**.
7. Press F3, definition will open.

```
void TestFunc(void) {
}
```

Figure 2-5. Function Definition

8. Press F3 again to switch to the function declaration.

```
void TestFunc(void);
```

Figure 2-6. Function Declaration

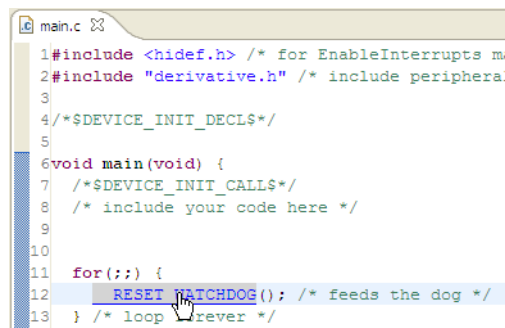
9. Press ALT + Left to switch to the original location.

Repeat step 7-9 to open function declarations and definitions of other functions in the source file.

2.2.3 If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations?

If you are using a Macintosh computer, just keep the CTRL key pressed and move the mouse cursor over the header file or function whose declaration or definition you want to open. The hyperlink for that header file or function enables.

Click the hyperlink to switch to the required declaration or definition, as the following figure shows:



```
main.c
1#include <hidef.h> /* for EnableInterrupts m
2#include "derivative.h" /* include periphera
3
4/*$DEVICE_INIT_DECL$*/
5
6void main(void) {
7 /*$DEVICE_INIT_CALLS*/
8 /* include your code here */
9
10
11 for(;;) {
12     RESET WATCHDOG(); /* feeds the dog */
13 } /* loop forever */
```

Figure 2-7. Hyperlink in Source Files

By default, the modifier key for hyperlinks in editor is Ctrl; however, you can change the modifier key as per your preference.

1. Select **Window > Preferences**.

The **Preferences** dialog box appears.

2. Type **Hyperlink** as the filter text to narrow down the list of preferences.
3. Select **General > Editors > Text Editors > Hyperlinking**.

The **Hyperlinking** page appears in the right panel.

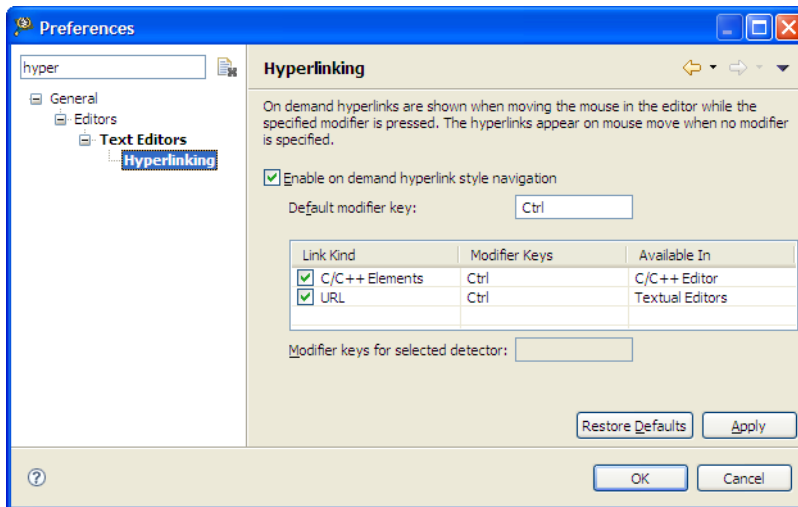


Figure 2-8. Hyperlinking Page

4. Enter the required modifier key in the **Default modifier key** text box.
5. Click **OK**.

The modifier key changes, as specified.

2.2.4 How can I change the color that highlights the current line in a source code file?

To change the color for the current line highlight in a source file:

1. Select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Type **text editor** as the filter text to narrow down the list of preferences.
3. Select **General > Editors > Text Editors** .

The **Text Editors** page appears in the right panel.

4. Ensure that **Highlight current line** checkbox is checked.
5. Select **Current line highlight** in the **Appearance color options** list.

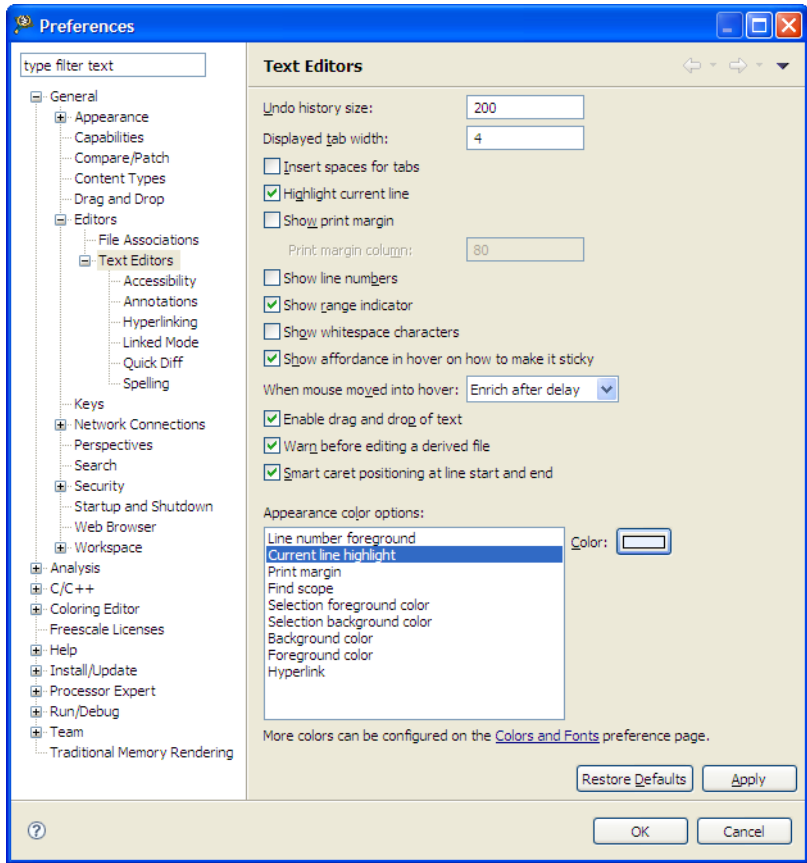


Figure 2-9. Text Editors Page > Current line highlight Selected

6. Click the **Color** button to open the color panel, and select the required color.
7. Click **OK** to close the color panel.
8. Click **OK** to save the settings.

The current line highlight color changes, as specified.

2.2.5 Is it possible to view definition of a macro or a variable in the source code file?

Yes, you can open definition of a macro or variable by hovering the mouse cursor over the macro or variable. A small popup window displaying the definition of the macro (refer to the image, *Macro Definition* as listed below) or variable (refer to the image, *Variable Declarations and Occurrences* as listed below) appears.

Also, the IDE highlights all the occurrences of the variable in the source file, so you do not need to search for it manually.

```
for(;;) {
    _RESET_WATCHDOG(); /* feeds the dog */
}
/*
Macro Expansion
{asm sta _SRS.Byte;} at you never leave main */
Press 'F2' for focus
*/
}
```

Figure 2-10. Macro Definition

```
timer.c | Timer.txt | keyboard.h | events.c | main.c | trigger.c
64
65 #if 1
66 void DBNC_Process (DBNC_FSMData *data) {
67     DBNC_KeySet keys;
68
69     switch (KBD_KeyState) {
70     case static KBD_KeyStateKinds KBD_KeyState = KBD_KEY_IDLE;
71         KBD_KeyState = KBD_KEY_IDLE;
72         KBD_LongKeyCount = 0;
73         KBD_KeyState = KBD_KEY_PRESSED; /* advance to next state */
74         TRG_AddTrigger (TRG_KEYPRESS, KBD_KEY_IDLE_TO_PRESSED_NOF_1
75         break;
76
77     case KBD_KEY_PRESSED:
78         keys = data->GetKeysFn();
79         if (keys == KBD_ScanValue) { /* still pressing the same ke
80             KBD_LongKeyCount++;
81             if (KBD_LongKeyCount >= (KBD_LONG_KEY_TIME_MS / (KBD_KEY_F
82                 /* yes, long key press detected */
83                 KBD_LongKeyCount = KBD_LONG_KEY_ITERATIONS; /* avoid c
84                 data->LongKeyEventFn (keys);
85                 KBD_KeyState = KBD_KEY_WAIT_RELEASE; /* advance to nex
86                 TRG_AddTrigger (TRG_KEYPRESS, KBD_KEY_PRESSED_TO_WAIT_F
87             } else {
88                 TRG_AddTrigger (TRG_KEYPRESS, KBD_KEY_PRESSED_TO_PRESSE
89             }
90         } else { /* we got another key set pressed: so it was only
91             data->ShortKeyEventFn (KBD_ScanValue);
92             KBD_KeyState = KBD_KEY_WAIT_RELEASE; /* advance to next
93             TRG_AddTrigger (TRG_KEYPRESS, KBD_KEY_WAIT_RELEASE_TO_WAI
94         }
```

Figure 2-11. Variable Declaration and Occurrences

2.2.6 Can the CodeWarrior IDE assist me in writing the source code?

Yes, CodeWarrior IDE can parse the source files in the background. Therefore, it provides you the code completion feature. To use the code completion feature, select **Edit > Content Assist** from the IDE menu bar.

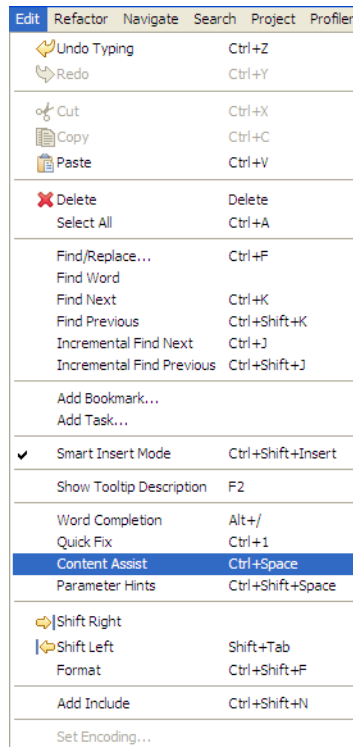


Figure 2-12. Edit > Content Assist

The Content Assist feature allows you to view the list of the field member of a structure, class, or union. The list of the field members appears automatically the moment you type `.` following the structure, class, or union name (refer to the image listed below). You can also press **Ctrl+Space** to view the list.

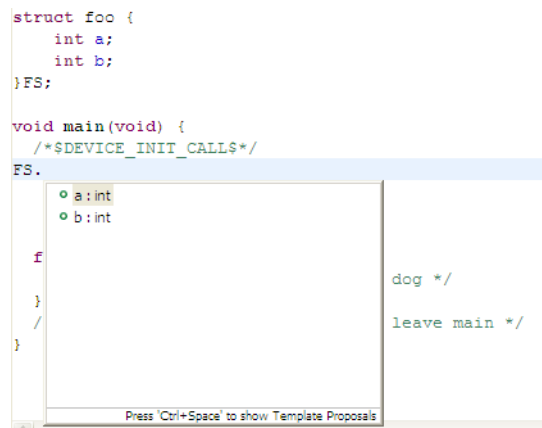


Figure 2-13. View Field Members of Structure

If you press **Ctrl+Space** without typing anything or by placing the cursor in a blank line in the source file, the complete list appears, as shown in the figure listed below.

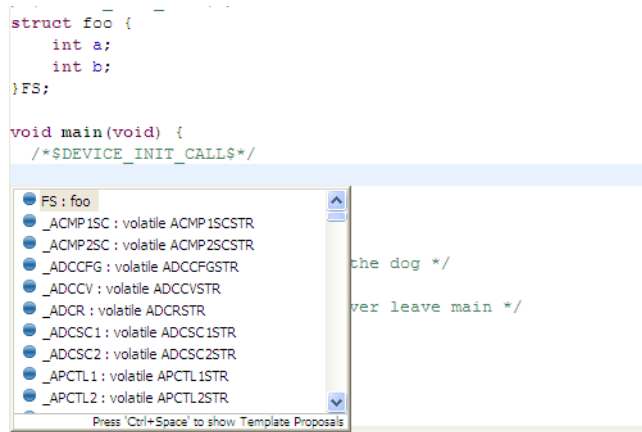


Figure 2-14. Content Assist - Without Typing Anything

You can also view the list of bitfields using Content Assist, as shown in the figure listed below.

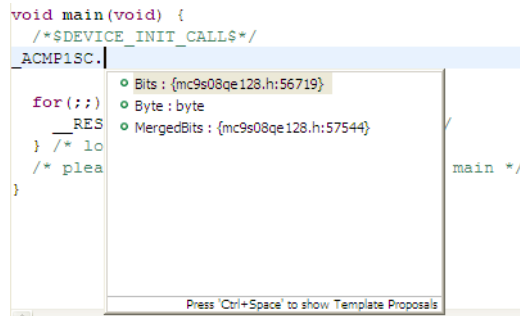


Figure 2-15. Bitfields

2.2.7 How can I change the tab width/size?

By default, the tab size in the CodeWarrior IDE is 4. To change the default tab size:

1. From the IDE menu bar, select **Window > Preferences**.
2. Type **tab** as the filter text to narrow down the list of preferences.
3. Select **Text Editors**.

The **Text Editors** preference page appears in the right panel of the **Preferences** dialog box.

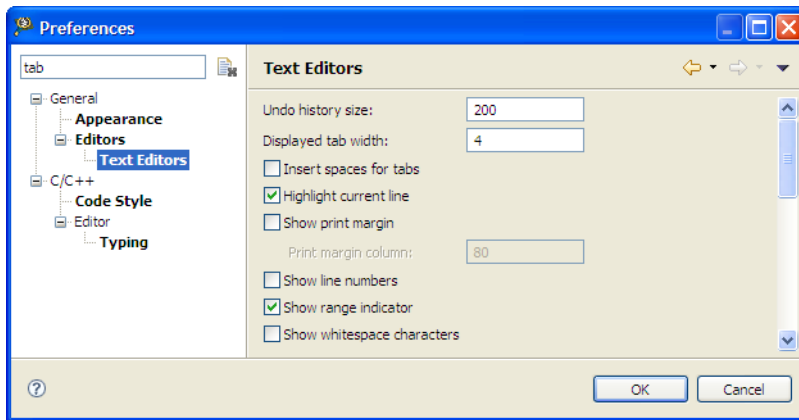


Figure 2-16. Text Editors Preferences Page

4. Enter the desired tab size in the **Displayed tab width** text box.
5. Click **OK**.

The default tab size changes.

2.2.8 Is it possible to display line numbers in Editor?

Yes, to display line numbers in the editor area:

1. From the IDE menu bar, select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Type **number** as the filter text to narrow down the list of preferences.
3. Select **Text Editors** . The **Text Editors** preference page appears in the right panel of the **Preferences** dialog box.
4. Check the **Show line numbers** checkbox.
5. Click **OK** .

The editor will display line numbers.

2.2.9 Is it possible to change the default code format settings?

Yes, to change the default code format settings:

1. From the IDE menu bar, select **Window > Preferences**.

The **Preferences** dialog box appears.

2. Type **code** as the filter text to narrow down the list of preferences.
3. Select **Code Style** in the left pane to modify the code formatting.

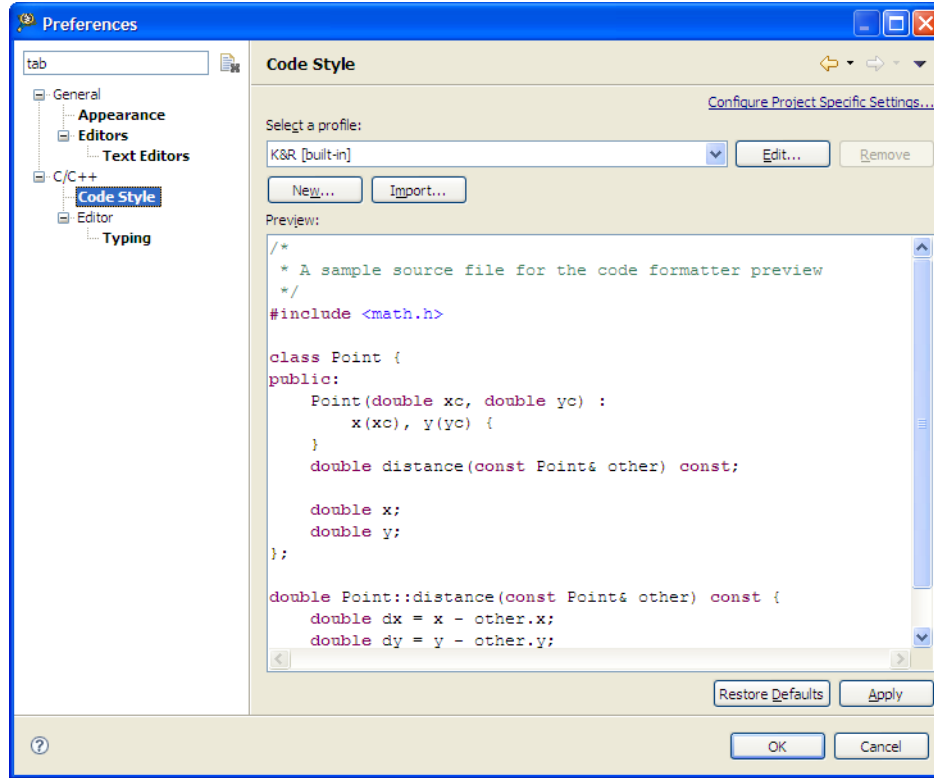


Figure 2-17. Code Style Preference Page

4. Select a predefined profile from the **Select a profile** drop-down list. Each of the profile specify different ways of formatting the code.

If the predefined profiles does not suit your requirements, you can create a new profile.

- a. Click the **New** button.

The **New Code Formatter Profile** dialog box appears.

- b. Enter the name of the new profile in the **Profile name** text box.
- c. Select the profile based on which you want to create the new profile from the **Initialize settings with the following profile** drop-down list.
- d. Click **OK**.

The edit dialog box appears.

- e. Specify the code formatting settings as required and click **OK**.

5. Click **OK** to close the **Preferences** dialog box.

The editor will now use the selected profile to format the code.

2.2.10 How can I quickly switch to a header file from within the source code file?

If you are working in an implementation (*.c/*.cpp) file, and you quickly want to open the corresponding header file, perform either of the following:

- press **CTRL+`** or
- right-click and select **Toggle Source/Header** .

Similarly, you can also toggle back to the source file from the header file.

2.2.11 How does the CodeWarrior IDE differentiate between enabled and disabled macros?

CodeWarrior IDE automatically marks the disabled macros in gray to help you determine which macros are active and which not.

```
58  
59#if PL_KBD_PULLUP_RESISTORS /* then check on bit cleared */  
60 #define KBD_TEST_MACRO(keys,switch) (!testReg8Bits(keys, switch))  
61#else  
62 #define KBD_TEST_MACRO(keys,switch) (testReg8Bits(keys, switch))  
63#endif  
64
```

Figure 2-18. Disabled Macros Marked in Gray

2.2.12 How can I configure predefined macros in the CodeWarrior IDE?

To configure predefined macros in the CodeWarrior IDE:

1. Select the project for which you want to configure the predefined macros in the **CodeWarrior Projects** view.
2. Select **Project > Properties** in the IDE menu bar.

The **Properties** dialog box appears.

3. Select **C/C++ General > Paths and Symbols** .

The **Paths and Symbols** properties page opens in the right panel of the **Properties** dialog box.

4. Select the **Symbols** tab.
5. Configure the symbols listed in the **Symbols** tab page.
6. Click **OK**.

The predefined macros are configured for the selected project.

2.2.13 Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE?

Yes, if you hover mouse cursor over a complex macro, the CodeWarrior IDE shows the evaluated expression (refer to the image listed below).

```

59 #if PL_KBD_PULLUP_RESISTORS /* then check on bit cleared */
60 #define ST_MACRO(keys, switch) (!testReg8Bits(keys, switch))
61 #else (3 != 5)
62 #define ST_MACRO(keys, switch) (testReg8Bits(keys, switch))
63 #endif
    
```

Figure 2-19. Pop-up Window Displaying Macro Expansion

Press F2 to bring in focus the macro expansion (refer to the image listed below).

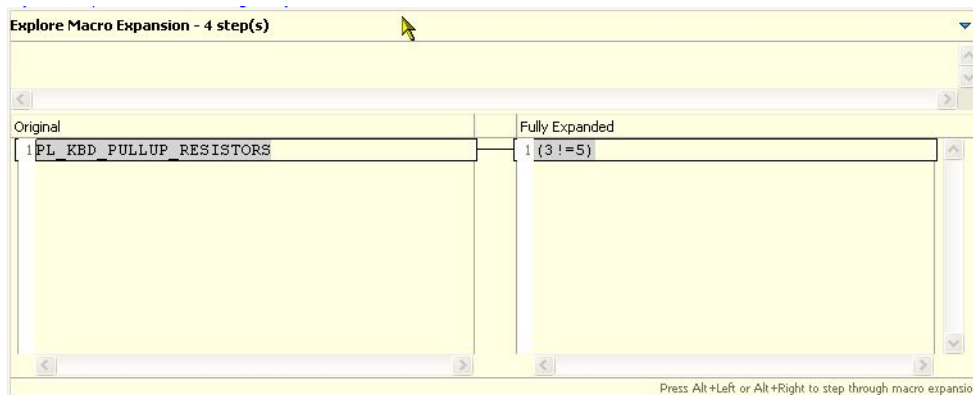


Figure 2-20. Focus over Macro Expansion

Using **Alt+Left** and **Alt+Right** you can step through the macro expansion. This gives you details of the steps in the macro expansion (refer to the image listed below).

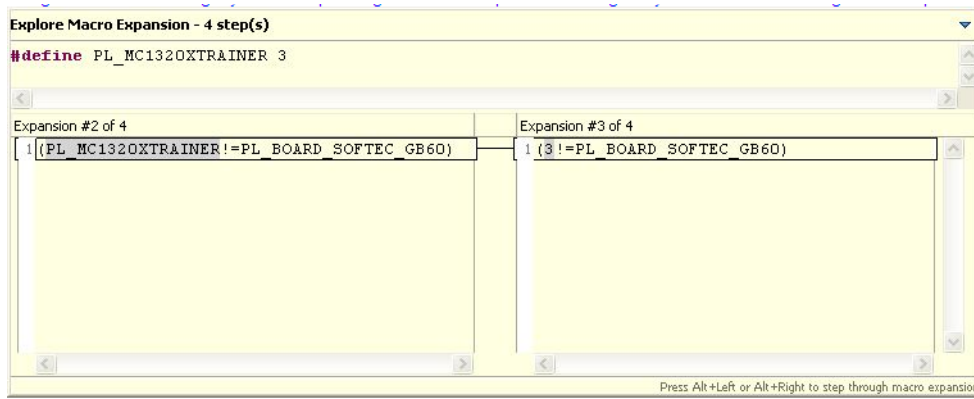


Figure 2-21. Macro Expansion Steps

2.2.14 Is it possible to roll back the changes I did to my source code?

Yes, local history can help roll back the changes you make to your source code.

To roll back the changes, perform these steps.

1. Right-click the updated source code in the editor.
2. From the context menu, select the **Replace with > Local History** command.

The **Compare** dialog box appears. The dialog box lists the date and time of the changes you saved.

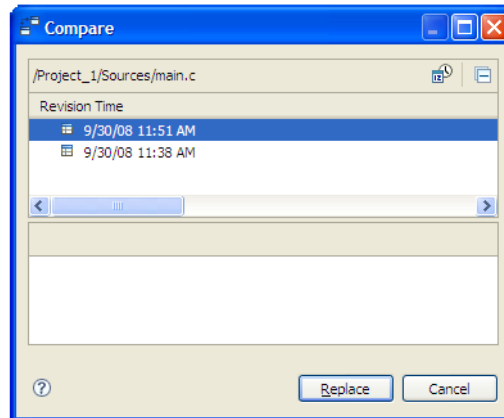


Figure 2-22. Compare Dialog Box

3. From the **Revision Time** options, select the desired date and time.
4. Click the **Replace** button.

The changes made to the file will be roll backed to the selected date and time.

NOTE

The local history is maintained for files only and not for projects or folders.

2.2.15 Is there a way to change the default color of comments that span across multiple lines?

Yes, in order to change the default color of comments that span across multiple lines in source code follow steps below:

1. From C/C++ perspective toolbar, select **Preferences**.

The **Preferences** window appears.

2. Select **C/C++ > Editor > Syntax Coloring**.

The **Syntax Coloring** page appears in the right panel (refer to the image listed below).

3. Select **Comments > Multi-line comment**.

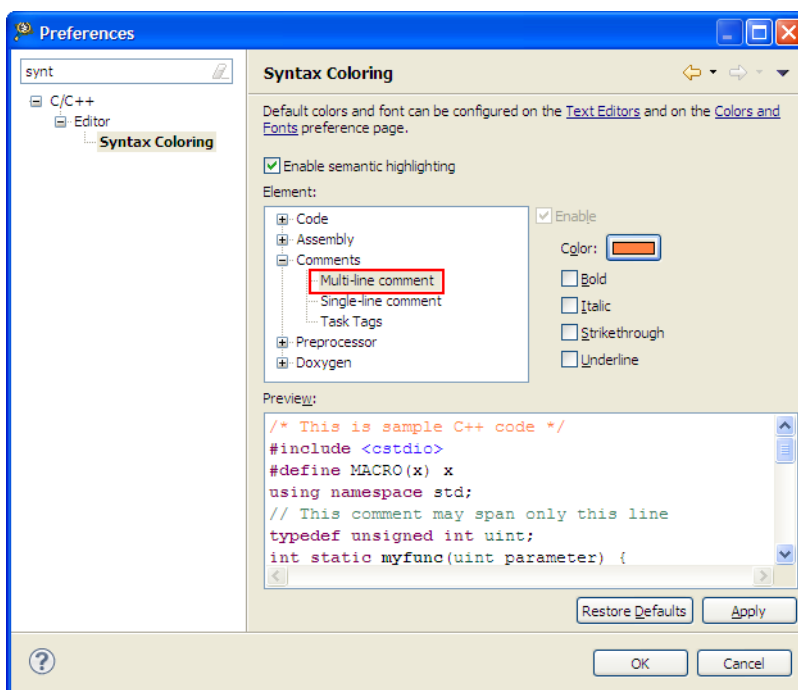


Figure 2-23. Preferences Dialog Box - Syntax Coloring

4. Select **Color**.

The color palette window appears.

5. From Color Palette, select a color of your choice.
6. Click **OK**.
7. Click **Apply**.
8. Click **OK**.

NOTE

If you want to change the color of a comment that is across a single line, then follow the steps given above, except selecting **Single-line comment** instead of **Multi-line comment**.

The color of the comments in the source code changes, as specified.

2.2.16 Is there a way to revert to the original contents of a source code file?

Yes, if undesired changes were made to source file, there is a way to revert to the previous file. In order to do this follow steps below:

1. Right-click the source file and select **Replace With > Local History** from the context menu.

The **Compare** view appears.

2. Double-click **Revision Time** tab.

The **Compare** window appears.

3. Select **Replace** .

The file is replaced.

NOTE

Another way to replace the file is to select **Replace With > Previous from Local History** from context menu and this command replaces the file with the last file that you saved.

2.2.17 How do I convert Line Delimiters to the Unix format?

The Default is Windows. To change to UNIX, select **File > Convert Line Delimiters To > Unix** from the IDE menu bar.

2.2.18 Why do I get the following error message when I try to edit a source code file?

```
File '../..../filename.c' is read-only. Do you wish to make it writable?
```

You get this error message because the file that you are trying to edit is read-only. If you do not want to make it writable then select **No** . If you want to find out the settings for this file, follow steps below:

1. From the CodeWarrior Projects window, right-click on the source file.

A context menu appears.

2. From context menu, select **Properties** .

The **Properties** window appears.

3. Select **Resource** .

On the right-hand side, the settings for the file appear.

2.2.19 What does an asterisk on the Editor title bar mean?

The asterisk on the editor's toolbar indicates there are unsaved changes.

2.2.20 How can I find out where a function is declared in my source code?

To find out where a function is declared in source file, follow steps below:

1. Double-click on source file.

Source file appears in the **Editor** view.

2. In Source file, right-click the function name.

A context menu appears.

3. From context menu, select **Open Declaration** .

The source file that contains the function declaration appears.

2.2.21 How can I open .tcl extension files in the Editor window?

In order to open .tcl extension files in the Editor window, a File Association needs to be created. To create a File Association, follow steps below:

1. From the CodeWarrior toolbar, select **Window > Preferences**.

The **Preferences** dialog box appears.

2. Select **General > Editors > File Associations**.

The **File Associations** preferences appear on the right hand side.

3. Select **File Associations**.

The **File Associations** pane appears.

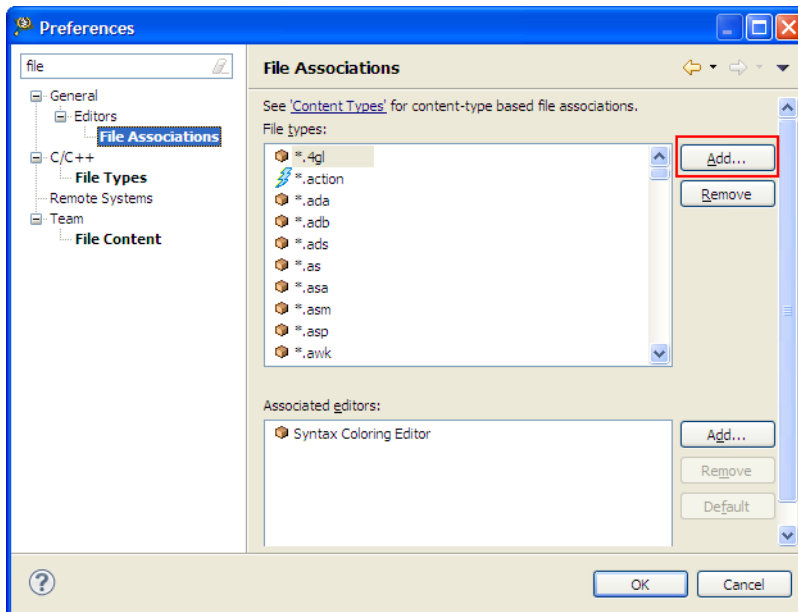


Figure 2-24. Preferences Dialog Box - File Association

4. Click **Add**.

The **New File Type** dialog box appears.

5. In the **File Type** textbox, enter `.tcl`.

6. Click **OK**.

The `.tcl` extension appears under **File Types** .

7. Under **Associated editors** pane, Click **Add**.

The **Editor Selection** dialog box appears.

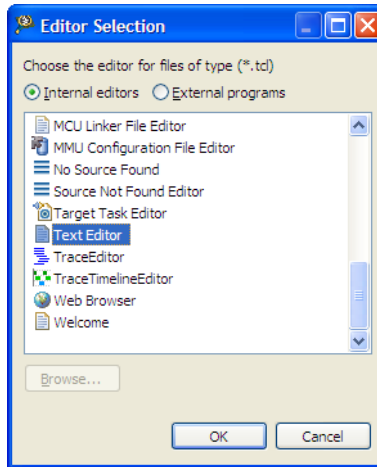


Figure 2-25. Editor Selection Dialog Box

8. Click **OK**.

The `.tcl` extension file types get associated with a Text Editor (as the following figure shows).

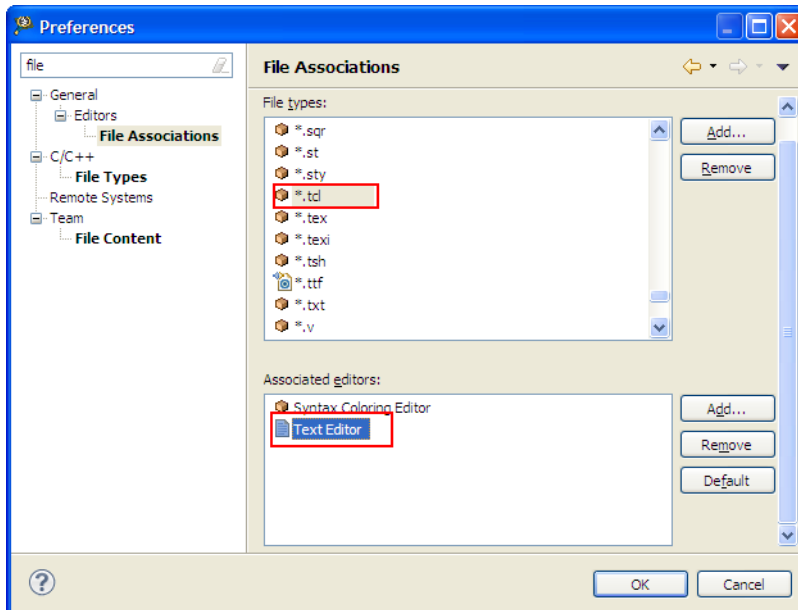


Figure 2-26. Preferences Dialog Box - File Association

9. Click **OK**.

You can now open the `tcl` extension files in the Editor window.

2.2.22 Is it possible to condense a large source file by folding/unfolding different portions of code?

Yes, you can enable folding in editor and specify levels of folding and which portion of code needs to be folded. To enable and configure folding/unfolding, perform these steps.

1. From the IDE menu bar, select **Window > Preferences**.

The **Preferences** dialog box appears.

2. Type folding as the filter text to narrow down the list of preferences.
3. Select **C/C++ > Editor > Folding**. The **Folding** preference page appears in the right panel of the **Preferences** dialog box.

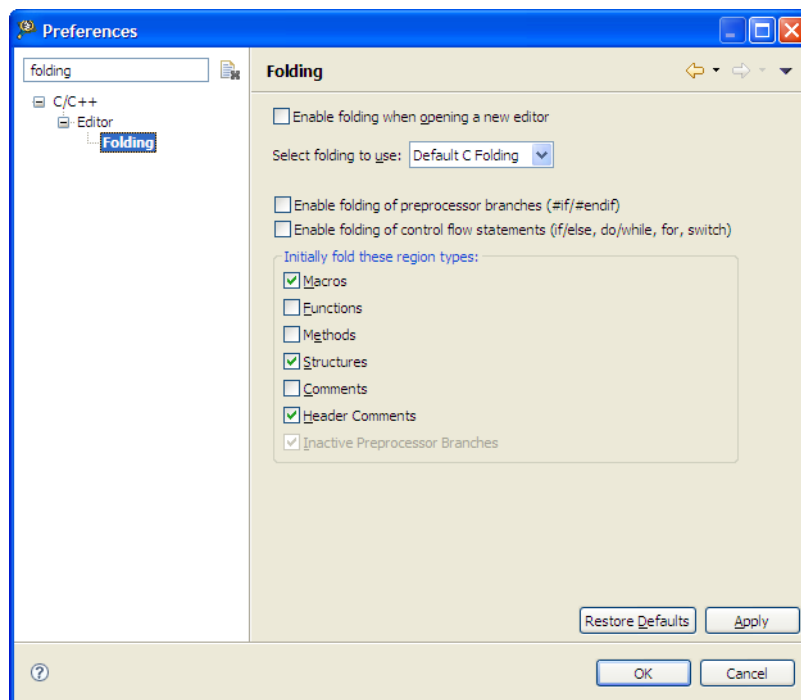


Figure 2-27. Folding Preference Page

4. Check the **Enable folding when opening a new editor** checkbox.
5. Configure level of folding and select which regions of code should be automatically folded.
6. Click **OK** and reopen your source file.

Now you can fold/unfold portions in your source code, as the figure listed below shows. The fold/unfold functionality is added with new markers (+/-) at the left side in the source file. This improves readability of the source code.


```

main.c X
#include <hidef.h> /* for EnableInterrupts macro */
#include "derivative.h" /* include peripheral declarations */

void InitSystems() {

    SOPT = 0x10;           // Disable COP,RSTO
    SPMSC1 = 0x00;        // Disable LVD
    MCGTRM = NVMCGTRM;    // Load Trim Values from flash
    MCGSC_FTRIM = NVFTRIM_FTRIM; // Load Fine Trim Value from fl
} // end InitSystems

// Initialize the FTM1CH3 for Output Compare (LED1)
void FTM1C3_Output_Compare() {
    FTM1SC_CPWMS = 0;    // Edge aligned
    FTM1C3SC_CH3IE = 1; // Enable channel 3 interrupt
    FTM1C3SC_MS3x = 1;  // Output compare
    FTM1C3SC_ELS3x = 1; // toggle output on compare
    FTM1C3CV = 0x8000;  // Interrupt after 32,768 decimal cycles
                        // At 32.768Khz, 32,768 cycles yields in
    FTM1SC_PS = 7;      // 4.194304Mhz/128 = 32.768Khz
    FTM1SC_CLKSx = 1;   // bus clock and start the timer
} //end FTM1C3_Output_Compare

// Initialize the FTM1CH2 for Output Compare (LED2)
void FTM1C2_Output_Compare() {
    FTM1SC_CPWMS = 0;    // Edge aligned
    FTM1C2SC_CH2IE = 1; // Enable channel 2 interrupt
}
    
```

Figure 2-28. Editor With Folding Enabled

To preview what is inside a folded region, you do not need to unfold it again. Hover your mouse cursor over a folded region and a popup appears displaying the details.

```

void InitSystems() {
    void InitSystems() {
        SOPT = 0x10;           // Disable COP,RS
        // In
        SPMSC1 = 0x00;        // Disable LVD
        void
        MCGTRM = NVMCGTRM;    // Load Trim Valu
        MCGSC_FTRIM = NVFTRIM_FTRIM; // Load Fine Trim
        // In
        } // end InitSystems
    void
        FTM1SC_CPWMS = 0;    // Edge aligned
    }
    
```

Figure 2-29. Folded Region Preview

2.2.23 Is there a way to determine in which workspace I am working currently?

Yes, there is a command line switch to help you determine the location of the workspace your CodeWarrior IDE is using currently. You may need this information if you have multiple instances of CodeWarrior IDE open and if you switch workspaces frequently.

Add the `-showlocation` command line shortcut to the properties of the CodeWarrior IDE shortcut, as the following figure shows.

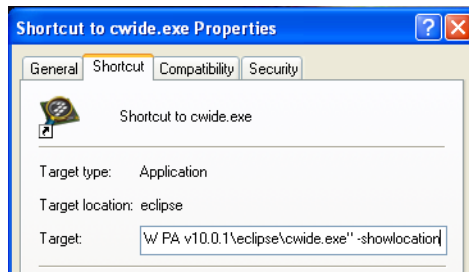


Figure 2-30. Determine Workspace Location

The current workspace location gets displayed on the title bar of your CodeWarrior IDE.

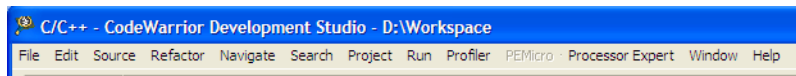


Figure 2-31. Current Workspace Location

2.2.24 How can I share same installation of the CodeWarrior software with multiple users?

By default, CodeWarrior IDE opens with the previously used workspace. However, you can configure the CodeWarrior IDE to open with the user-specific workspace, that is depending upon the user logged on to the system.

You can specify workspace path by passing the following command switch to the CodeWarrior IDE on startup.

```
-data workspace path
```

If you want to open workspace folder per user basis, pass the following command line switch to the CodeWarrior IDE.

```
-data "%USERPROFILE%\perUser_workspace
```

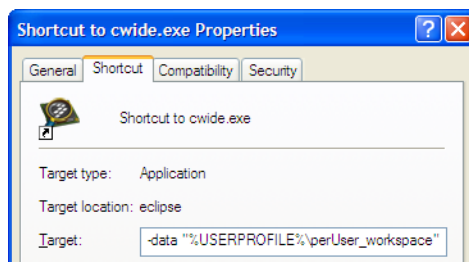


Figure 2-32. Specify User-Specific Workspace

2.2.25 How can I comment/uncomment a block of code in the editor?

In ANSI C, you use `/*` and `*/` for commenting a code line. For example:

```
/* this is a comment */
```

You can also span a comment over multiple code lines.

```
/*  
do {  
    EVNT1_HandleEvent();  
    cnt--;  
} while (EVNT1_EventsPending() && cnt>0);  
*/
```

However, you cannot nest comments with `/* */` comments. CodeWarrior IDE marks it with a ``?'` symbol.

```
75/*  i++; /* comment */  */
```

In addition, most compilers, if not put into the strict ANSI C mode, also allow the usage of the `//` characters for commenting a code line.

```
// this is a comment too
```

This also allows you to comment out a line with comments.

```
// Function(); /* this is a call to my function*/
```

However, if you want to comment a whole code block, for example, the following code lines, commenting may not work for all the code lines in the block.

```
do {  
    EVNT1_HandleEvent();  
    cnt--; /* limit number of iterations */  
} while (EVNT1_EventsPending() && cnt>0);
```

One way to comment a code block is to use `#if 0 ... #endif`.

```
#if 0  
do {  
    EVNT1_HandleEvent();  
    cnt--; /* limit number of iterations */  
} while (EVNT1_EventsPending() && cnt>0);  
#endif
```

Figure 2-33. Commenting Code Block

The source file parser in CodeWarrior IDE automatically detects the `false` condition and grays out the code for better readability.

Another way is to comment/uncomment is to select the code lines you want to comment/uncomment and right-click. The following figure shows the context menu that appears.

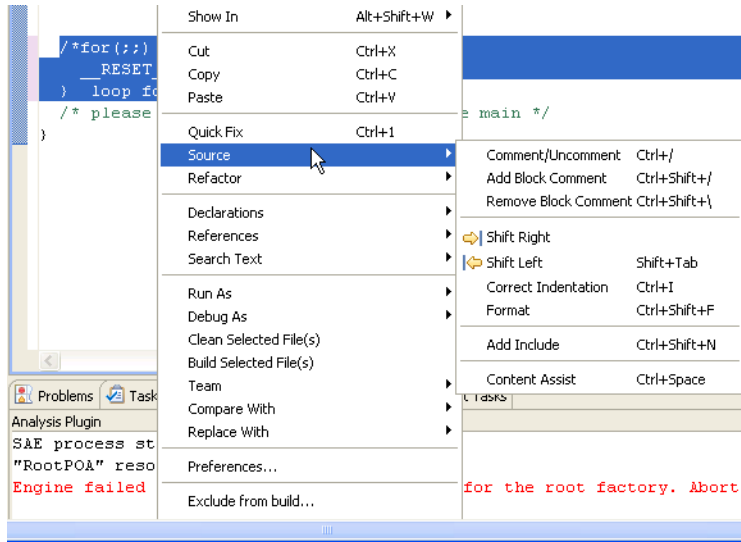


Figure 2-34. Editor Context Menu

Select **Source > Comment/Uncomment** from the editor context menu. The selected code lines are commented using the `//` characters.

```

cnt = 16;
//do {
//EVNT1_HandleEvent();
//cnt--; /* limit number of iterations */
//} while (EVNT1_EventsPending() && cnt>0);
} else {

```

Similarly, you can also uncomment the code block.

To comment code block using the `/* */` characters and uncomment, use **Add Block Comment** and **Remove Block Comment** in the editor context menu.

```

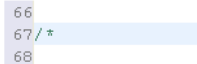
cnt = 16;
/*
do {
EVNT1_HandleEvent();
cnt--; limit number of iterations
} while (EVNT1_EventsPending() && cnt>0);

```

```
*/  
} else {
```

But note that, nested `/* */` comments are not allowed. The CodeWarrior IDE removes the nested comment tokens, as shown in the example above. You need to check the comments in the code block after applying comment using the `/* */` characters.

In addition, CodeWarrior IDE editors include comment auto-templates. For example, if you type `/*` in the editor and press enter, editor automatically creates a comment block for you.



```
66  
67 /*  
68
```

Figure 2-35. Comment Block



```
66  
67 /*  
68 *  
69 */
```

You can immediately start typing your comment text inside the comment block.

To configure comment template settings:

1. Select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Type "comment" as the filter text to narrow down the list of the preferences.
3. Select **Code Style > Code Templates**.
4. Make the required settings in the **Code Template** preference panel and click **OK** .

The template settings are configured, as specified.

2.2.26 How can I compare files in CodeWarrior IDE?

To compare files:

1. Hold the CTRL key and select the files you want to compare.
2. Right-click and select **Compare With > Each Other**.

The **Compare** view appears.

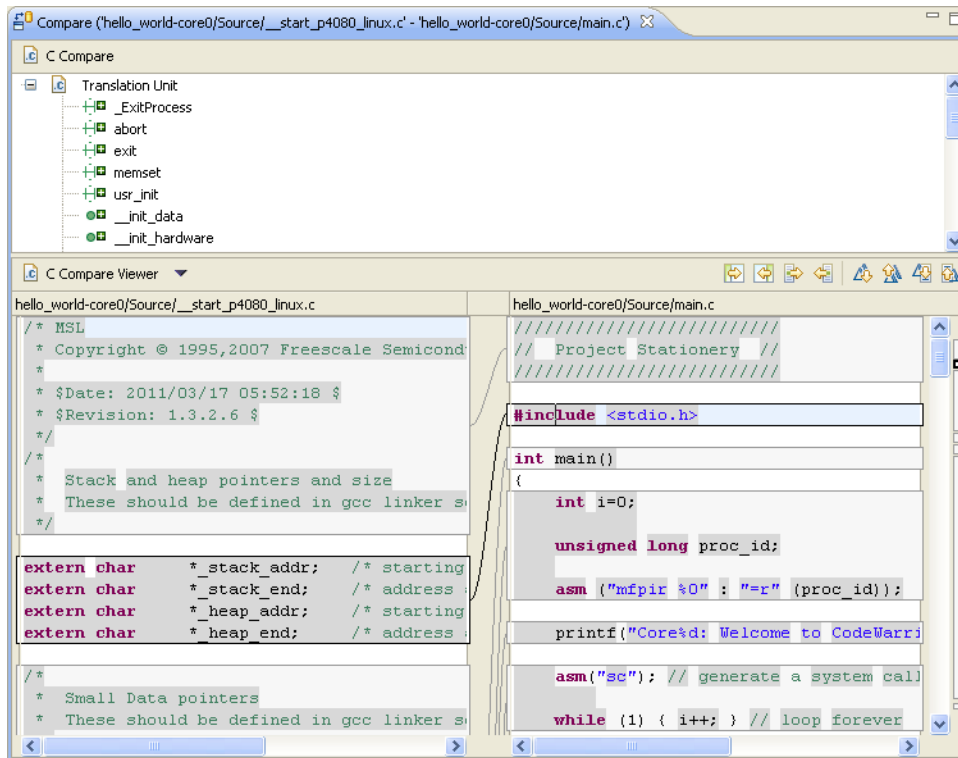


Figure 2-36. Compare View

The **C Compare** area displays the list of items added or removed from the files. The items are marked with specific addition and deletion icons. Double-click an item to navigate to it in the files.

Using the toolbar, you can:

- Copy all changes, or only the not conflicting ones
- Copy the current change to left or right
- Go to the next or previous change or difference

Similarly, you can also compare more than two files in CodeWarrior IDE. To compare more than two files, you need to specify which file should be used as the common ancestor.

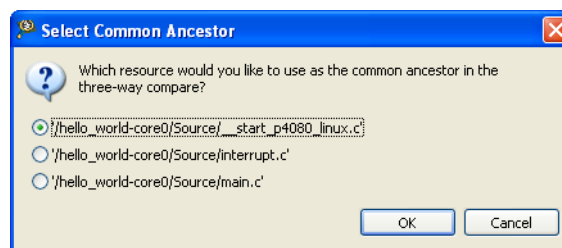


Figure 2-37. Specify Common Ancestor File

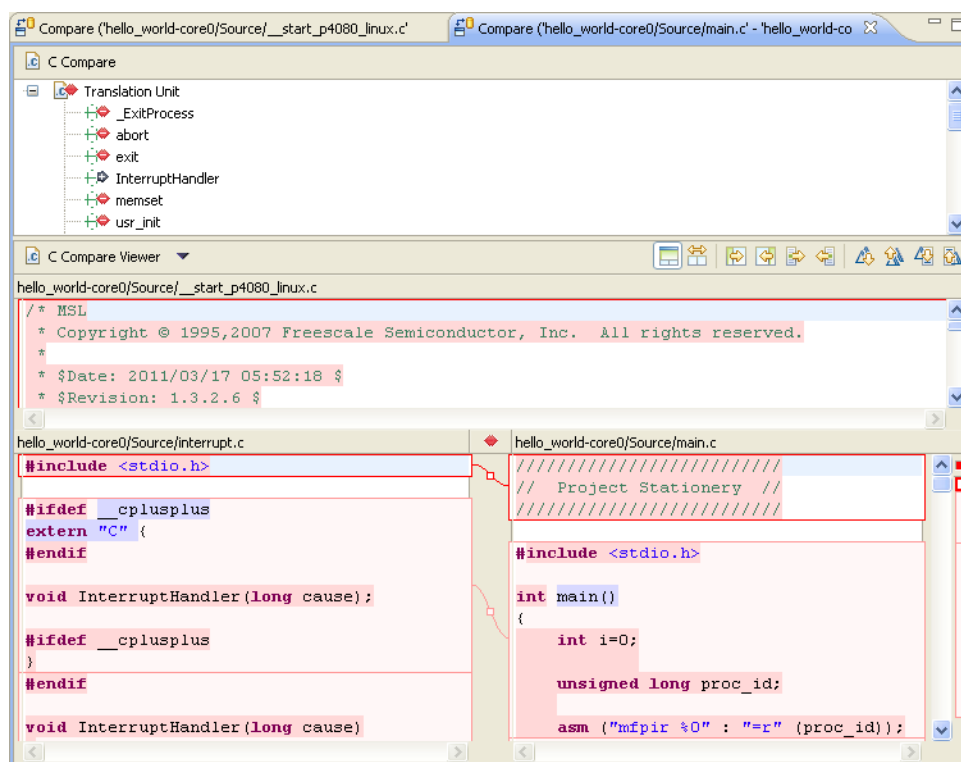


Figure 2-38. Compare More Than Two Files

Using the **Show Ancestor Pane** and **Two-Way Compare (Ignore Ancestor)** toolbar buttons, you can display or close the **Ancestor** view.

2.2.27 Is there a way to compare files that are not in project?

To compare files that are not in a project, you need to use an external diff tool, for example KDiff, which is an open source tool.

1. Download and install KDiff from the following location:

<http://kdiff3.sourceforge.net/>

2. Download the plugin file from the following location:

<http://externaldiff.sourceforge.net/>

3. Select **Window > Preferences** to configure the tool.
4. Type `compare' as the filter text and select **Compare/External Tool** panel.
5. Browse to the location where KDiff is installed and select **kdiff.exe** .
6. Click **OK** .
7. Select the files you want to compare using KDiff.
8. Right click and select **Compare With > Each Other (external tool)** .

The result of comparison appears in the KDiff editor.

2.2.28 How do I switch between the source files in the Editor view?

To switch between the source files in the Editor view while creating your project:

1. Press **Ctrl+Tab** .

A small pop-up window appears listing all open source files in the Editor view.

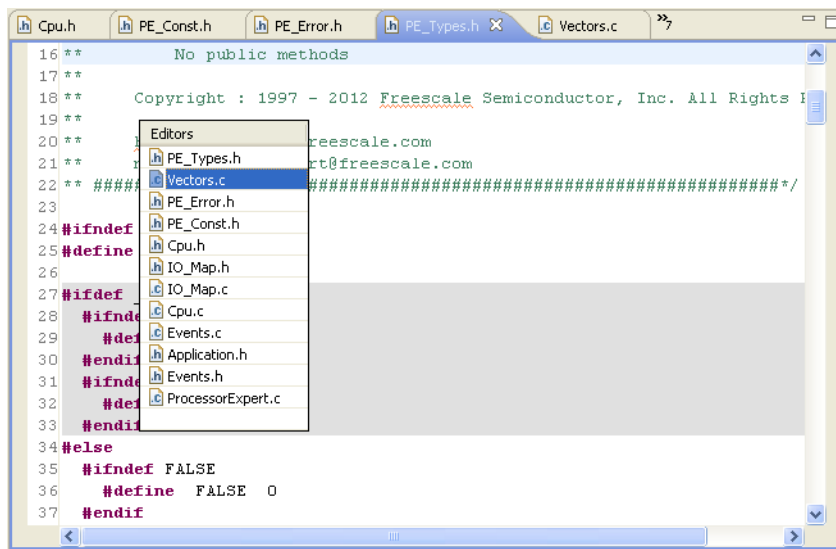


Figure 2-39. Switching Between Source Files in Editor View

2. Press **Ctrl+Tab** repeatedly to iterate through the source files list.

By using this shortcut, you can quickly swap between different source files.

2.2.29 I want to edit my source files, but all the other views are using the space as well. How do I maximize the editor view to have better working area?

To maximize the views in the CodeWarrior IDE, just double-click on the view tab to toggle between maximized view and the normal view. Alternatively, right-click on the view tab and select **Maximize** from the context-menu to maximize the view. Similarly, right-click on the maximized view tab select **Minimize** from the context-menu to return to the normal view.

2.2.30 How do I remove blank lines in the Editor view?

You can remove the empty or blank lines in the Editor view by following any of these methods:

- By using the keyboard shortcut

Place the cursor on a line and press **Ctrl+D** (for line Delete). You can also use **Ctrl+D** to delete multiple selected line.

Tip

To get more commands for *deleting* the blank lines, press **Ctrl+3**, enter *delete* in the textbox, the related commands with their keyboard shortcut appears in the **Commands** list.

- By using the search and replace method
 - a. Select **Edit > Find/Replace** from IDE menu bar, alternatively press **Ctrl+F**.

The **Find/Replace** dialog box appears.

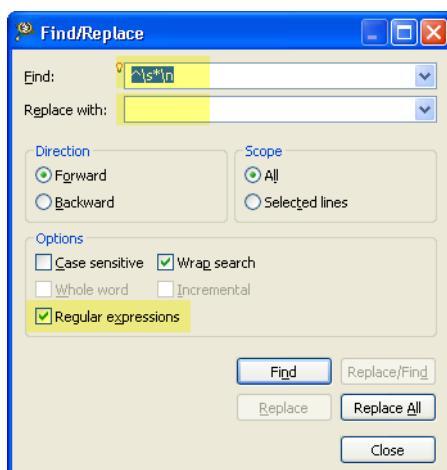


Figure 2-40. Find/Replace Dialog Box

- b. Enter **^\s*\$\n** in the **Find** textbox and leave the **Replace with** textbox empty.
- c. Check the **Regular expressions** checkbox.
- d. Click **Replace All**.

The editor removes the blank lines.

2.3 Workbench Window

In this topic, Workbench window related FAQs are listed.

- [What controls the initial layout of a view in a Workbench window?](#)
- [Is it possible to restrict the number of resources that appear on my Workbench window?](#)
- [How do I minimize a Workbench window?](#)
- **NEW** [How do I customize my Workspace perspectives?](#)
- **NEW** [How do I enable full screen mode in my CodeWarrior IDE?](#)
- **NEW** [How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with?](#)
- **NEW** [How can I view all keyboard shortcuts available in the CodeWarrior IDE?](#)
- **NEW** [How do I edit the keyboard shortcuts in the CodeWarrior IDE?](#)

2.3.1 What controls the initial layout of a view in a Workbench window?

A perspective defines the initial layout of the views in a workbench window. One workbench window contains many perspectives. Each of the perspectives is task oriented.

2.3.2 Is it possible to restrict the number of resources that appear on my Workbench window?

Yes, working sets help reduce the number of resources displayed. A working set is a group of elements you want to display.

To create or define a new working set, perform the following steps.

1. Click  in the **CodeWarrior Projects** view or **Project Explorer** view toolbar.

The pull-down menu appears.

2. Select the **Select Working Set** command.

The **Select Working Set** dialog box appears.

3. Click the **New** button.

The **New Working Set** wizard appears.

4. From the **Working set type** options, select the appropriate working set. For example, select `c/c++`.

5. Click **Next**.

The **<selected> Working Set** page appears. For example, C/C++ Working Set.

6. In the **Working set name** field, enter an appropriate name for the working set. For example, Demo_Set.

7. In the **Working set content** tree, check the items you want in your view. For example, check Project_1 and Project_3 (refer to the image listed below).

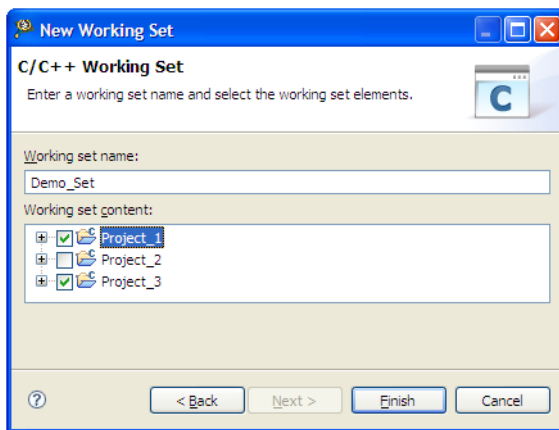


Figure 2-41. New Working Set Wizard - C/C++ Working Set Page

8. Click **Finish** to close the **New Working Set** wizard.

9. The **Select Working Set** dialog box reappears with the newly created working set.

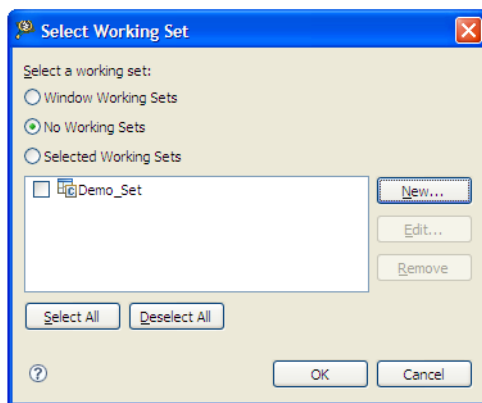


Figure 2-42. Select Working Set Dialog Box

10. Check the checkbox corresponding to the desired working set and click **OK**.

The **CodeWarrior Projects** view or **Project Explorer** view will now display the selected resources only.

NOTE

To deselect an active working set, select the **Deselect Working Set** command from the view pull-down menu.

NOTE

To edit an active working set, select the **Edit Active Working Set** command from the view pull-down menu.

2.3.3 How do I minimize a Workbench window?

When initiated for the first time, the Workbench window tends to appear in a peculiar state. The window seems to stretch to fit the entire screen but might not be maximized. This generally happens, because the window takes the dimensions of the last application window you opened. To minimize such a window, ensure that the **Maximize** button indicates an unmaximized window. Next, drag the window edge (the cursor will appear as a left-right arrow \leftrightarrow) to the desired width.

2.3.4 How do I customize my Workspace perspectives?

To customize your Workspace perspectives:

1. Create a new workspace. This way you have a clean start for customization and no changes from an existing modified workspace.
2. Close the views that you do not want to use for your project manually.
3. For customizing the buttons and menus visibility in the desired perspective:
 - a. Select **Window > Customize Perspective** from the IDE menu bar.

The **Customize Perspective - <perspective name>** dialog box appears.

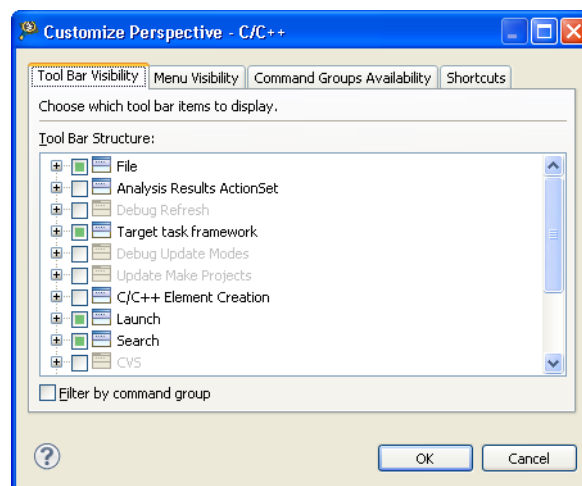


Figure 2-43. Customize Perspective - <perspective name> Dialog Box

- b. Click the desired tab to select the desired settings for tool bar, menu, command groups and shortcuts visibility.
- c. Click **OK**.

The **Customize Perspective - <perspective name>** dialog box closes.

4. To save the changes you have made to the IDE's perspective, select **Window > Save Perspective As** from the IDE menu bar.

The **Save Perspective As** dialog box appears.

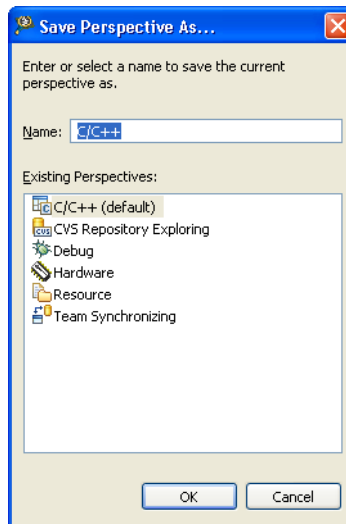


Figure 2-44. Save Perspective As Dialog Box

5. Select **OK**.
6. The **Overwrite Perspective** dialog box appears.
7. Click **Yes** to overwrite the previous default setting with the changes you made.

2.3.5 How do I enable full screen mode in my CodeWarrior IDE?

To change your CodeWarrior IDE to full screen mode:

1. The Eclipse full screen plugin is available at <http://code.google.com/p/eclipse-fullscreen/> web address.
2. Download the zip file.
3. Extract the `cn.pande.eclipsex.fullscreen_<version>.jar` file from the zip file you have downloaded.
4. Copy the extracted jar file in the `<CWInstallDir>\CW MCU v10.x\eclipse\plugins` folder, where `<CWInstallDir>` is the installation directory of your CodeWarrior software.

The full screen mode option is now available in your CodeWarrior Eclipse IDE.

To use the full screen mode feature:

1. Open the CodeWarrior IDE.
2. Select **Window > Full Screen** from the IDE menu bar, alternatively, press **Ctrl+Alt+Z** to switch to the full screen mode.

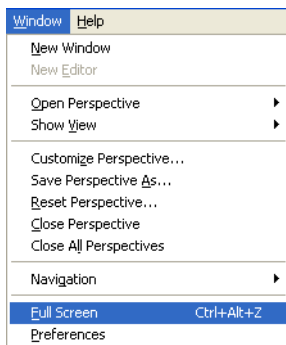


Figure 2-45. Full Screen Menu Option

The CodeWarrior Eclipse IDE switches to full screen mode, hiding the toolbar, menu bar and status bar from the screen.

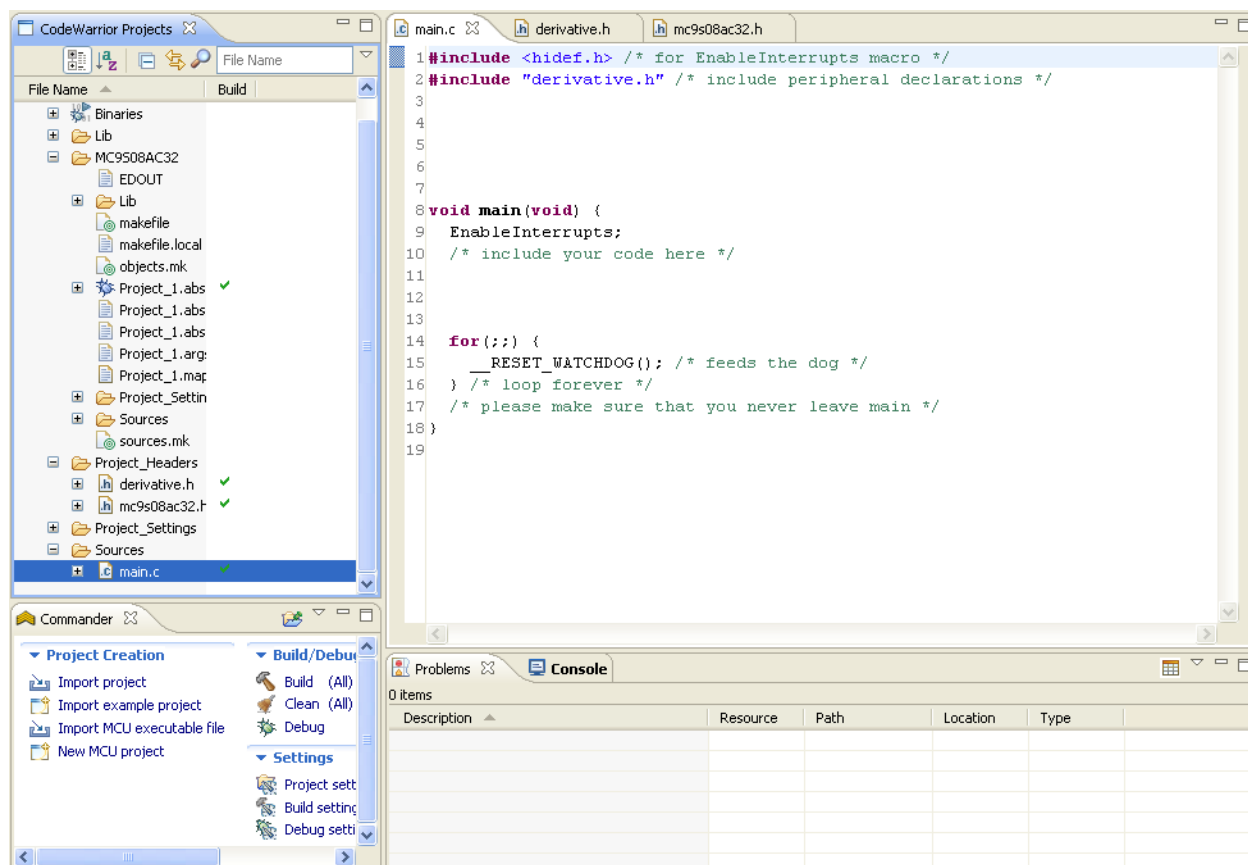


Figure 2-46. CodeWarrior Eclipse IDE - Full Screen Mode

3. Press **Esc** , or **Ctrl+Alt+Z** to exit the full screen mode.

The CodeWarrior Eclipse IDE switches back to the normal screen mode.

NOTE

The **Ctrl+Alt+Z** is the default shortcut to toggle between *normal* and the *full screen* mode for the CodeWarrior Eclipse IDE.

2.3.6 How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with?

To hide the toolbar from the IDE, right-click on the toolbar and select **Hide Toolbar** from the context-menu. The following figure shows the IDE toolbar context-menu:

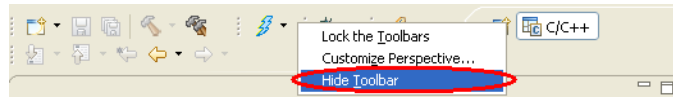


Figure 2-47. Hide Toolbar - Toolbar Context-menu Option

To display the toolbar again in the IDE, select **Window > Show Toolbar** from the IDE menu bar. The following figure shows the **Show Toolbar** menu option:

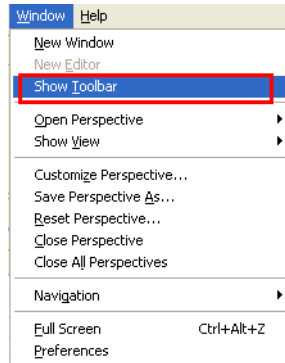


Figure 2-48. Show Toolbar - Menu Option

2.3.7 How can I view all keyboard shortcuts available in the CodeWarrior IDE?

To view all keyboard shortcuts available in the CodeWarrior IDE:

1. Open the CodeWarrior IDE.
2. Press **Ctrl+Shift+L** .

The drop-down list containing the commands and their keyboard shortcuts appears in the IDE.

Activate Editor	F12
Add Block Comment	Ctrl+Shift+/-
Add Include	Ctrl+Shift+N
Backward History	Alt+Left
Build All	Ctrl+B
Build Project	Alt+B, B
Clean All Projects	Alt+B, L
Close	Ctrl+W
Close All	Ctrl+Shift+W
Collapse	Ctrl+Numpad_Subtract
Collapse All	Ctrl+Shift+Numpad_Divide
Comment/Uncomment	Ctrl+/
Content Assist	Ctrl+Space
Context Information	Ctrl+Shift+Space
Copy	Ctrl+C
Copy Lines	Ctrl+Alt+Down
Cut	Ctrl+X
Debug	Alt+B, D
Delete	Delete
Delete Line	Ctrl+D
Delete Next Word	Ctrl+Delete
Delete Previous Word	Ctrl+Backspace
Delete to End of Line	Ctrl+Shift+Delete
Duplicate Lines	Ctrl+Alt+Up
Expand	Ctrl+Numpad_Add
Expand All	Ctrl+Numpad_Multiply
Explore Macro Expansion	Ctrl+=

Press "Ctrl+Shift+L" to open the preference page.

Figure 2-49. Keyboard Shortcuts Drop-down List

2.3.8 How do I edit the keyboard shortcuts in the CodeWarrior IDE?

To edit the keyboard shortcuts available in the CodeWarrior IDE:

1. Press **Ctrl+Shift+L** twice.
2. The **Preferences** dialog box with the **Keys** page appears.

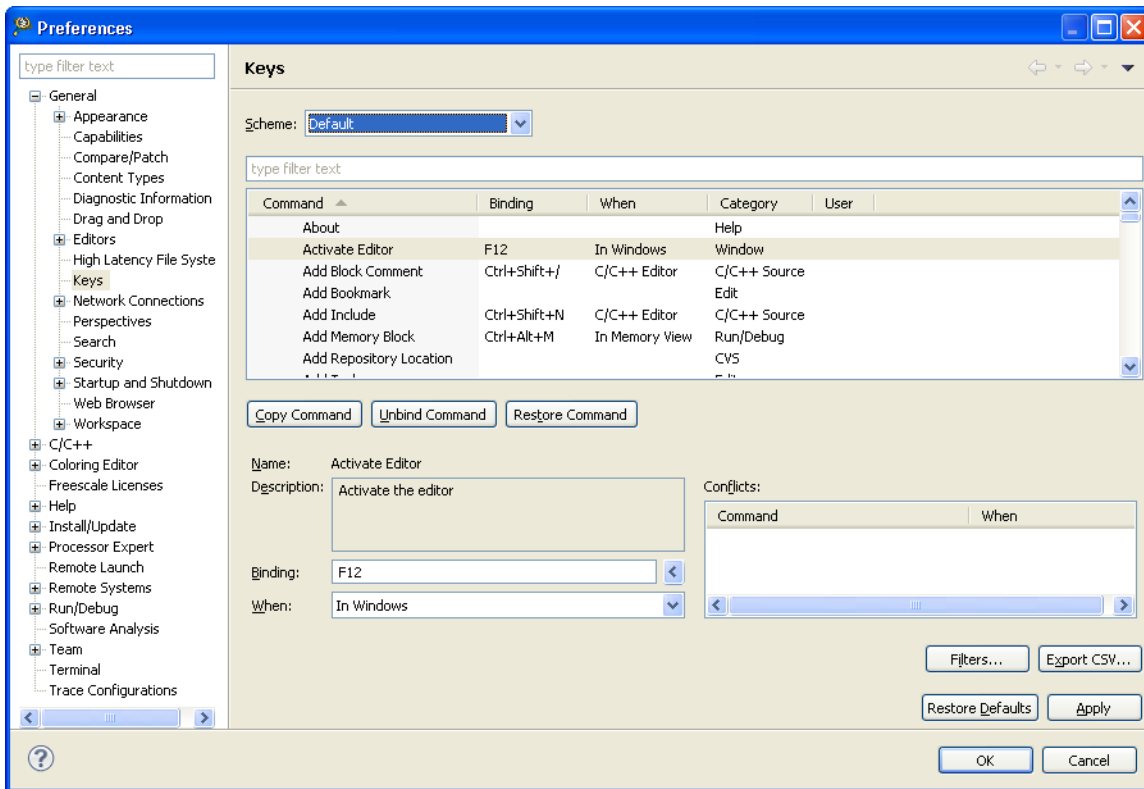


Figure 2-50. Preferences Dialog Box - Keys Page

3. Select the desired command.
4. Click the **Copy Command** , **Unbind Command** , and **Restore Command** buttons to copy, unbind or restore the selected command respectively.
5. Click **Apply** .
6. Click **Ok** .

The IDE saves and applies the changes you made.

2.4 CVS

In this topic, CVS related FAQs are listed.

- [Does the CodeWarrior IDE support version control systems?](#)
- [How can I add my project to CVS from within the CodeWarrior IDE?](#)

2.4.1 Does the CodeWarrior IDE support version control systems?

Yes, CodeWarrior IDE does provide support for the version control systems. One of the version control system supported by the CodeWarrior IDE is CVS.

To use CVS with the CodeWarrior IDE:

1. From the IDE menu bar, select **Window > Show View > Other**.

The **Show View** dialog box appears.

2. Expand the **CVS** tree control and select **CVS Repositories**.
3. Click **OK**.

The **CVS Repositories** view appears.

4. Click the **Add CVS Repository** command on the **CVS Repositories** view toolbar.

The **Add CVS Repository** dialog box appears (refer to the image, *Add CVS Repository Dialog Box* listed below).

5. Enter host name and path of the repository that you want to browse through in the **Host** and **Repository path** text boxes respectively.
6. Enter user name and password in the **User** and **Password** text boxes respectively.



Figure 2-51. Add CVS Repository Dialog Box

7. Click **Finish**.

You can browse through the specified repository in the **CVS Repository** view.

8. Right-click the desired folder and select the desired action from the context menu to perform CVS actions, such as **Check Out** (refer to the image listed below).

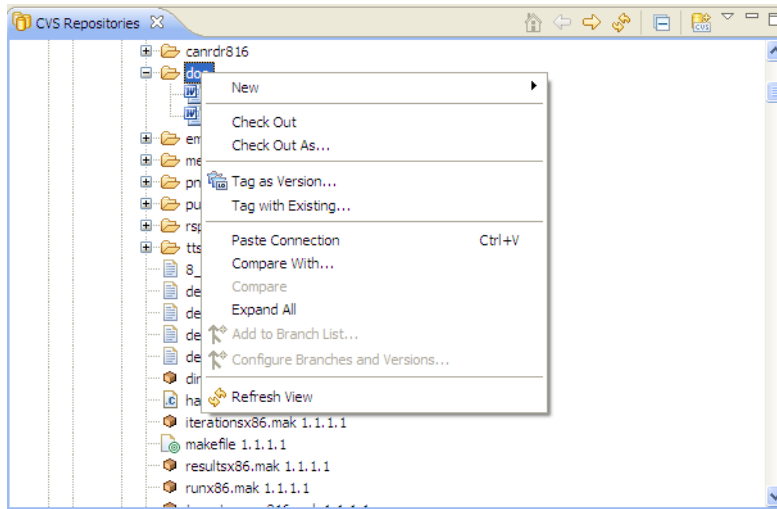


Figure 2-52. CVS Repository View

You can now use CVS with the CodeWarrior IDE.

2.4.2 How can I add my project to CVS from within the CodeWarrior IDE?

To add a project to the CVS:

1. Right-click the project in the **CodeWarrior Projects** view and select **Team > ShareProject**.

If you have already configured a CVS repository with the CodeWarrior IDE, the **Share Projectwith CVS Repository** page of the **Share Project** wizard appears. Otherwise, the **Enter Repository Location Information** page appears.

2. Specify the repository to which you want to add the project and click **Next**.

The **Enter Module Name** page appears.

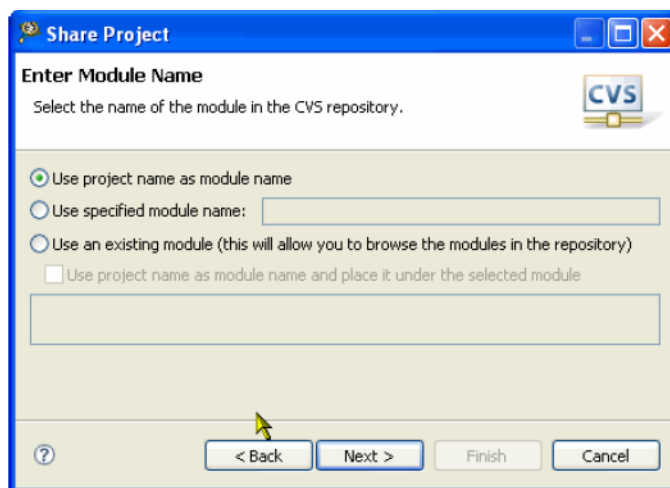


Figure 2-53. Enter Module Name Page

3. Specify name of the module in the CVS repository and click **Next**.

The **Share Project Resources** page appears.

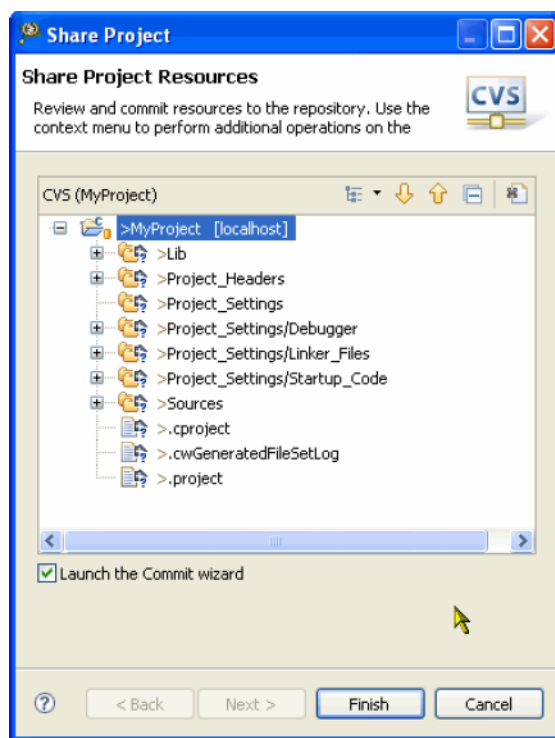


Figure 2-54. Share Project Resources Page

4. Select the files that you want to add to the CVS.

If the project includes generated file, then you may not want to add them to the CVS module. You can add such files to the `.cvsignore` list. The `.cvsignore` is a special text file which specifies all the files that should be ignored/filtered out. The `.cvsignore` file can also use wildcards, such as `*.tmp` for filtering.

5. Check the **Launch the Commit wizard** checkbox and click **Finish**.

The **Commit Files** wizard starts displaying the list of unknown files.

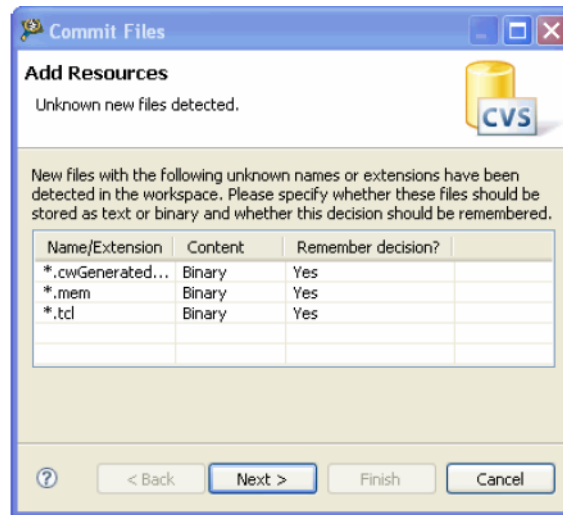


Figure 2-55. Commit Files Wizard

- Specify whether these files are binary or ASCII Text. In this example, CVS does not identify three file extensions and assumes them as binary files. However, as *.mem, *.tcl, and *.cwGeneratedFileSetLog are text files, the file type is changed to ASCII Text.

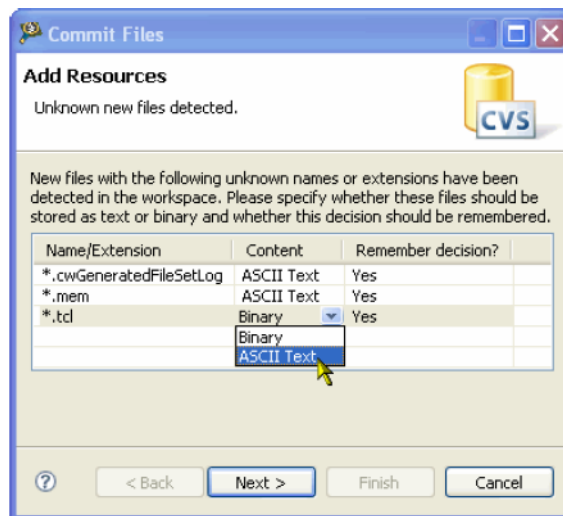


Figure 2-56. Specify How to Store Unknown Files

- Click **Next**.

The **Commit** page appears.

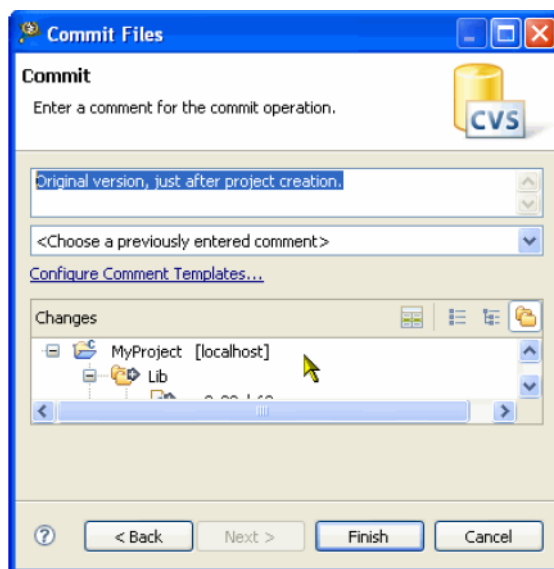


Figure 2-57. Commit Page

8. Enter appropriate comment for the commit action and click **Finish**.

The icons of the files and folders in the project change, which specify that the project is now under CVS control.

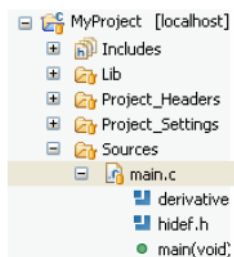


Figure 2-58. Project Under CVS Control

Now, if you have done changes to a source file, and you need to view the differences between your local copy and the repository copy, perform these steps.

1. Right-click the modified file and select **Team > Synchronize with Repository**. The **C Compare Viewer** opens displaying comparison between the files (refer to the image listed below).

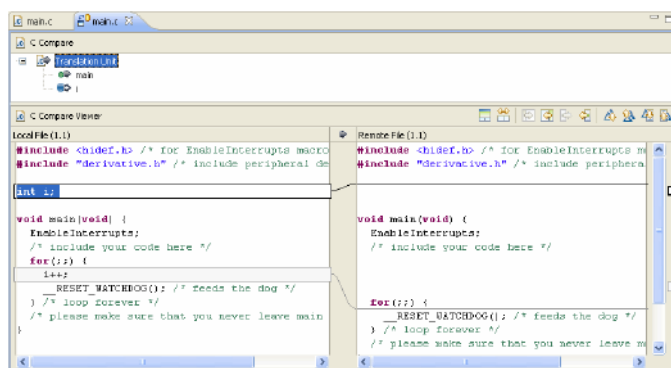


Figure 2-59. C Compare Viewer

2. Review the changes. If you are ready to commit the changes, right-click the file and select **Team > Commit**.
3. Enter the appropriate comment and click **Finish**.

You can also perform the synchronization in the **Synchronize** view in the **Team Synchronizing** perspective (refer to the image listed below).

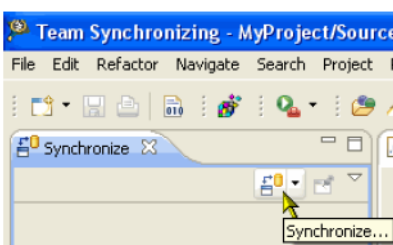


Figure 2-60. Synchronize View

The **Synchronize** view displays what has been changed either locally or remotely. For example, in the **Synchronize** view (in the following figure), the file `main.c` is on revision 1.2 with some outgoing changes, that is some changes are done locally in the file, and there is a new file `test.c` which is not committed yet.

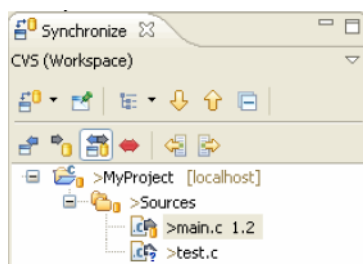


Figure 2-61. Synchronize View Displaying File Revisions

2.5 Search and Replace Action

In this topic, Search and Replace related FAQs are listed.

- How do I perform a simple search in the CodeWarrior IDE?
- Is it possible to preview the changes of the Replace action?
- Is there a way to change the variable or the function name in a CodeWarrior project?
- How can I search for specific files in a workspace?
- Is there a way to replace a variable easily?
- How can I go back to the previous location if the Editor takes me to another source or header file when, for example, I select the Open Declaration option?
- Is there a shortcut to open the search window for a specified text that is equivalent to the Ctrl + Shift + M shortcut key available in the Classic CodeWarrior IDE?
- Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings?

2.5.1 How do I perform a simple search in the CodeWarrior IDE?

You can either use the **Search** menu in the menu bar or select the **Edit > Find/Replace** command. However, there is a difference. The **Search** menu displays the commands **C/C++**, **Search**, **File**, and **Text**.

- **C/C++** - Opens the search dialog on the C/C++ search page
- **Search** - Opens the search dialog for your current editor
- **File** - Opens the search dialog on the File search page
- **Text** - Opens the submenu for full-text search in given scope (workspace, project, file or working set)

Selecting the **Select** menu opens the **Search** dialog box that includes two specialized tabs: **File Search** and **C/C++ Search**.

Whereas, the **Find/Replace** command displays the **Find/Replace** dialog box that lets you search for an expression in the active editor, and replace the expression with a new expression.

Thus, for a simple find and replace use **Edit > Find/Replace**.

2.5.2 Is it possible to preview the changes of the Replace action?

Yes, to preview the changes of the Replace action, follow these steps:

Search and Replace Action

1. Select **Search > File** (refer to the image, *Search Menu* as listed below) from the IDE menu bar.

The **File Search** tab page of the **Search** dialog box appears (refer to the image, *Search Dialog Box* as listed below).

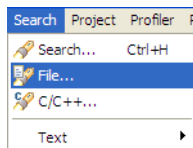


Figure 2-62. Search Menu

2. Specify the text that you want to replace in the **Containing text** text box.

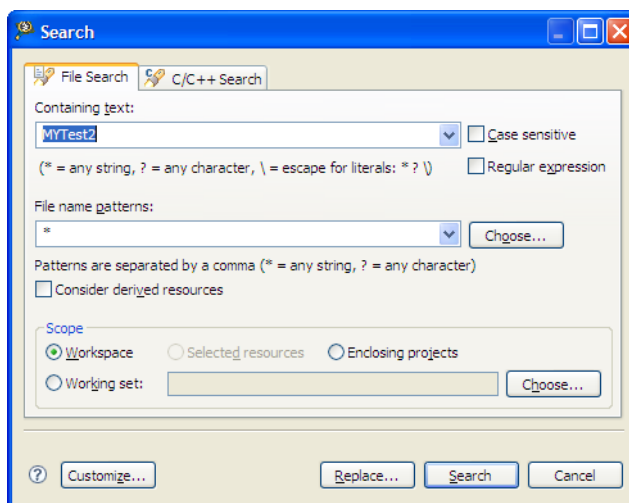


Figure 2-63. Search Dialog Box

3. Click the **Replace** button. The **Replace Text Matches** dialog box appears.
4. In the **With** text box, enter the text with which you want to replace the text entered in the **Replace** text box.



Figure 2-64. Replace Text Matches Dialog Box

5. Click the **Preview** button.

The dialog box lists the files to which the changes will be performed as a result of the replace action. You can preview each of the change and check or clear a file to accept or reject the change as required.

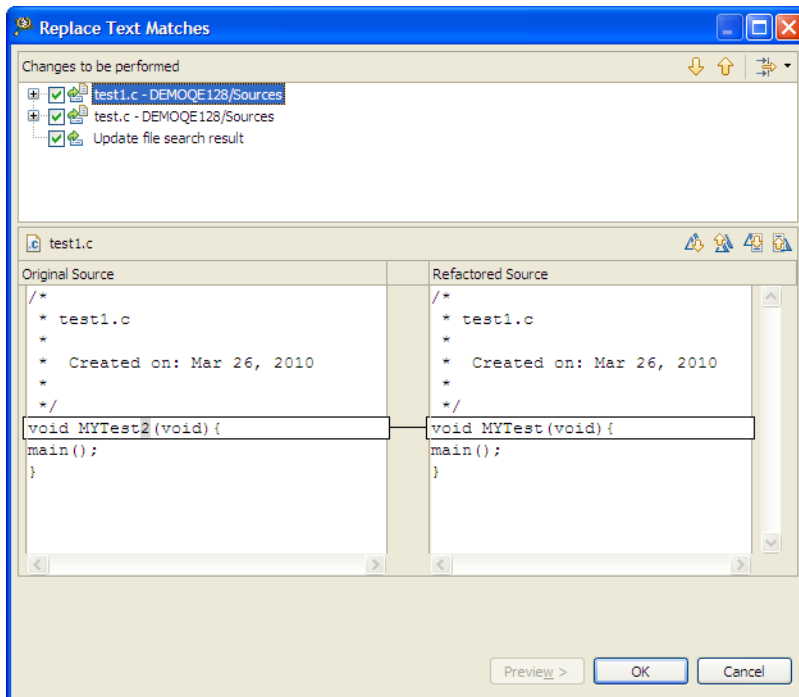


Figure 2-65. Preview Replace Action

6. Click **OK**.

The **Replace Text Matches** dialog box is closed.

The changes as a result of the replace action are performed to the selected files.

2.5.3 Is there a way to change the variable or the function name in a CodeWarrior project?

Yes, you can use the Refactoring feature of the CodeWarrior IDE to replace a function or a variable name.

1. Select the variable or function and select **Refactor > Rename** from the IDE menu bar.

The **Rename global variable`<variable name>'** dialog box appears.

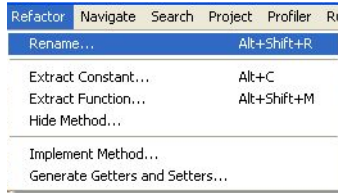


Figure 2-66. Refactor Menu

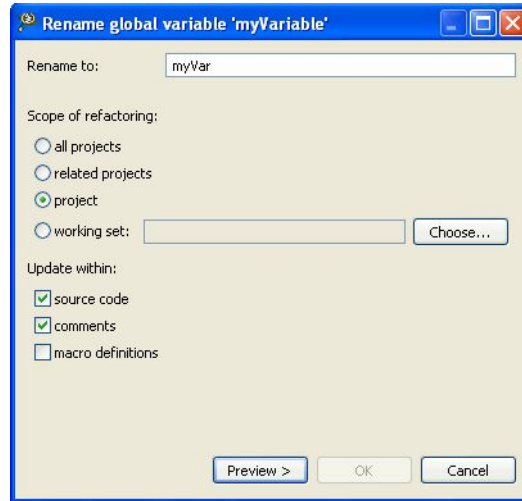


Figure 2-67. Rename global variable Dialog Box

2. Specify the new name for the variable.
3. Specify the scope of refactoring, such as all projects, related projects, project, or working set.
4. Specify where in the source file you want to make the changes, such as source code, comments, and/or macro definitions.
5. Click the **Preview** button to preview the changes.

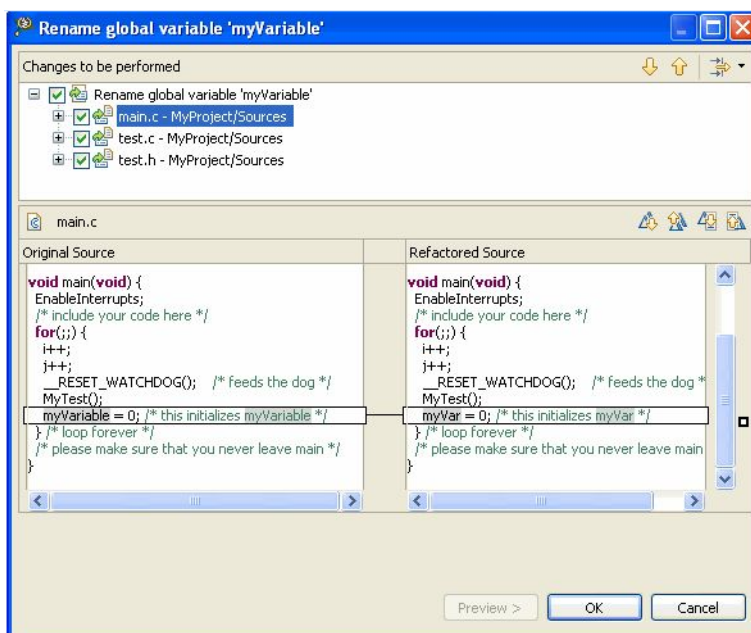


Figure 2-68. Preview Replace Action

6. Check or clear a file to accept or reject a change as required.
7. Click **OK** .

The specified changes are performed.

2.5.4 How can I search for specific files in a workspace?

To search for specific files in a workspace, follow these steps:

1. From the IDE menu bar, select the **Search > File...** option.

The **Search** dialog box appears.

2. Specify the search string in the **Containing Text** field. Specify other search options according to your requirements (refer to the image listed below).

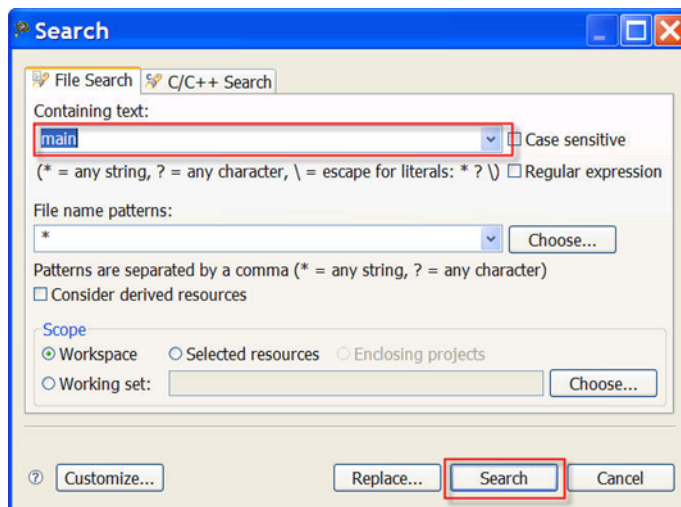


Figure 2-69. Search Dialog Box

3. Click the **Search** button.

The **Search** view appears (as the figure listed below shows). The **Search** view displays the results of your search. Right-click any item in the **Search** view to open a pop-up menu that allows you to remove items from the list, copy search results to the clipboard, or rerun the search.

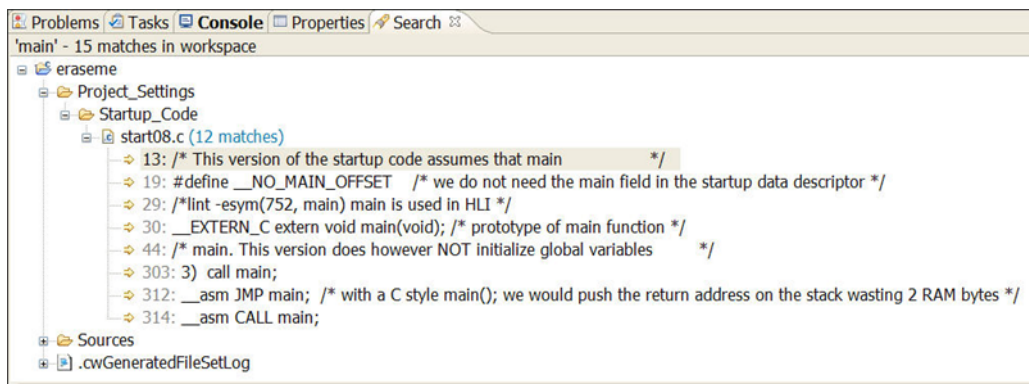


Figure 2-70. Search View

2.5.5 Is there a way to replace a variable easily?

Yes, to replace a variable, follow these steps:

1. From the IDE menu bar, select the **Search > File...** option.

The **Search** dialog box appears.

- Specify the search string in the **Containing Text** field. Specify other search options according to your requirements (refer to the image listed below).

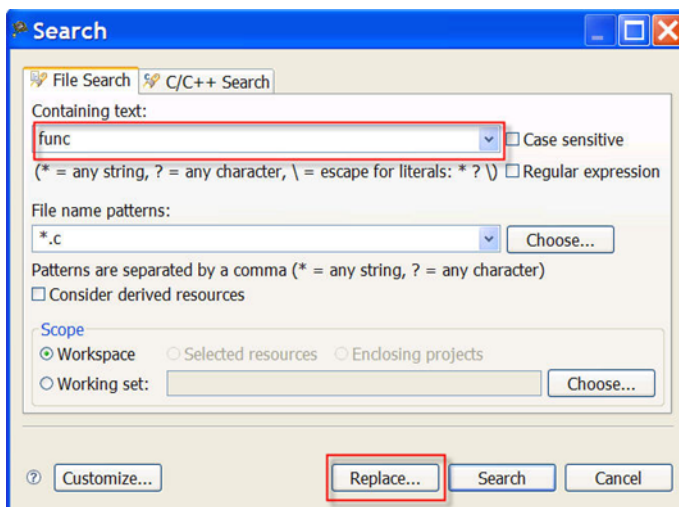


Figure 2-71. Search Dialog Box

- Click **Replace**.

The **Replace Text Matches** dialog box appears.

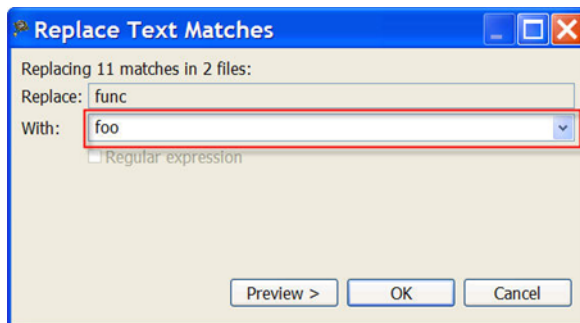


Figure 2-72. Replace Text Matches Window

- Click **Preview**.

The **Replace Text Matches** window appears (refer to the image listed below). This window gives lets you preview the changes that will be made. This allows you to inspect each proposed change and accept or deny it individually.

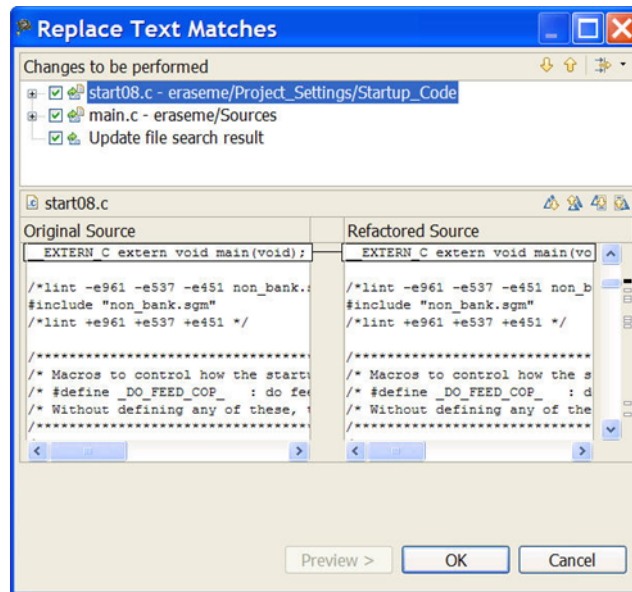


Figure 2-73. Replace Text Matches Window

2.5.6 How can I go back to the previous location if the Editor takes me to another source or header file when, for example, I select the Open Declaration option?

Use the **Alt+left** shortcut key. This option is also listed under the **Navigation** menu.

2.5.7 Is there a shortcut to open the search window for a specified text that is equivalent to the Ctrl + Shift + M shortcut key available in the Classic CodeWarrior IDE?

You can create a custom key binding for the **Ctrl+Shift+M** shortcut key in the **Preferences > General > Keys** dialog box.

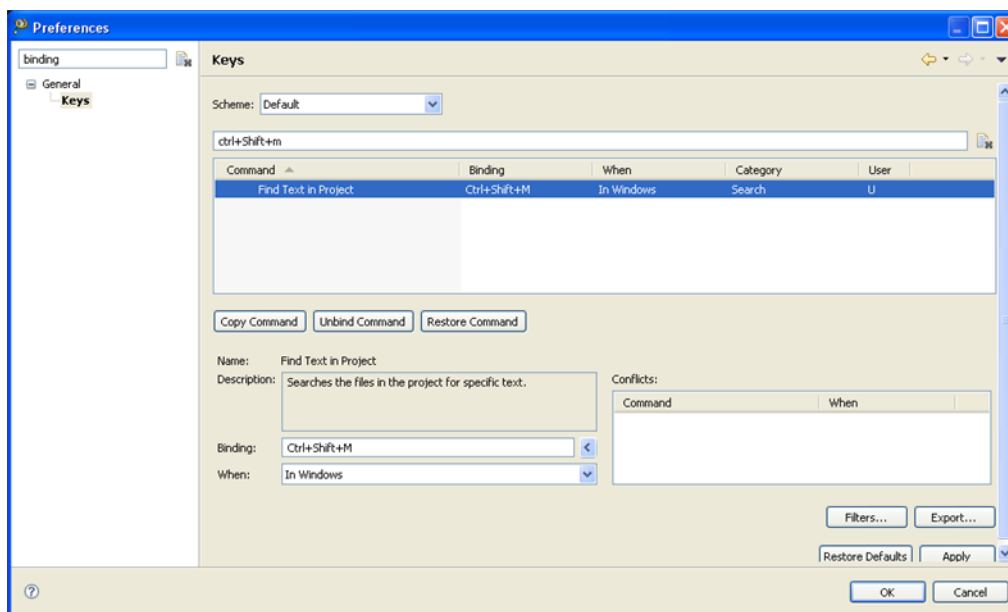


Figure 2-74. Preferences Dialog Box - Keys Page

NOTE

For more information on key binding, refer to the topic [How can I change a key binding?](#)

2.5.8 Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings?

The files you need to store in a version control system:

- `.project` and `.cproject` - contains the list of files and the build tool settings.
- `.ttf` - targets the needing target tasks for flash programming (ColdFire V2 and Kinetis IAR JLink)
- `.launch` - required for the debugger launch configuration
- `rseHostSettingsCache.xml` - contains the cached RSE settings. You need this to have the RSE settings imported.
- `*.c/*.h/etc` source files and linker file, if not using Processor Expert
- If you are using Processor Expert - `ProcessorExpert.pe` (contains all the settings), `events.c/h`, and the `ProcessorExpert.c` with user modifications

Any generated folders (like `PE Generated_Code`, `PE Documentation` or the folder where the object/make files are placed (usually named by the CPU name) can be stripped off to reduce file size.

You can also export/store the workspace settings, like editor/VCS settings/etc, by selecting, **File > Export > General > Preferences** from the IDE menu bar.

2.6 Miscellaneous

In this topic, miscellaneous FAQs related to the CodeWarrior IDE are listed.

- [How can I improve the performance of the CodeWarrior IDE I am working with?](#)
- [What is the first thing that I see when I start the CodeWarrior IDE?](#)
- [What is a perspective?](#)
- [How can I find the version of the CodeWarrior that I am using?](#)
- [Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project?](#)
- [How do I update my local history settings?](#)
- [Is there a way to filter the settings in the Preferences window to find a particular setting?](#)
- [How can I change a key binding?](#)
- [Is it possible to view the change history of a source code file?](#)
- [How can I view a graphical representation of source code in the CodeWarrior IDE using Dot and Doxygen?](#)
- [How can I determine which header files are associated with my source code file?](#)
- [How do I view call hierarchy in my source code?](#)
- [Why the project that I just created is not visible in the CodeWarrior Projects view?](#)
- [I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why?](#)
- [Why does the Rename option appear grayed out?](#)
- [Why duplicating a configuration in the debugger perspective does not duplicate the run configuration?](#)
- [Can I still use Ctrl+Tab keys to navigate between open windows?](#)
- [How can I change the debugger key bindings to the ones that I used to have in the Classic CodeWarrior IDE?](#)
- [How can I change the name of the executable that is generated when I build my project?](#)
- [Why the console view does not display all the warning and error messages by default?](#)
- [Is there a file that contains all the launch configuration settings?](#)
- [How the CodeWarrior Project Importer handles recursive access paths?](#)
- [Can I disassemble my source code file?](#)
- [Where exactly the disassemble file gets created?](#)

- Why the .metadata folder in my workspace stores a huge history?
- How to have projects from different workspaces or locations in a workspace?
- How can I switch to another workspace?
- What is the purpose of the Tasks view?
- What is the purpose of the Properties view?
- What is the purpose of the Outline view?
- How can I find out if certain files contain debug information?
- How can I resolve the following error message that I get when I start the CodeWarrior IDE?
- How can I open an existing project in the CodeWarrior IDE?
- What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button?
- How can I start the post-build steps in the CodeWarrior IDE?
- Is it necessary to have the project name identical to the name of the directory that contains the .project file?
- Where does the CodeWarrior IDE save the debug configuration as a local file by default?
- When I save my launch configurations as a local file and then delete the project, all the local configurations get deleted as well. How can I resolve this issue?
- How can I modify and save the files in GBK encoding using the CodeWarrior IDE?
- How are the PARENT-COUNT-MyVariable definitions defined in the CodeWarrior IDE?
- Is there a way to instruct the CodeWarrior IDE to use relative paths instead of absolute ones to store the project file location in a workspace?
- **NEW** What is a Working Set? How do I build/compile a Working Set?
- **NEW** How do I search within the Working Set?
- **NEW** How do I manage the Working Sets?
- **NEW** How can I filter the projects listed in the CodeWarrior Projects view based on Working Sets?
- **NEW** How can I import/export a Working Set in the CodeWarrior eclipse IDE?
- **NEW** What are Icon decorators?
- **NEW** How do I enable or disable an icon decorator?
- **NEW** Can the icon decorators be combined?
- **NEW** When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, "You are opening a large file. Scalability mode has been turned on for this editor to help improve performance by disabling features requiring to parse the source file language." What does it mean?
- **NEW** How do I configure the Scalability settings?

2.6.1 How can I improve the performance of the CodeWarrior IDE I am working with?

Yes, you can improve the performance of your CodeWarrior IDE by disabling the features that slow down the software.

1. Disable Indexer Options

- a. From the CodeWarrior IDE menu bar, select **Window > Preferences**.

The **Preferences** dialog box appears.

- b. Type **index** as the filter text to narrow down the list of preferences.
- c. Select **C/C++ > Indexer**.

The **Indexer** page appears in the right panel of the **Preferences** dialog box.

- d. Clear the **Enable indexer** checkbox.

2. Disable Scalability Mode Settings

- a. In the **Preferences** dialog box, type **scalability** as the filter text.
- b. Select **C/C++ > Editor > Scalability**.

The **Scalability** page appears in the right panel of the **Preferences** dialog box.

- c. Clear the **Enable all scalability mode options** checkbox.

3. Disable Content Assist Auto Activation

- a. In the **Preferences** dialog box, type **content** as the filter text.
- b. Select **C/C++ > Editor > Content Assist**.

The **Content Assist** page appears in the right panel of the **Preferences** dialog box.

- c. Clear all checkboxes in the **Auto-Activation** section.

4. Click **OK**.

The performance of the CodeWarrior IDE improves.

2.6.2 What is the first thing that I see when I start the CodeWarrior IDE?

The window that you see when you bring up the IDE is called a workbench. A workbench consists of perspective, views, and editors. The term *workbench* refers to the host development environment.

2.6.3 What is a perspective?

A perspective is an arrangement of views and editors. You can quickly switch between different perspectives.

2.6.4 How can I find the version of the CodeWarrior that I am using?

To find the version of the CodeWarrior that you are using, follow these steps:

1. From the IDE menu bar, select the **Help > About CodeWarrior Development Studio** option.

The **About CodeWarrior Development Studio** dialog box appears.

The dialog box that appears displays the version number and the build number under **Installed Products** section.

2.6.5 Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project?

Yes, you can retrieve your deleted files. However, instead of the system's Recycle Bin you need to retrieve the file from the local history. The local history maintains a copy of files you modify or delete.

To restore a deleted file, you need to perform the following steps.

1. In the **CodeWarrior Projects** view, right-click the project branch you deleted the file from.
2. From the context menu, select the **Restore from Local History** command.

The **Restore from Local History** dialog box appears.

NOTE

The dialog box contains a list of deleted files, along with a checkbox for each file in the list.

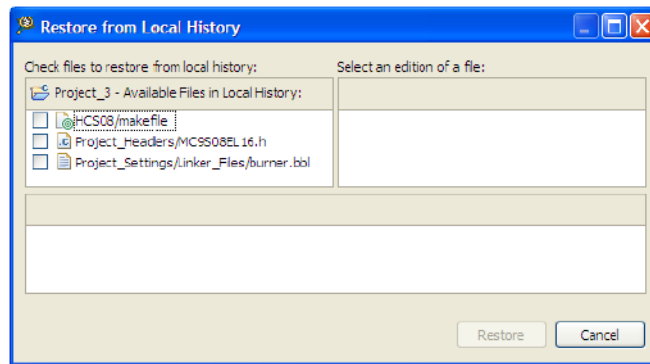


Figure 2-75. Restore from Local History Dialog Box

3. Check the files you want to restore from the local history.
4. Click the **Restore** button.

The selected files are restored.

2.6.6 How do I update my local history settings?

To update the local history settings or preferences, perform the following steps.

1. From the IDE menu bar, select the **Window > Preferences** option.
The **Preferences** dialog box appears.
2. Expand the tree control to select the **General > Workspace > Local History** option.
The **Local History** page appears.

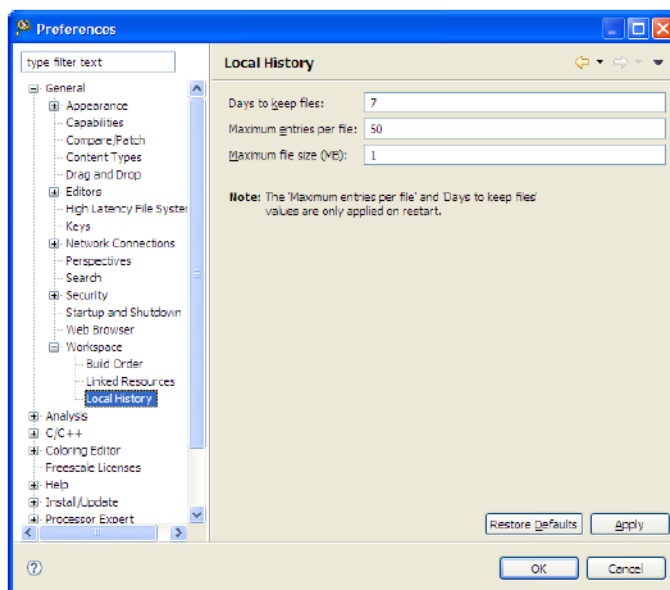


Figure 2-76. Preferences Dialog Box - Local History Page

3. In the **Days to keep files** field, specify the number of days you want to maintain changes in the local history. The default value is 7.

NOTE

History state older than the default value will be lost.

4. In the **Maximum entries per file** field, specify how many history states per file you want to maintain in the local history. The default value is 50.

NOTE

If you exceed the default value, you will lose older history to make room for new history.

5. In the **Maximum file size (MB)** field, specify the maximum size of individual states in the history store. The default value is 1.

NOTE

If you exceed the default value, the individual states will not be stored.

6. Click **Apply** to confirm and save the settings.
7. Click **OK**.

The **Preferences** dialog box closes.

The local history settings is updated.

NOTE

To revert to the default settings, click the **Restore Defaults** button in the **Preferences** dialog box.

2.6.7 Is there a way to filter the settings in the Preferences window to find a particular setting?

Yes, you can filter settings using the filter text. For example, if you are looking for some settings to change the color in a view, type **color** as the filter text in the Preferences dialog box (refer to the image listed below). The preferences with text **color** in it are listed. This reduces the number of preferences to browse through.

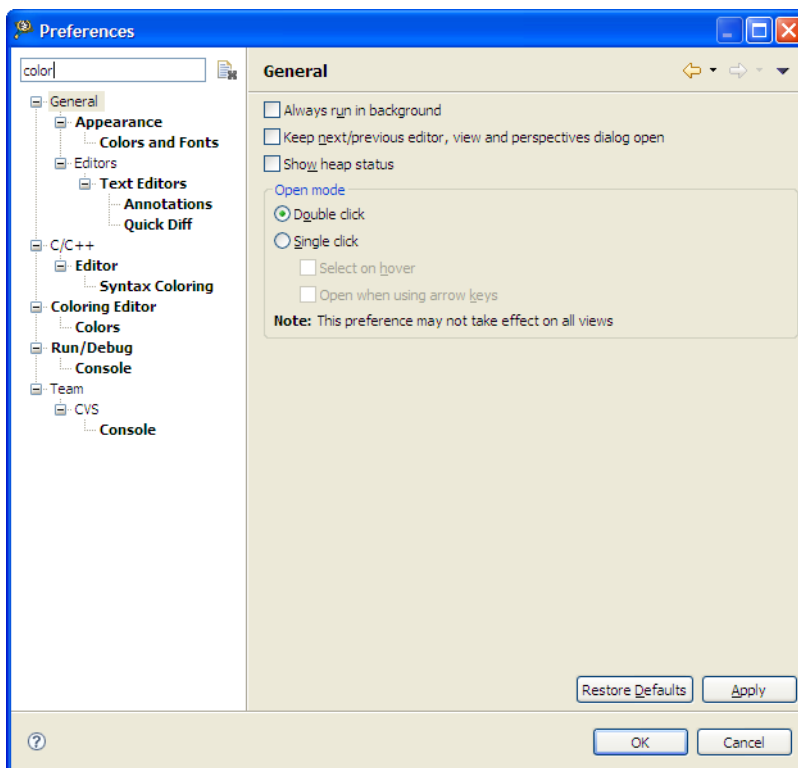


Figure 2-77. Filtered Preferences

You can clear the filter text using either the keyboard or the 'X' icon.

You can also use wildcards in the filter text.

- * - Substitutes for zero or more characters
- ? - Substitutes for any one character

2.6.8 How can I change a key binding?

To change a keyboard shortcut or a keyboard binding:

1. From the IDE menu bar, select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Type filter text as **key** to narrow down the list of the preferences.

The modifications need to be done in the **General > Editors > Keys** preference page.

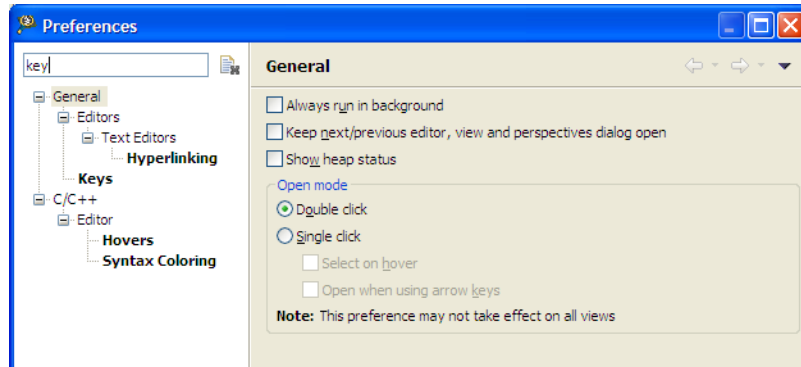


Figure 2-78. Preferences Dialog Box

3. Select **General > Editors > Keys** .

The **Keys** preferences page appears in the right panel of the **Preferences** dialog box.

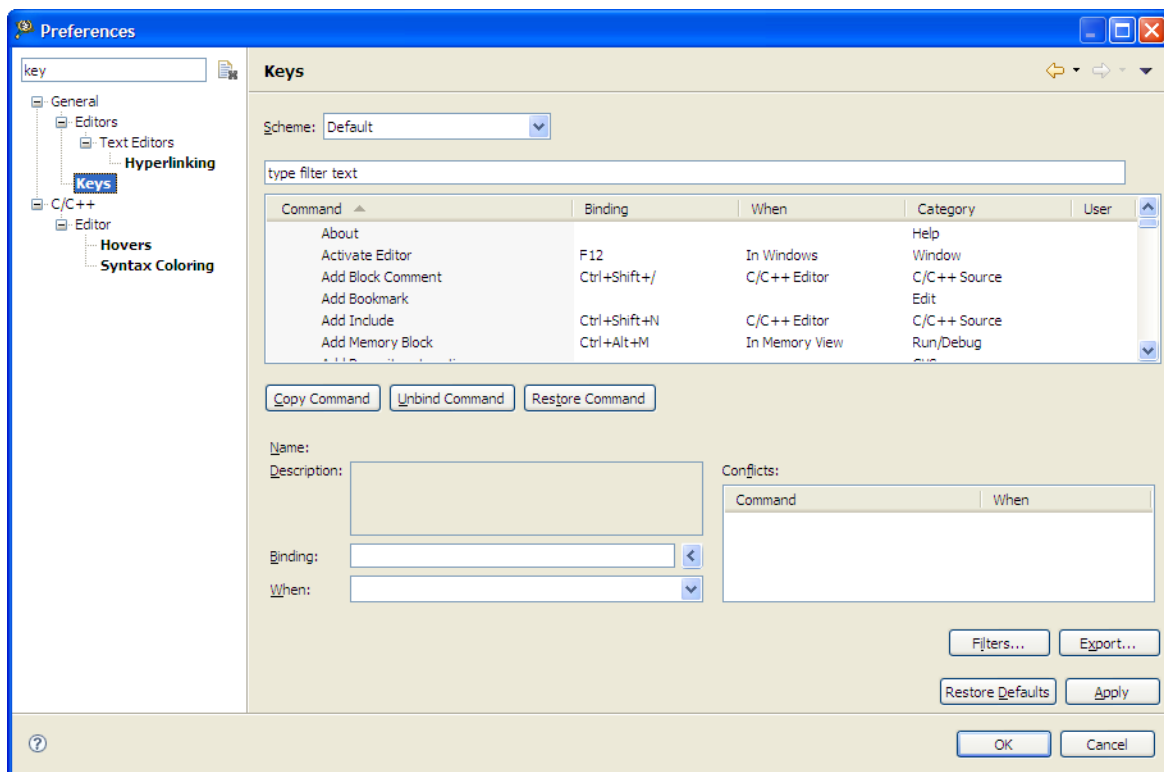


Figure 2-79. Keys Preferences Page

4. From the **Scheme** drop-down list select the require scheme, such as **Microsoft Visual Studio** or **Emacs** .
5. Type **content assist** as the filter text to filter the list of keys.

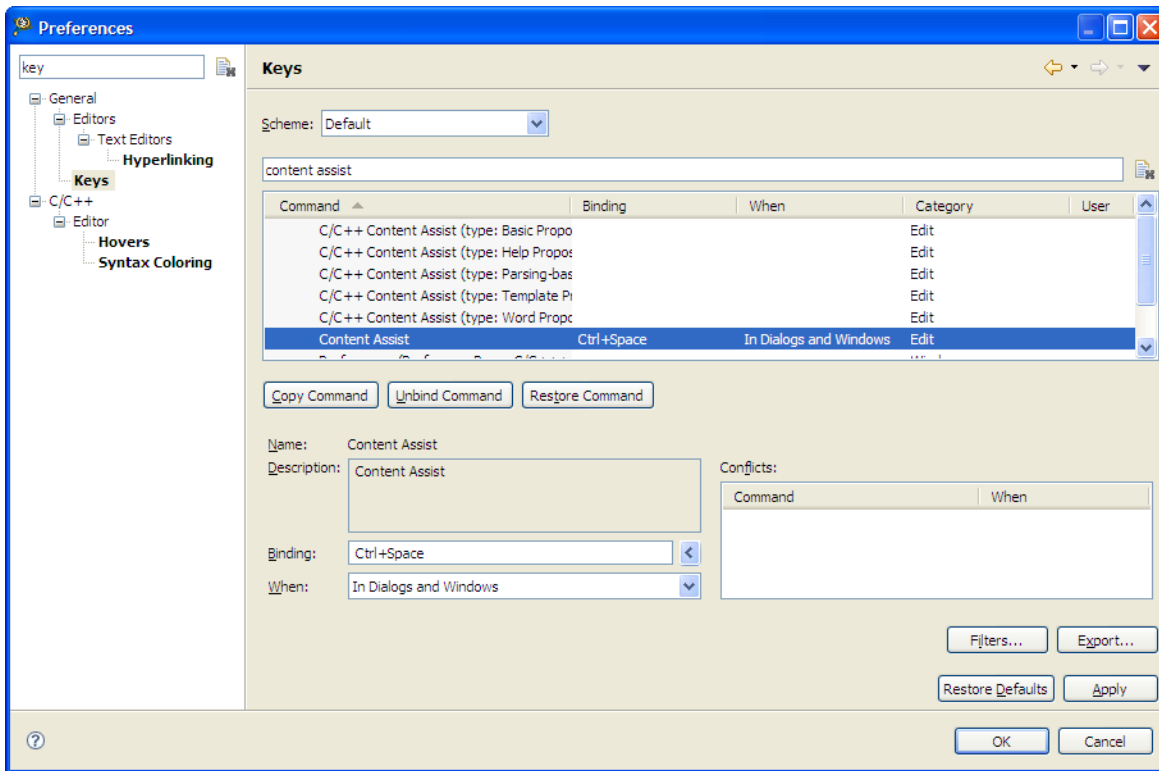


Figure 2-80. Filtered List of Commands in Keys Preference Page

6. Select the entry you want to modify and type the new key binding, such as **Ctrl+Alt +C** in the **Binding** text box. You can also assign multiple key bindings.
7. From the **When** drop-down list, select the context in which the key binding apply.
8. Click **OK** .

The **Preferences** dialog box closes.

The keyboard shortcut or the keyboard binding are modified, as specified.

2.6.9 Is it possible to view the change history of a source code file?

Yes, you can view the history of the changes done in a source file by following the steps given below:

1. Right-click the source file and select **Compare With > History** from the context menu.

The **History** view opens displaying the date and time when the changes were made in the source file.

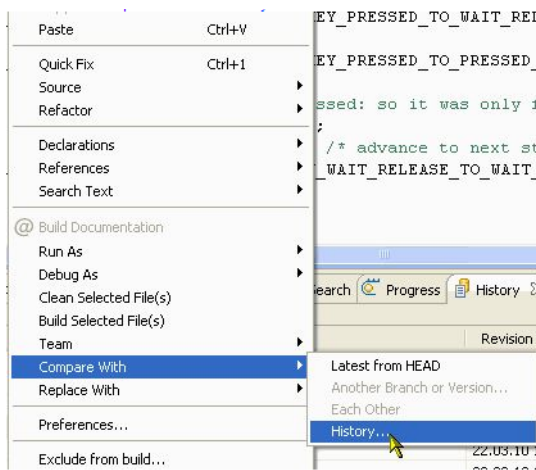


Figure 2-81. Compare With > History

2. Double-click the required revision time in the **History** view.

The editor area displays the comparison of different file revisions, and also highlights the changes. You can determine the changes done in the source file by browsing through the compared files.

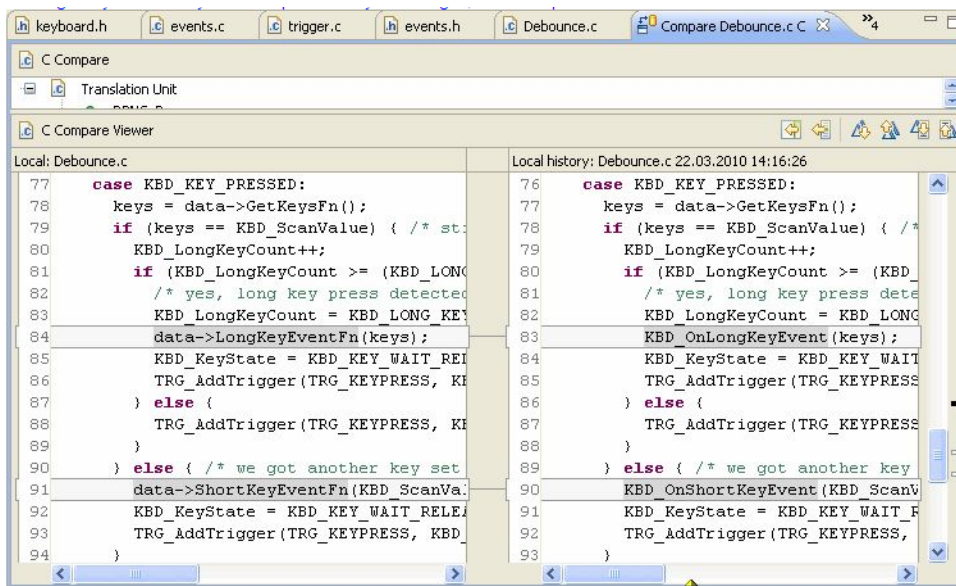


Figure 2-82. Comparison of File Revisions

2.6.10 How can I view a graphical representation of source code in the CodeWarrior IDE using Dot and Doxygen?

You need to install doxygen, dot/Graphviz, and Eclox eclipse plugins from the following locations respectively:

www.doxygen.org

www.graphviz.org

<http://download.gna.org/eclox/update>

While working with Dot+Doxygen, you can view graphical representation of the source code. For example, if you have a state machine, describe the state machine (as shown in the figure, *State Machine Description* listed below) in the source file.

Dot+Doxygen will then display the graphical representation of the state machine, as shown in the figure *Graphical Representation of State Machine* listed below.

```

42#define KBD_LONG_KEY_ITERATIONS (KBD_LONG_KEY_TIME_MS/(KBD_KEY_PRESSED_TO_PRESSED_NOF_10MS*10))
43 /*!< Number of iterations we need to go for a long key detection. */
44
45/*! \brief Key scan routine which implements the state machine.
46\dot
47digraph example_api_graph {
48 node [shape=box];
49 KBD_IDLE [fillcolor=lightblue,style=filled,label="KBD_KEY_IDLE" ];
50 KBD_PRESSED [fillcolor=lightblue,style=filled,label="KBD_KEY_PRESSED" ];
51 KBD_WAIT [fillcolor=lightblue,style=filled,label="KBD_KEY_WAIT_RELEASE"];
52 KBD_IDLE -> KBD_PRESSED -> KBD_WAIT -> KBD_IDLE ;
53 KBD_PRESSED -> KBD_PRESSED ;
54 KBD_IDLE -> KBD_IDLE ;
55}
56\enddot
57*/
58

```

Figure 2-83. State Machine Description

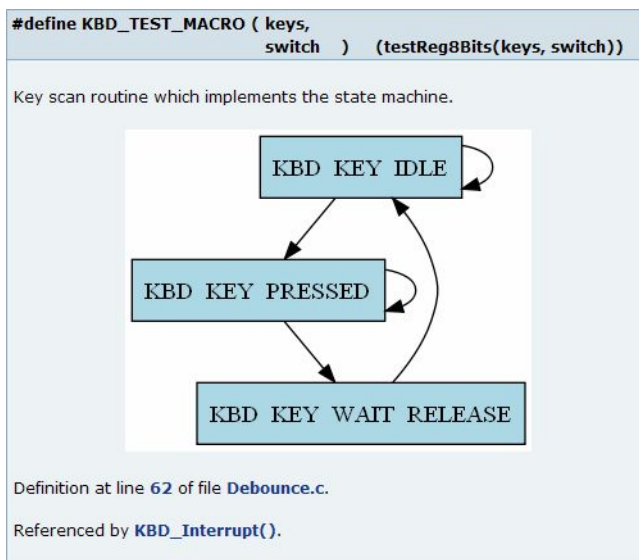


Figure 2-84. Graphical Representation of State Machine

Dot+Doxygen also gives you a graphical view of all include file in the source file.

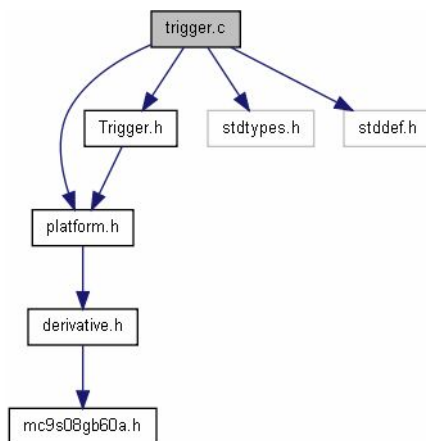


Figure 2-85. Graphical View of Include Files

2.6.11 How can I determine which header files are associated with my source code file?

To determine which header files are associated with your source code file, you can view the graphical representation of the include files in the **Include Browser** view.

1. Select **Window > Show View > Other**.
2. Select **C/C++ > Include Browser** in the **Show View** dialog box and click **OK**.

The **Include Browser** view appears.

3. Drag and drop the required source file from the **CodeWarrior Projects** view in the **Include Browser** view.

The **Include Browser** view displays all the include file in the selected source file (refer to the image listed below).

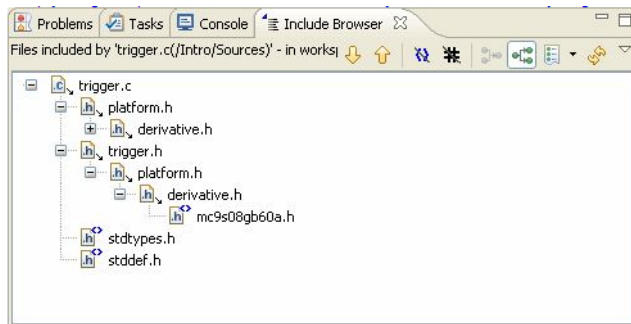


Figure 2-86. Include Browser View

2.6.12 How do I view call hierarchy in my source code?

To view or browse through the call hierarchy:

1. Right-click on a symbol in the source file and select **Open Call Hierarchy** .

The call graph appears in the **Call Hierarchy** view.

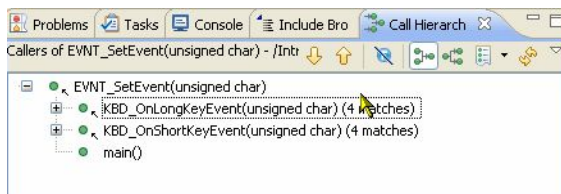


Figure 2-87. Call Hierarchy View

2. Click the **Show Callees** icon in the **Call Hierarchy** view toolbar.

The callees appear in the view.

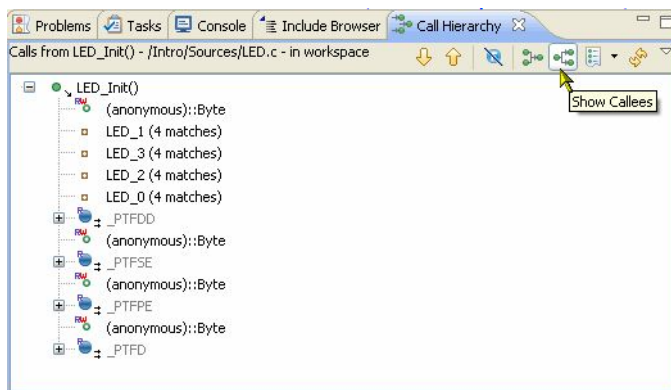


Figure 2-88. Show Callees

2.6.13 Why the project that I just created is not visible in the CodeWarrior Projects view?

A newly created project might not be visible in the CodeWarrior Projects view if you choose to use working sets. This is because a new project is not added to the active working set automatically. Thus, to see the new project in your active working set, select **Edit Active Working Set** from the view pull-down menu. From the **Working set content** options, select the newly created project and click the **Finish** button. The new project will now be visible in the active working set.

2.6.14 I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why?

The **CodeWarrior Projects** view cannot watch for changes in your computer's file system. When you create a new file away from the Eclipse platform the CodeWarrior application is not aware of the changes. Thus to view the new file, you need to force the **CodeWarrior Projects** view to update its content. To forcefully update contents, select the project whose directory contains the updated file and from the **File** menu bar select the **Refresh** command. Alternatively, you can press the F5 key. The new file will now appear in the **CodeWarrior Projects** view.

2.6.15 Why does the Rename option appear grayed out?

The **Rename** command will be enabled only if a file or folder is selected in the **CodeWarrior Projects** view. In case you select a file or folder and click elsewhere on the Workbench, the **Rename** command appears grayed out. So, ensure that you select the desired file or folder (highlighted in blue) before trying to rename it.

2.6.16 Why duplicating a configuration in the debugger perspective does not duplicate the run configuration?

In CodeWarrior, the debug and the launch configuration are not associated with a build configuration. Instead, they are associated with an executable and a project. For example, by duplicating the build configuration, the executable remains the same. Therefore, even for the new build configuration, the old launch/debug configurations still apply. This behavior is different than the previous CodeWarrior IDE tools in that respect.

2.6.17 Can I still use Ctrl+Tab keys to navigate between open windows?

Yes, you can. When you select Ctrl+Tab, a new floating window appears that lets you navigate between the open CodeWarrior windows.

2.6.18 How can I change the debugger key bindings to the ones that I used to have in the Classic CodeWarrior IDE?

The corresponding default binding for this functionality is Ctrl+F6. You can reassign this feature to Ctrl+Tab by doing the following steps:

1. From the IDE menu bar, select **Window > Preferences**.

The **Preferences** dialog box appears.

2. Select **General > Keys**.

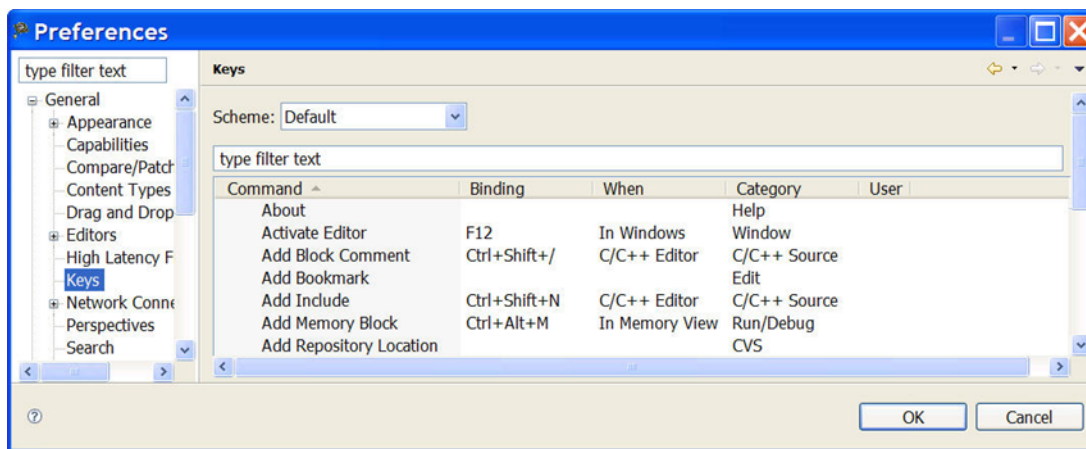


Figure 2-89. Preferences Dialog Box-Keys Page

Under the **Command** column is a list of available commands. The **Binding** column contains the keys that can be used to invoke the command. The **When** column indicates when this command will be executed.

3. Click on the **Command** column to get the key sequence in alphabetical order.

For the debugger, the keys to change are **Step Into**, **Step Over** and **Step Return**.

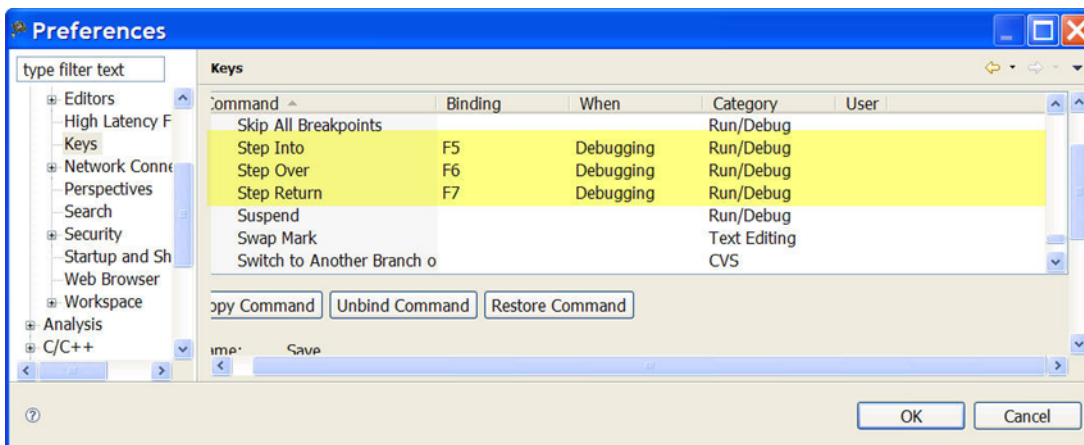


Figure 2-90. Preferences Dialog Box-Debugger Keys

By default, these key bindings are set to F5, F6 and F7 respectively. In order to change the key sequence for Step Over from F6 to F10, follow these steps:

- Select **Step Over** in the **Command** column.
- From the **Binding** textbox, remove F6 by deleting it from the binding text box.
- Press F10 on computer keyboard so that F10 will be inserted into Binding textbox.

The user change symbol, which is a blue triangle, will appear in the right-most column to indicate a user binding.

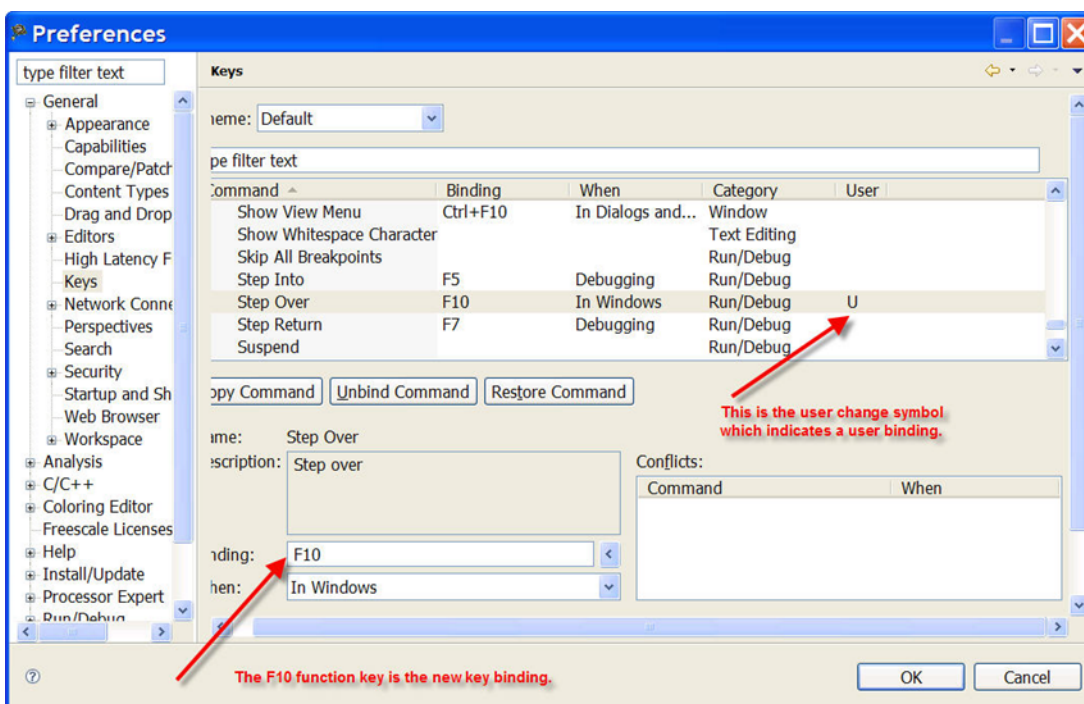


Figure 2-91. Preferences Dialog Box-New Key Binding

- Click **Apply**.

The Binding for **Step Over** appear as **F10**. When you use debugger, press the **F10** key for a **Step Over**. In the context menu that appears on the Debugger window, you see **Step Over F10**.

4. Click **OK**.

The **Preferences** dialog box closes.

The default key bindings changes, as specified.

2.6.19 How can I change the name of the executable that is generated when I build my project?

To change the name of the executable that is generated when you build your project, follow the steps given below:

1. Select the project in the **CodeWarrior Projects** view and select **Project > Properties** from the IDE menu bar.

The **Properties** dialog box appears.

2. Select **C/C++ Build > Settings** from the left panel.

The **Settings** page appears in the right panel of the **Properties** dialog box.

3. Select **Build Artifact** tab.

The **Artifact name** textbox contains the name of the executable that will be created when the **Build Project** option is selected. The **Artifact extension** textbox contains the name extension of the binary that will be created. If you want to change the name of the executable, replace the one in the **Artifact name** textbox.

4. Click **Apply**.
5. Click **OK**.

Now when you build the project using the **Build Project** option, the new binary file name appears in the project folder.

2.6.20 Why the console view does not display all the warning and error messages by default?

By default, only 500 lines are displayed in the console window. In order to change the number of lines displayed in the console window, follow these steps:

1. From the IDE menu bar, select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Select **C/C++ > Build Console** from the left panel.

The **Build Console Settings** page appears.

3. Change the **Limit console output (#lines)** to the number of lines that you would like to appear in the console window. By default, it is set to 500.
4. Click **Apply** .
5. Click **OK** .

The number of lines displayed in the console window changes, as specified.

NOTE

Since this is a global setting, the console window will display the specified number of lines for every project that is compiled within the workspace.

2.6.21 Is there a file that contains all the launch configuration settings?

Yes there is a file that describes the launch configuration settings. This file is located in the project folder with a **.launch** extension.

2.6.22 How the CodeWarrior Project Importer handles recursive access paths?

The CodeWarrior Project Importer allows the user to automatically have the recursive access paths flattened. For example, if the original project has a recursive access path "{\Project}\", the imported access paths could be something like this:

```
"{Project}\"
```

```
"{Project}\src"
```

```
"{Project}\src\subdir"
```

2.6.23 Can I disassemble my source code file?

Yes you can disassemble source code files. To create a disassemble file follow steps given below:

1. Right-click the source file

A context menu appears.

2. From the context menu, select **Disassemble** option.

The **Disassemble Job** window appears.

The disassembling file provides a way to show the results of object code produced from a C/C++ source file in the Editor. Once the Disassemble command is executed, it will proceed to compile, disassemble the file and show the resulting disassembled file in a new editor window, titled **sourcefilenameXXXXX.lstXXXXXXXXXXXXXXXXXXXX.txt** , where **XXXXXXXXXXXXXXXXXXXX** represent random numbers.

2.6.24 Where exactly the disassemble file gets created?

This file gets created in the `temp` directory located under:

```
<Profiles>\<username>\<LocalSettings>\<Temp>
```

Under this path you find **sourcefilenameXXXXX.lstXXXXXXXXXXXXXXXXXXXX.txt** file.

2.6.25 Why the .metadata folder in my workspace stores a huge history?

This is a standard Eclipse feature. Eclipse keeps a local history of edited files by default. It lets you restore a file content from a previously saved local history by right-clicking a source file and selecting the **Restore from local history...** option. You can configure how the local history is stored. See [How do I update my local history settings?](#)

2.6.26 How to have projects from different workspaces or locations in a workspace?

You can have projects which are in different workspaces or locations in your workspace. For this, you can import projects from other workspaces, without copying them into your workspace.

However, keep in mind that you can only have one workspace open in one eclipse instance. If you want to view multiple workspaces side by side, then run another instance of the CodeWarrior software with a different workspace.

2.6.27 How can I switch to another workspace?

To switch to another workspace:

1. From the CodeWarrior IDE menu bar, select **File > Switch Workspace > Other** .

The **Workspace Launcher** dialog box appears.

2. Click **Browse** .

The **Select Workspace Directory** dialog box appears.

3. Select the workspace you want to switch to.
4. Click **OK**.
5. Click **OK**.

The selected workspace appears in the **CodeWarrior Projects** view.

2.6.28 What is the purpose of the Tasks view?

The **Tasks** view lets you view tasks that have been created.

NOTE

The task does not refer to operating system related process or task, but with a (engineering) work task.

A task could be created to record reminders or follow-up on something later. In order to create a **Task**, select **Add Task** in the **Task** view. Tasks can also be linked to a resource that would allow you to write a note to yourself so that it can be executed or looked at a later time. In order to create a new task that is associated with a resource follow steps given below:

1. Double-click the source file.

Source file appears in the Editor view.

2. Right-click the left-hand side of source line where a task should be added.

A context menu appears.

3. From the context menu, select **Add Task**.

The **Properties** dialog box appears.

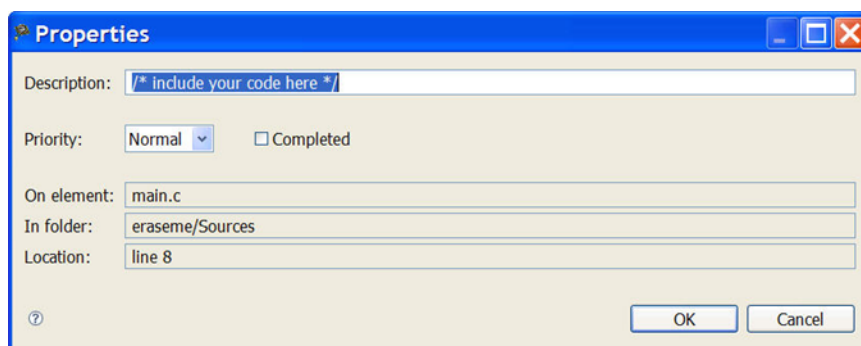


Figure 2-92. Properties Dialog Box

4. Type a description of what should be done with this task in the **Description** textbox.
5. Click **OK**.

The new task appears in the **Task** view.

Alternatively, you can also add a comment block with a `\todo` tag in the source code file. For example, following comment block in the source code file creates a new task in the **Tasks** view.

```
/*! \todo I need to fix this bug here... */
```

2.6.29 What is the purpose of the Properties view?

The purpose of the **Properties** view is to display property names and the basic properties of a selected resource. The kind of properties that are displayed depends on the specific resource selected as well as the features and plug-ins that are installed on the Workbench.

2.6.30 What is the purpose of the Outline view?

The Outline view displays an outline of a structured file that is currently open in the editor area, and lists the structural elements.

2.6.31 How can I find out if certain files contain debug information?

To find out if certain files contain debug information, see the file listing in the **CodeWarrior Projects** view.

To find out if certain files contain debug information:

1. Select the project in the CodeWarrior Projects view and select **Project > Properties** from the IDE menu bar.

The **Properties** dialog box appears.

2. Select **C/C++ Build > Settings** from the left panel.

The Settings page appears in the right panel of the **Properties** dialog box

3. Select **Power Linux Compiler > Debugging** option available under **Tool Settings** tab page. Set **Debug Level** to **Default (-g)** from the drop-down list.
4. Click **OK**.

This enables the debug option for the files.

2.6.32 How can I resolve the following error message that I get when I start the CodeWarrior IDE?

A debug or protocol-plugin license for the product expired or was not found

To resolve this error message, add the path to the V10.x license file to the LM_LICENSE_PATH environment variable. In addition, it is also possible that the launch config file is corrupted.

2.6.33 How can I open an existing project in the CodeWarrior IDE?

You can open only those projects that are already in your workspace and have been closed. To bring an external project (one that is created on another machine or in another workspace) into your workspace you can follow the steps given below:

1. From the C/C++ Perspective toolbar, select **File > Import** .

The **Import** dialog box appears.

2. Select **General Folder > Existing Project into Workspace**
3. Click **Next** .

The **Import Project s** dialog box appears.

4. Click **Browse** .

The **Browse For Folder** dialog box appears.

5. Select **Project to import**
6. Click **OK** .

The imported project appears under Projects in the Import Projects window.

7. Click **Finish** .

The selected project is imported into the workspace.

2.6.34 What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button?

It can not be replaced by the Properties button. The **Manage Configuration** button displays the list of defined configurations for a given project. You can select a button to add, rename, or remove a configuration. You can also set an active configuration.

2.6.35 How can I start the post-build steps in the CodeWarrior IDE?

In order to start a batch file after build, you just need to specify the name of the .bat file in the Command for post-build step.

Note that the current directory is the **Build location** specified in the project properties **C/C++ Build** panel. Therefore, you need to adjust the path to retrieve the .bat file relative to this location.

For example, consider that:

- the project is in `D:\Freescale\Work\Eclipse\testbss`
- the build location is `D:\Freescale\Work\Eclipse\testbss\Debug_SC3x50_PACC`
- the batch file is located in `D:\Freescale\Work\Eclipse\testcbe\testbss` and is called `postb.bat`.

Then, the post-build command is `..\postb.bat`.

2.6.36 Is it necessary to have the project name identical to the name of the directory that contains the .project file?

Yes, it has to be located in the folder with the same name as the project.

2.6.37 Where does the CodeWarrior IDE save the debug configuration as a local file by default?

It goes inside the .metadata subfolders and then it becomes user dependent. Perform a search on the launch and you will still find it. If you create a new launch configuration, it is saved as a local file by default.

2.6.38 When I save my launch configurations as a local file and then delete the project, all the local configurations get deleted as well. How can I resolve this issue?

Follow the steps given below:

1. From the IDE menu bar, select **Window > Preferences** .
2. Select **Run/Debug> Launching > Launch Configurations** .

The **Launch Configuration** settings appear.

3. Make sure that the **Delete configurations when associated resource is deleted** checkbox is cleared.

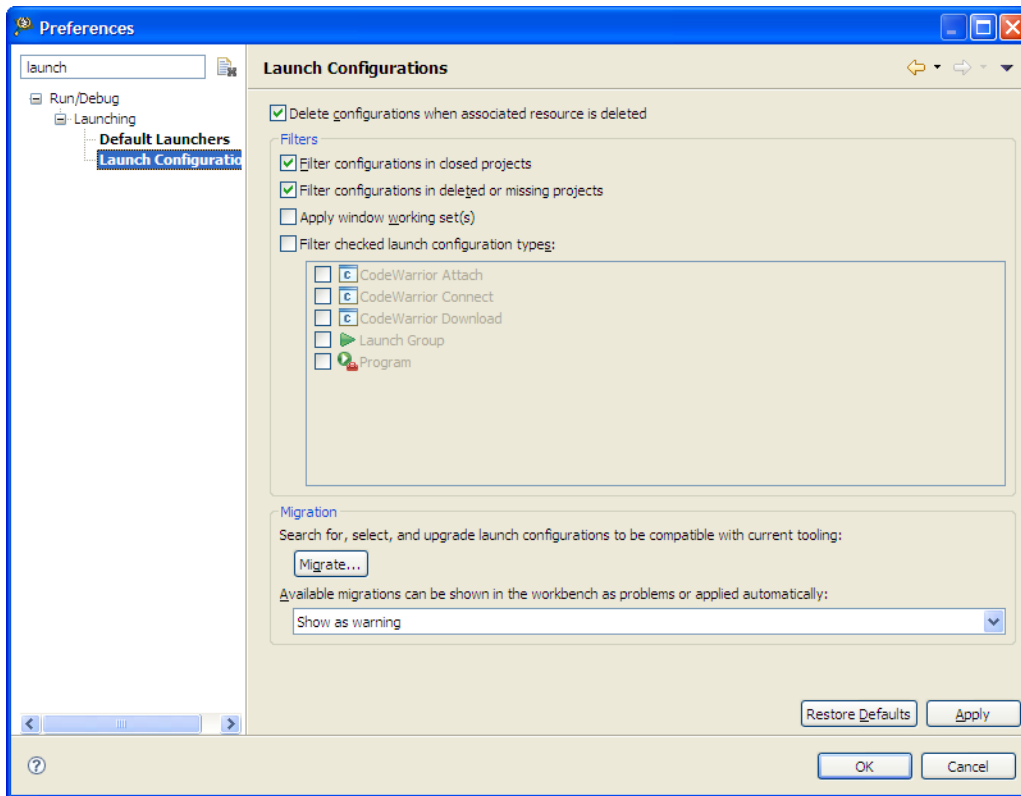


Figure 2-93. Preferences Dialog Box - Launch Configuration

4. Click **OK** .

Now, the configurations will not be deleted when the associated resource is project.

2.6.39 How can I modify and save the files in GBK encoding using the CodeWarrior IDE?

In order to be able to save these files, follow the steps given below:

1. From C/C++ perspective toolbar, select **Window > Preferences** .

The **Preferences** window appears.

2. Select **General > Workspace** .
3. Under **Text file encoding** , type **GB18030** in the **Other** textbox.
4. Click **Apply** .
5. Click **OK** .

Now, you can save the files in GBK encoding.

2.6.40 How are the PARENT-COUNT-MyVariable definitions defined in the CodeWarrior IDE?

The Linked resource variable PARENT can be used to define path variables relative to a parent directory of another one. For example, a path variable FOO can be set to `${PARENT-2-PROJECT_LOC}/foo`, which is equivalent of setting it to `${PROJECT_LOC}../../foo`. The reason of this strange syntax is for the Eclipse compatibility constraints. You can not have the `"..."` characters inside a path variable value (at the resource level).

2.6.41 Is there a way to instruct the CodeWarrior IDE to use relative paths instead of absolute ones to store the project file location in a workspace?

Yes. When you create the project in the default location which is the workspace directory, no `.location` file gets created and you just have to zip the project and the workspace. Before you create the zip file make sure that you remove the following:

- The project build directory since there will be absolute paths in the `.mk` and `.args` files.
- The workspace `.metadata\plugins\org.eclipse.ltk.core.refactoring` directory.
- The workspace `.metadata\plugins\org.eclipse.ltk.core.resources\history` directory.
- The `.PDOM` files from `.metadata\plugins\org.eclipse.cdt.core` directory.

2.6.42 What is a Working Set? How do I build/compile a Working Set?

The Working Set is a CodeWarrior eclipse IDE feature, that allows you to group elements for display in views. With that, you can do operations on a set of elements in that working set. Especially when you are using many projects the same time, working sets are a big help to focus on the right set of things at a time.

To build/compile a Working Set:

1. Select **Project > Build Working Set > Select Working Set** from the IDE menu bar.

The **Select Working Set** dialog box appears.

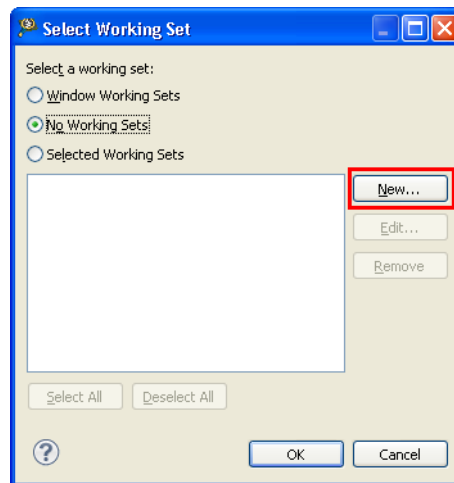


Figure 2-94. Select Working Set Dialog Box

2. If you have no Working Set defined, then click **New**.

The **New Working Set** dialog box appears.

3. Select **C/C++** and click **Next**.

The **C/C++ Working Set** page appears.

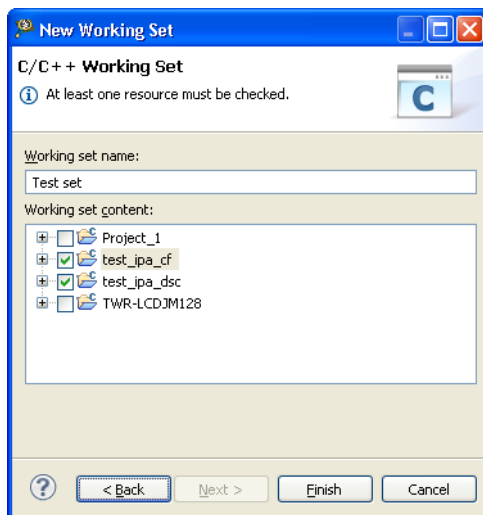


Figure 2-95. New Working Set - C/C++ Working Set Page

4. Specify a name in the **Working set name** text box, and select the desired projects that you want to include from the **Working set content** list.
5. Click **Finish**.

The **New Working Set** dialog box closes and the newly created working set appears in the **Select Working Set** dialog box.

6. Select the newly created Working Set.

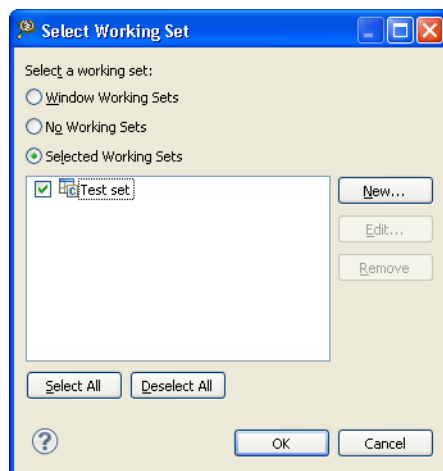


Figure 2-96. Select Working Set Dialog Box with Newly Created Working Set

7. Click **OK**.

The IDE builds/compiles the projects included in the Working Set.

2.6.43 How do I search within the Working Set?

To search within a Working Set:

1. Select **Search > Search** from the IDE menu bar. Alternatively, press **Ctrl+H**.

The **Search** dialog box appears.

2. Specify the desired settings in the **Containing text** and the **File name patterns** text boxes.

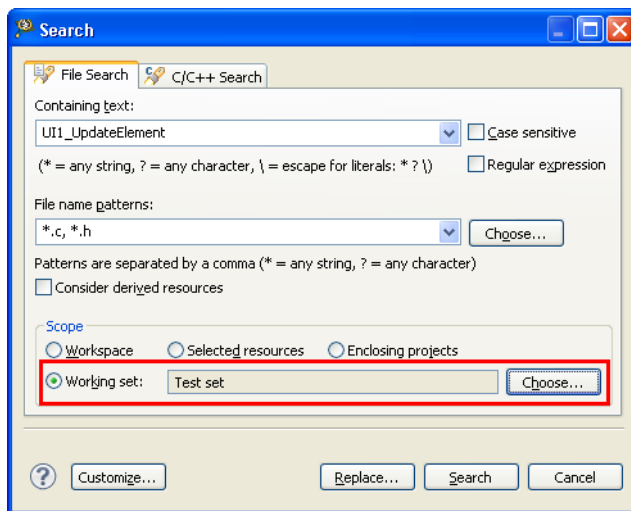


Figure 2-97. Search Dialog Box

3. Select **Working set** from the **Scope** group and click **Choose** to select the desired working set where you want the IDE to conduct the search.
4. Click **Search**.

The IDE conducts the search and lists the result in the **Search** view.

2.6.44 How do I manage the Working Sets?

To manage the Working Sets:

1. Select **Project > Build Configurations > Manage Working Sets** from the IDE menu bar.

The **Manage Working Set Configurations** dialog box appears listing the available working sets.

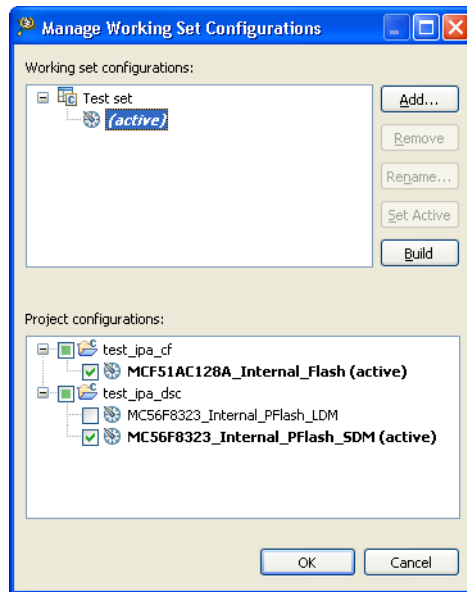


Figure 2-98. Manage Working Set Configurations Dialog Box

Using this dialog box, you can add, remove, rename, set active, and build the desired Working Sets.

2.6.45 How can I can filter the projects listed in the CodeWarrior Projects view based on Working Sets?

To filter the projects listed in the **CodeWarrior Projects** view based on working sets, you have to click the *inverted triangle* from the toolbar and select the **Select Working Set** or select one from the most recently used listed working sets.

The following figure shows the inverted triangle menu to filter the projects listed in the **CodeWarrior Projects** view based on working sets:

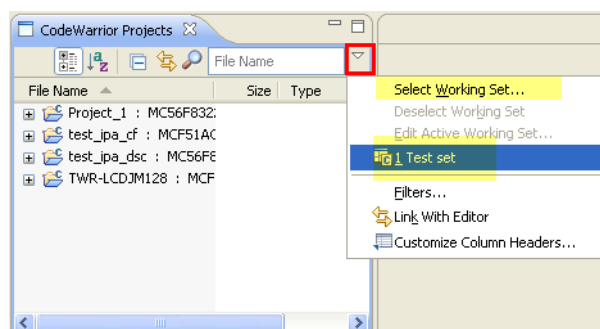


Figure 2-99. CodeWarrior Projects View - Select Working Set

2.6.46 How can I import/export a Working Set in the CodeWarrior eclipse IDE?

To import/export a Working Set in the CodeWarrior eclipse IDE, you have to install AnyEdit Tools plug-in provided by Andrey Loskutov. This plug-in allows you to:

- Import and Export of Eclipse Working Sets
- Text Conversion and Sorting actions
- Extended Compare and Replace
- Save Console View
- Open File under Cursor

To install AnyEdit Tools plug-in:

1. Select **Help > Install New Software** from the IDE menu bar.

The **Install** dialog box appears.

2. Enter the following web address in the **Work with** textbox and press *Enter*:

`http://andrei.gmxhome.de/eclipse/`

The available plug-in for various eclipse versions appears in the list.

3. Expand the tree-control of the desired eclipse version and select **AnyEditTools**.

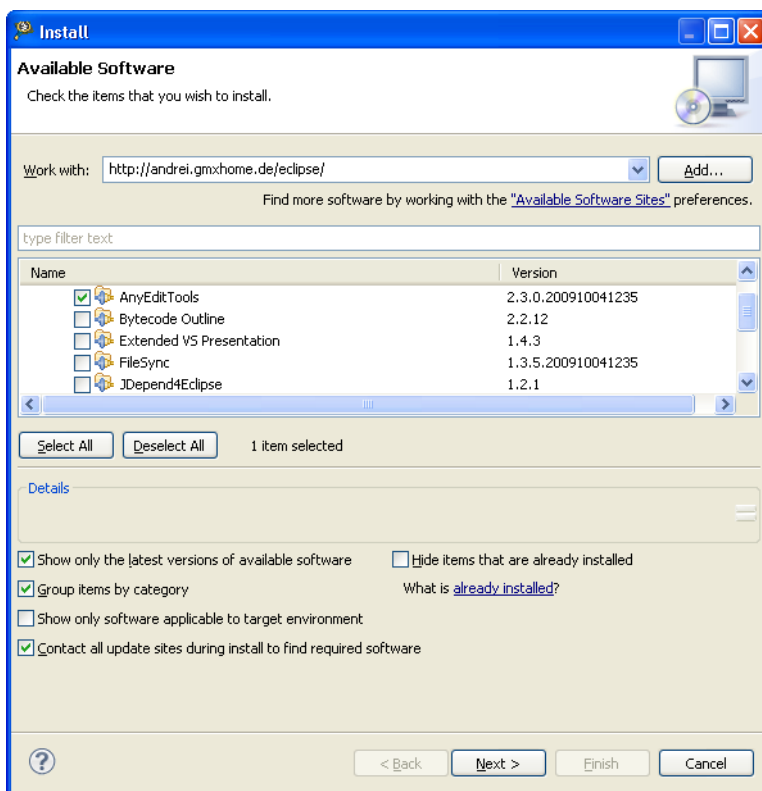


Figure 2-100. Install Dialog Box - AnyEditTools Plug-in

4. Click **Next**.

The **Install details** page appears.

5. Select **AnyEditTools** from the list.

6. Click **Next**.

7. Select the option to accept the license agreement.

8. Click **Finish**.

The IDE installs the AnyEdit Tools for your CodeWarrior Software.

The AnyEdit Tools includes an importer and exporter File action for Working Sets. To export and import the Working Sets:

- Select **File > Export** from the IDE menu bar and follow the onscreen instructions to export a working set. The following figure shows the **Export** dialog box with **Working Sets** option:

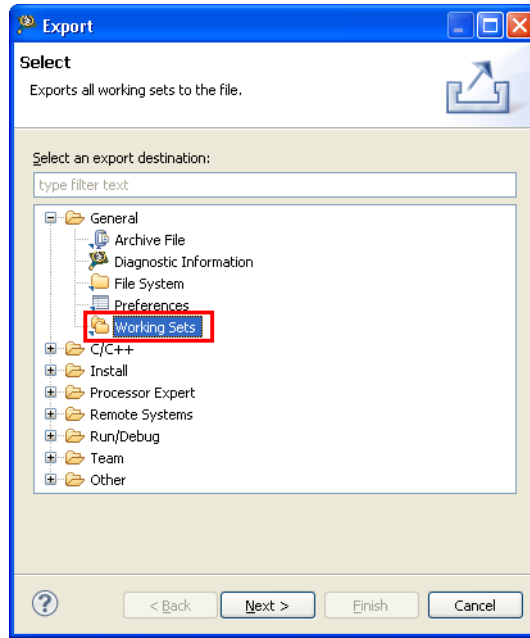


Figure 2-101. Export Dialog Box with Working Sets Option

- Select **File > Import** from the IDE menu bar and follow the onscreen instructions to import a working set. The following figure shows the **Import** dialog box with **Working Sets** option:

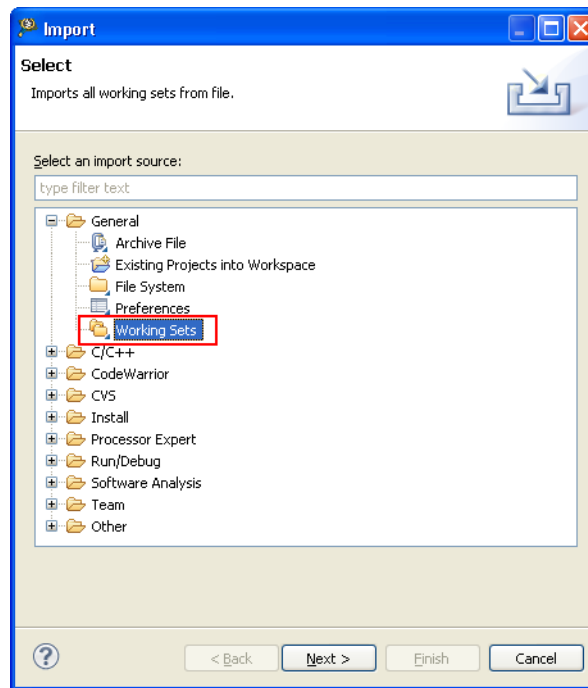


Figure 2-102. Import Dialog Box with Working Sets Option

2.6.47 What are Icon decorators?

The icon and label decorators are the symbols used for annotating the icon or label with the useful information. The following figures shows the example of source file icon decorators:

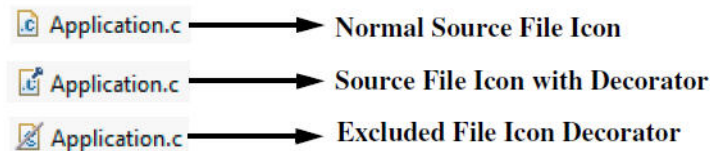


Figure 2-103. Example - Decorator for Application.c File

2.6.48 How do I enable or disable an icon decorator?

To enable/disable an icon decorator:

1. Select **Window > Preferences** from the IDE menu bar.

The **Preferences** dialog box appears.

2. Expand the **General** tree control and select **Label Decorations** from the left panel.

The **Label Decorations** page appears in the right panel. The following figure shows the **Preferences** dialog box with **Label Decorations** page:

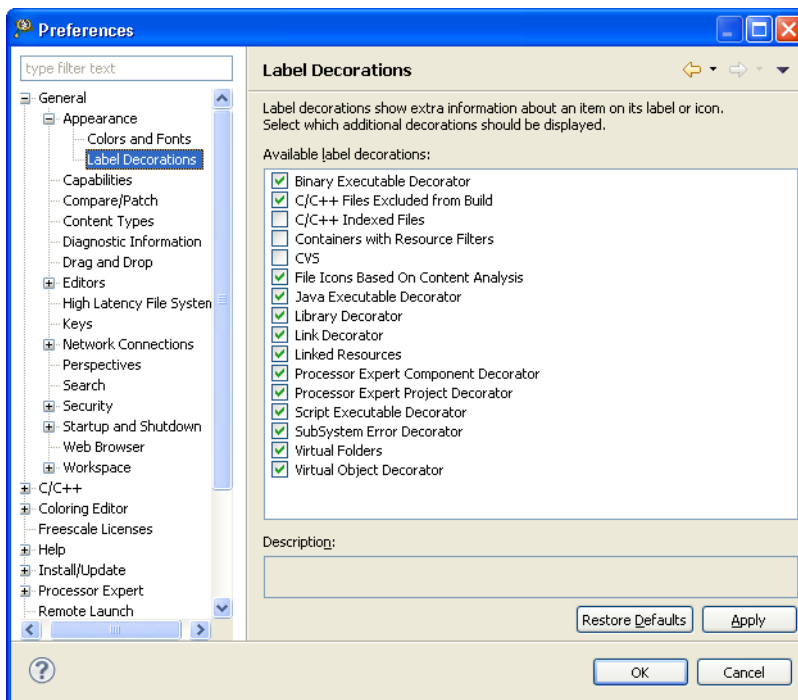


Figure 2-104. Preferences Dialog Box - Label Decorations Page

3. Check or clear the checkboxes for the desired label decorators to enable or disable the decorators respectively.
4. Click **Apply** to save the changes.
5. Click **OK** to close the dialog box.

The IDE applies and saves the changes you selected for the icon decorations.

2.6.49 Can the icon decorators be combined?

Yes, the icon decorators can be combined. The following figure shows an example:

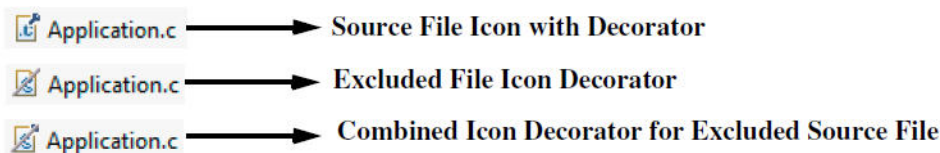


Figure 2-105. Example - Combined Icon Decorators

2.6.50 When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, "You are opening a large file. Scalability mode has been turned

on for this editor to help improve performance by disabling features requiring to parse the source file language." What does it mean?

While processing a file in the editor, there are various other processes that runs in the background, this includes *Eclipse Indexer*, *Outline View* or *Syntax Coloring*.

When you open a large file, the IDE tends to get slower to use more memory to process that file. To avoid this, the IDE turns on the *Scalability Mode*, and close the functions like, *Eclipse Indexer*, *Outline View* or *Syntax Coloring* to increase the performance.

2.6.51 How do I configure the Scalability settings?

You can configure the Scalability Mode to be enabled only for very large files. To configure the **Scalability** settings:

1. Select **Window > Preferences** from the IDE menu bar.

The **Preferences** dialog box appears.

2. From the left panel, expand the **C/C++ > Editor** tree controls and select **Scalability** .

The **Scalability** page appears in the right panel.

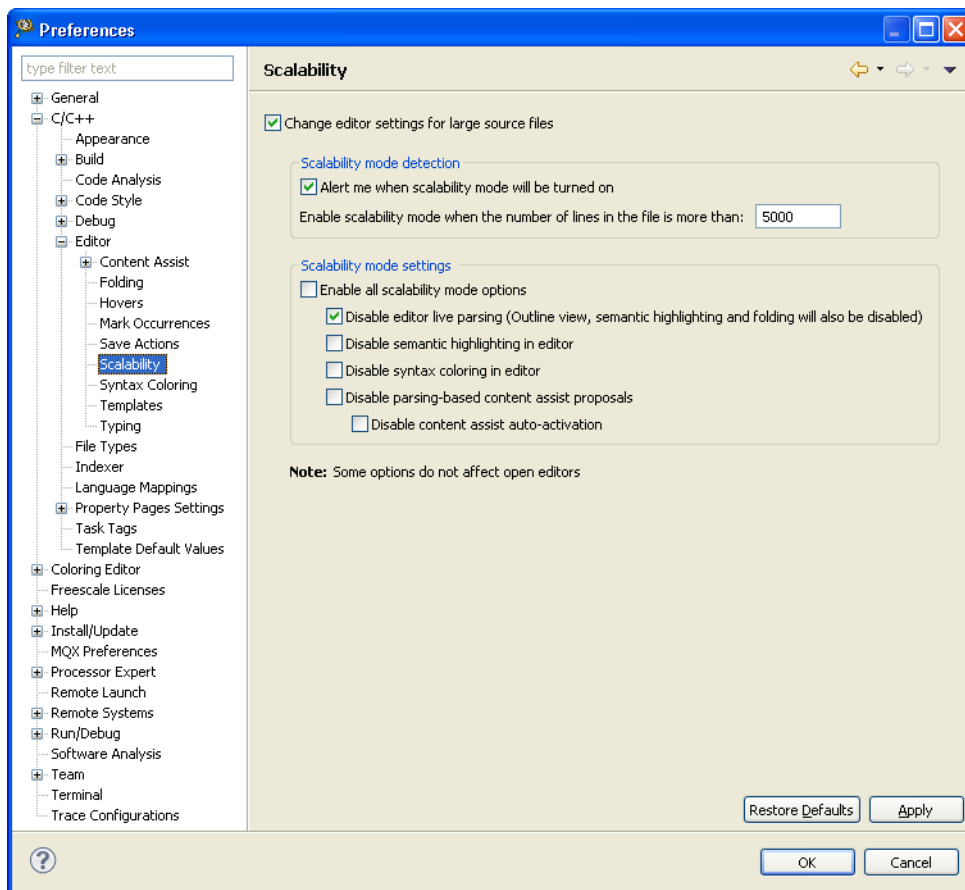


Figure 2-106. Preferences Dialog Box - Scalability Page

3. Specify the desired settings in the **Scalability mode detection** and the **Scalability mode settings** groups.
4. Click **Apply** .
5. Click **OK** .

The **Preferences** dialog box closes. The IDE saves and applies the changes you specified.

Chapter 3

Project Management

In this chapter, you find Frequently Asked Questions (FAQs) related to project management in CodeWarrior IDE.

- [FAQs - Project Management](#)

3.1 FAQs - Project Management

This topic lists the project management related FAQs.

- [How can I find out where the source files related to a CodeWarrior project are saved?](#)
- [How can I add another source or header file to my project?](#)
- [What is a workspace?](#)
- [What is a project?](#)
- [How can I view the contents of various types of source code files that appear in my project folder?](#)
- [Is there a way to compare the Property Settings of two different projects?](#)
- [How do I resize the various views in a Perspective?](#)
- [How can I add a project from some other workspace to my current workspace?](#)
- [How do I ensure that the CodeWarrior IDE saves all modifications that I make to the project properties?](#)
- **NEW** [While working with the CodeWarrior IDE, the Remote System Changed dialog box appears. Why does it appear and how can I resolve the issue?](#)

3.1.1 How can I find out where the source files related to a CodeWarrior project are saved?

To know where your source file is located, follow these steps:

1. In the C/C++ project window, right-click on source file.
A context menu appears.
2. In the context menu, select **Show in Windows Explorer** .
The Window Explorer window appears.

Alternatively, follow these steps:

1. In the C/C++ project window, right-click on source file.
A context menu appears.
2. In the context menu, select **Properties** .
The **Properties** window appears.
3. Select **Resource** .

The **Location** field displays the full path of the folder.

NOTE

CodeWarrior always warns the user before performing a delete operation and describes whether it is about to delete just the link, or to delete the file or folder permanently.

3.1.2 How can I add another source or header file to my project?

To add a source file or header to project follow steps below:

1. In the CodeWarrior Project window, right-click on the folder where you want the new source file or header to be added.
A context menu appears.
2. From the context menu, select **New > Source File** .
The **New Source File** window appears.
3. In the **Source File** textbox, enter the name of the source file.
4. Click **Finish** .

The source file gets added to your project.

NOTE

Another way to accomplish is to drag and drop the file from Windows Explorer. A copy of the file is added to the project's workspace. If you make changes to the source file, the original source file is not modified. Only source file in the project is modified.

3.1.3 What is a workspace?

A workspace is a directory for your work. This is where your project is located. The name and location of the workspace is specified when you start the CodeWarrior IDE.

3.1.4 What is a project?

A project is a container for organizing files and folders related to a specific area.

3.1.5 How can I view the contents of various types of source code files that appear in my project folder?

To view the contents of source files, follow steps below:

1. From the CodeWarrior Projects window, right-click on the source file.

A context menu appears.

2. From the context menu, select **Open With > Text Editor** .

The text editor window appears showing the contents of the file.

Tip

Double-click on the source file to view its contents in the **C/C++ Editor** .

3.1.6 Is there a way to compare the Property Settings of two different projects?

Currently the way to do this is to do a text compare of the .project files. To do this, follow steps below:

1. In the CodeWarrior Projects folder, click on .cproject.
The .cproject is highlighted.
2. Press the **Ctrl** key.
3. In the CodeWarrior Projects folder, click on second .cproject.
The second .cproject is highlighted.
4. Right-click on .cproject.
A context menu appears.
5. From context menu, select **Compare With > Each Other** .
The **Compare** window appears.

You can compare property settings of the projects using the **Compare** window.

3.1.7 How do I resize the various views in a Perspective?

To resize the various view in a Perspective, click on the dividers (gray bands) between the different views and drag them with the mouse.

3.1.8 How can I add a project from some other workspace to my current workspace?

In order to add a project from another workspace follow steps below:

1. Open the workspace in Windows Explorer.
2. Select the project folder and drag it over to the CodeWarrior IDE.

The IDE effectively handles the files and folders dropped to the workbench. A link to the existing project is created in the **CodeWarrior Project** view.

3.1.9 How do I ensure that the CodeWarrior IDE saves all modifications that I make to the project properties?

The settings specified in the Properties dialog box are saved when you click **Apply** and then **OK** to close the dialog box. The only reason this does not work could be:

- You only have read access to the `.project` file in the project directory.

-or-

- You closed the **Properties** dialog box by clicking **Cancel** instead of **OK**.

3.1.10 While working with the CodeWarrior IDE, the Remote System Changed dialog box appears. Why does it appear and how can I resolve the issue?

The projects that reference s Explorer (RSE) Systems contain an internal cache of the referenced RSE Systems. When such projects are imported in the workspace, the RSE settings in the cache may differ from the actual RSE settings in the workspace. Or the RSE settings may be missing entirely from the project.

In such situations, the **Changed** dialog box appears.

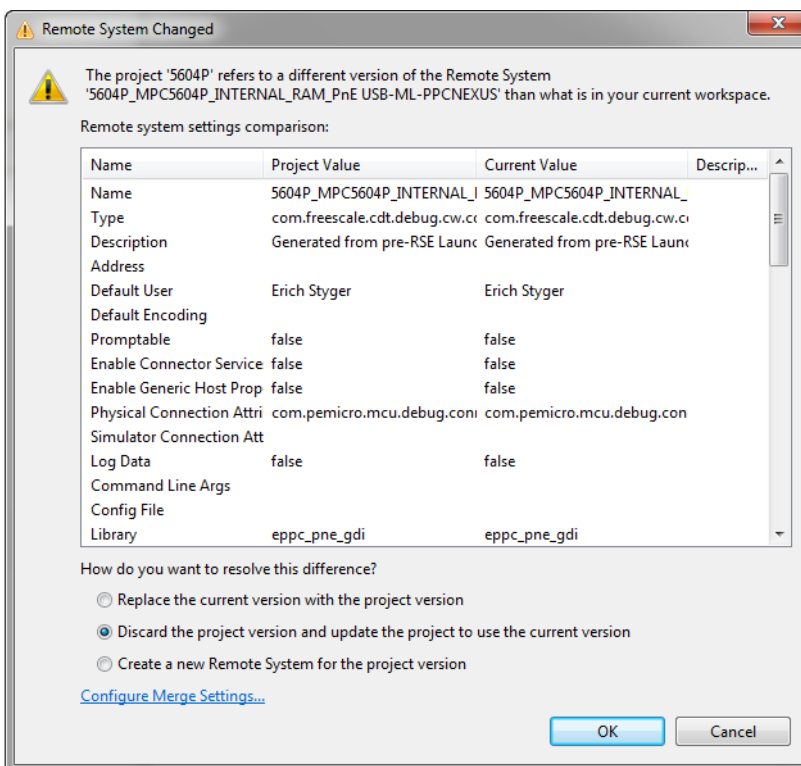


Figure 3-1. Changed Dialog Box

The RSE settings are stored in the `.metadata` workspace folder as a part of the project settings and in the `rseHostSettingsCache.xml` file (refer to the image listed below), as the cached version, in the project.

As with any cache, the cache might be out of sync or needs to be flushed, which causes a mismatch between the RSE settings in the cache and the actual RSE setting in the project.

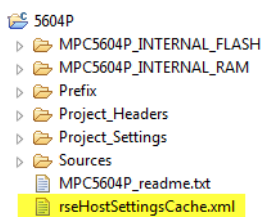


Figure 3-2. RSE Settings

To resolve the issue, you need to configure the merge settings.

To configure IDE's response when an RSE system mismatch is detected, perform the following steps:

1. Click **Configure Merge Settings** in the **s Changed** dialog box.

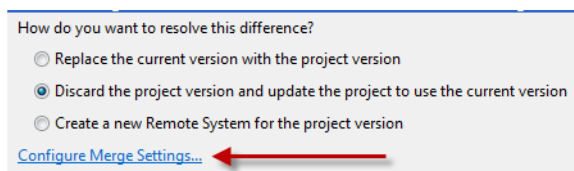


Figure 3-3. Configure Merge Settings

Alternatively, select **Window > Preferences** from the CodeWarrior menu bar. In the **Preferences** dialog box, select **s > Merge**.

The **Merge** page appears in the **Preferences** dialog box.

2. Select the appropriate option and click **Apply**.

The merge settings are configured.

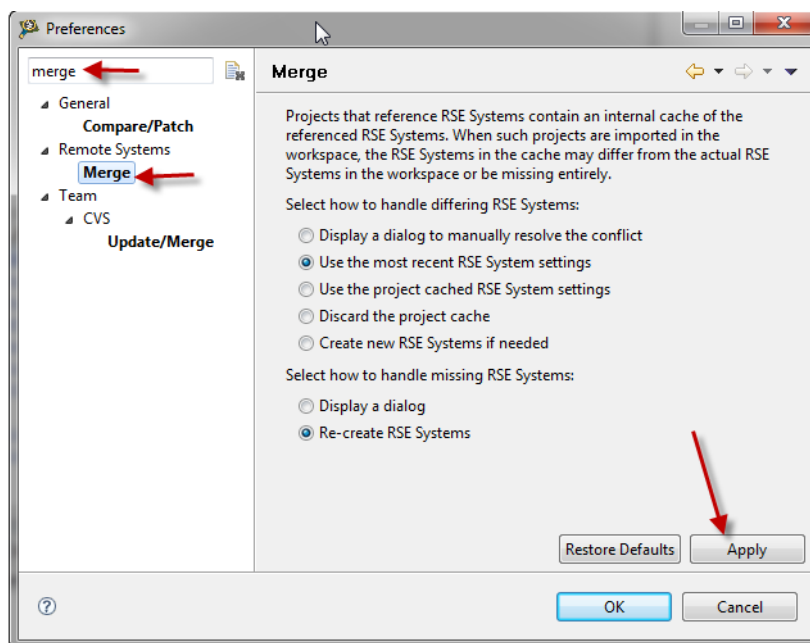


Figure 3-4. Merge Settings | Preferences Dialog Box



Chapter 4

Project

In this chapter you find Frequently Asked Questions (FAQs) related to projects in the CodeWarrior IDE.

- [FAQs - Project](#)

4.1 FAQs - Project

This topic lists the project related FAQs.

- [How do I add a new Source Folder to my project?](#)
- [How can I add to my project a folder that exists outside the current workspace?](#)
- [How do I add to my project a new file that exists outside the current workspace?](#)
- [How can I export a project that I could send to someone else?](#)
- [How do I instruct the CodeWarrior IDE to save changes to my source code files automatically before building the project?](#)
- [How do I compare two source code files?](#)
- [How do I compare two directories?](#)
- [How can I specify the number of days to keep local history for source code files?](#)
- [Is there a shortcut to remove object code that is equivalent to the Ctrl + - shortcut key available in the Classic CodeWarrior IDE?](#)
- **NEW** [While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?](#)
- **NEW** [While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?](#)
- **NEW** [How do I enable the incremental build on my project?](#)
- **NEW** [How do I filter the items listed in the Problems view based on the project selected?](#)

4.1.1 How do I add a new Source Folder to my project?

To add a new source folder to a project follow steps below:

1. In the CodeWarrior Project window, right-click on project name.
A context menu appears.
2. From context menu, select **New > SourceFolder** .
The **New Source Folder** window appears.
3. In the folder name textbox, enter the name for new folder.
4. Click **Finish** .

The new source folder appears in window.

4.1.2 How can I add to my project a folder that exists outside the current workspace?

To add files to an existing project, follow steps below:

1. In the CodeWarrior Project window, right-click on project name.
A context menu appears.
2. From context menu, select **New > Folder** .
The **New Folder** window appears.
3. Click **Advanced** .
The Advanced Settings appear.
4. Select **Link to folder in the file system** checkbox.

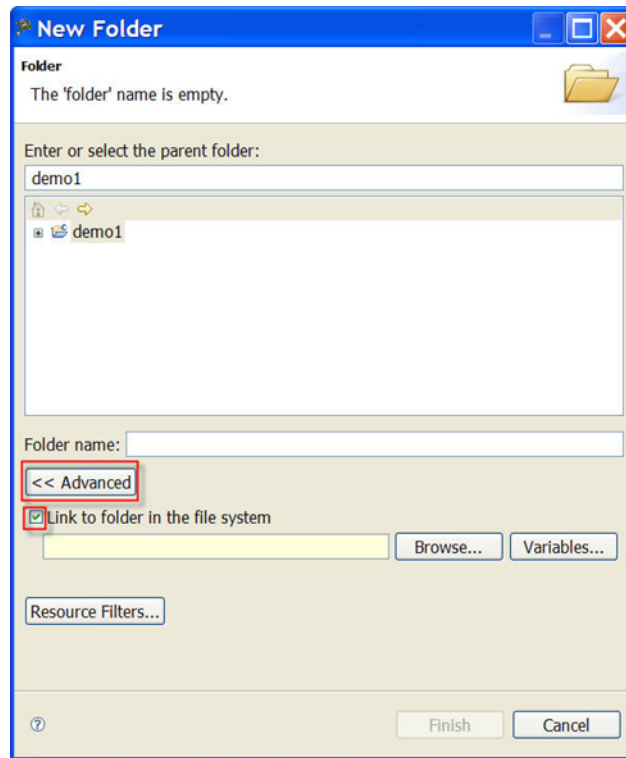


Figure 4-1. New Folder

5. Click **Browse** .

The **Browse for Folder** window appears.

6. Select the folder that you want to add to the project.

7. Click **OK** .

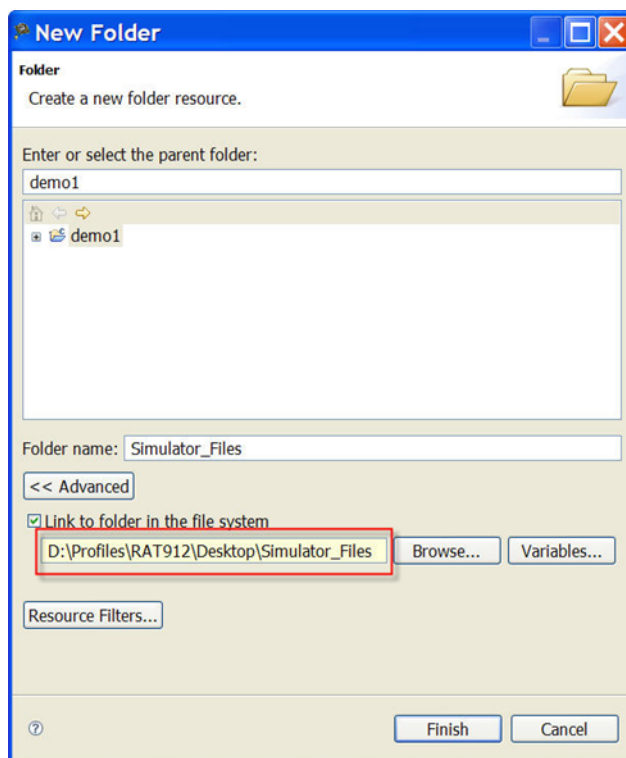


Figure 4-2. New Folder

8. Click **Finish** .

This will put the new folder into the project under CodeWarrior projects. When the new folder is expanded, it will point to the location where the folder is linked from. If you delete the whole folder from the project, it will be deleted from the project only. If you try to delete a file from this folder, it will be deleted from that location leaving no way to recover it.

4.1.3 How do I add to my project a new file that exists outside the current workspace?

To add a specific file to an existing project, drag the file that you want to add from the Windows Explorer into the CodeWarrior Projects.

4.1.4 How can I export a project that I could send to someone else?

To export a project follow steps below:

1. In CodeWarrior Project window, right-click on project name.

A context menu appears.

2. From context menu, select **Export** .

The **Export** window appears.

3. Expand **General Folder** .
4. Select **Archive File** .
5. Click **Next** .

The **Archive File** window appears.

NOTE

By default, the project that you select will be exported along with their children. Optionally, use the checkboxes in the left and right panes to select the set of resources to export. Also, use radio buttons, such as **Select Types** to filter the types of files that you want to export.

6. In the **To Archive File** textbox, enter the archive file which you want to export the selected resources to.
7. Click **Finish** .

A zip file is created in the **Eclipse** folder within the **Tools Installation** directory.

4.1.5 How do I instruct the CodeWarrior IDE to save changes to my source code files automatically before building the project?

To instruct the CodeWarrior IDE to save changes to your source code files automatically before building the project, follow steps below:

1. From the C/C++ perspective toolbar, select **Window > Preferences** .

The **Preferences** window appears.

2. Expand **General** .
3. Select **Workspace** .

The **Workspace** settings appear.

4. Under **Save required dirty editors before building** , select the **Always** radio button.

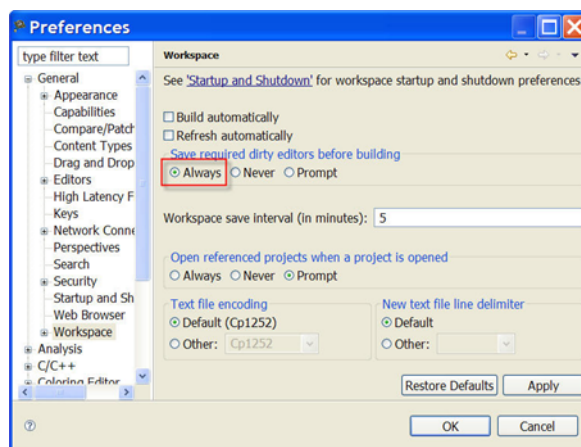


Figure 4-3. Preferences-Workspace Page

5. Click **Apply** .
6. Click **OK** .

Now, the CodeWarrior IDE will save changes to the source code files automatically before building the project.

4.1.6 How do I compare two source code files?

In order to compare two source files, follow steps below:

1. In the CodeWarrior Projects folder, click on source file.
The source file is highlighted.
2. Press the **Ctrl** key.
3. In the CodeWarrior Projects folder, click on second source file.
The source file is highlighted.
4. Right-click on source file.
A context menu appears.
5. From context menu, select **Compare With > Each Other** .
The **Compare** window appears.

You can compare the source files using the **Compare** window.

4.1.7 How do I compare two directories?

In order to compare two directories follow steps below:

1. In the CodeWarrior Projects folder, click on directory.

The directory folder is highlighted.

2. Press the **Ctrl** key.
3. In the CodeWarrior Projects folder, click on second directory folder.

The directory folder is highlighted.

4. Right-click on directory folder.

A context menu appears.

5. From context menu, select **Compare With > Each Other** .

The **Compare** window appears.

You can compare the directories files using the **Compare** window.

4.1.8 How can I specify the number of days to keep local history for source code files?

To change the history length for the C source editor, follow steps below:

1. From the C/C++ perspective toolbar, select **Window > Preferences** .

The **Preferences** window appears.

2. Select **General > Workspace > Local History** .
3. The **Local History** settings appear.
4. In the **Days to Keep Files** textbox, change the days to keep files value.
5. Click **Apply** .
6. Click **OK** .

The history length for the C source editor is updated.

4.1.9 Is there a shortcut to remove object code that is equivalent to the Ctrl + - shortcut key available in the Classic CodeWarrior IDE?

No. However, if you want to assign the **Ctrl+-** shortcut key for removing the object code, then you can manually assign it by changing the key binding. For more information on how to assign or change a key binding, see [How can I change a key binding?](#)

When you change the key binding, a dialog box appears asking you which project you want to clean, and if you want to automatically start the build thereafter. If you do not want this dialog box to appear, then you can select the **Build Automatically** checkbox under **Window > Preferences > General > Workspace Settings** .

4.1.10 While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?

The CodeWarrior software has the following default settings.

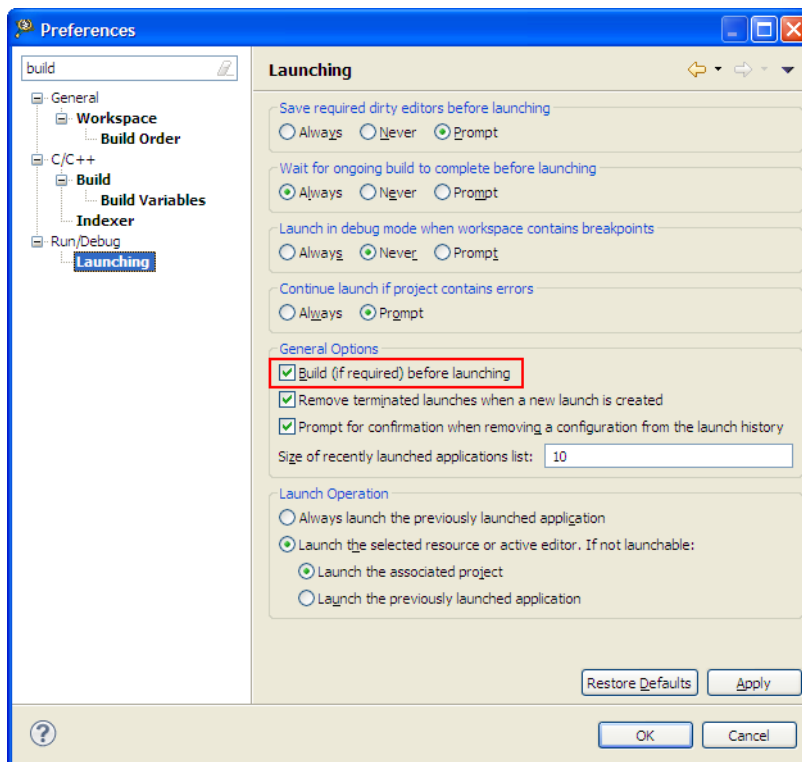


Figure 4-4. Build Before Launching

And, the launch configuration refers the project and not the build target of the project.

This is why the CodeWarrior software automatically build the current active build target.

To ensure that the CodeWarrior software builds the right target in CodeWarrior for Power Architecture:

- Make sure you have the right build set as active, or
- Or, clear the **Build (if required) before launching** checkbox, and build the project yourself before launching.

4.1.11 While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?

The `DE.EXE` might locks the file. The `DE.EXE` (Debugger Engine Executable) locks the file during debugging, but might have failed to release it. To release the lock, simply close (kill) the `DE.EXE` application from the task manager.

4.1.12 How do I enable the incremental build on my project?

Follow the steps given below to perform incremental build on your project:

1. In the CodeWarrior Project window, right-click on project name.

A context menu appears.

2. From the context menu, select Properties.

The Properties window appears.

3. Select C/C++ Build.

The C/C++ Build page appears in the right panel of the Properties dialog box.

4. Select the **Behaviour** tab.

5. Check the **Build (Incremental build)** checkbox to build your project incrementally.

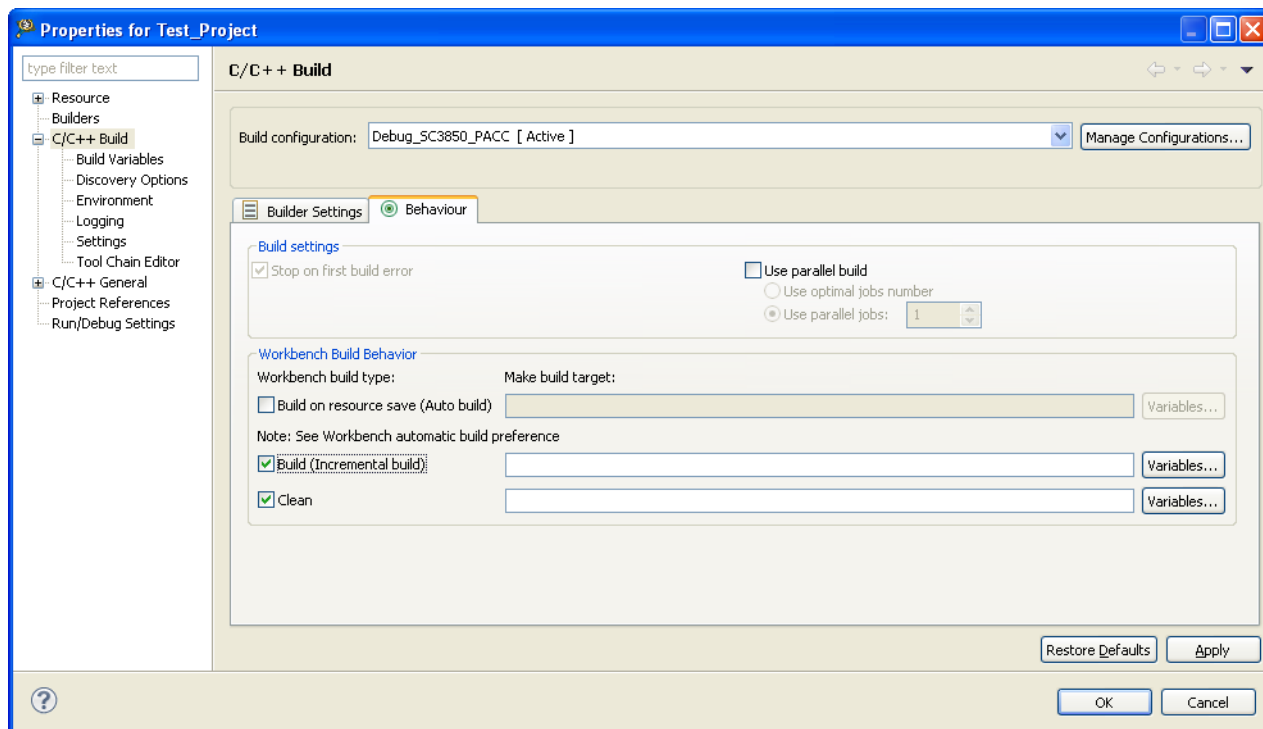


Figure 4-5. Properties Dialog Box-C/C++ Build Page

6. Click OK.

Also, you need to check if there are some pre-build steps associated with your build configuration. If yes, then the CodeWarrior will always perform a full rebuild. Follow the steps given below to check if there are any pre-build steps associated with your project.

1. In the CodeWarrior Project window, right-click on project name.

A context menu appears.

2. From the context menu, select Properties.

The Properties window appears.

3. Select C/C++ Build > Settings.

The Settings options appear on right hand side of properties window.

4. Click the **Build Steps** tab.

5. Check if pre-build steps are defined in the **Command** field.

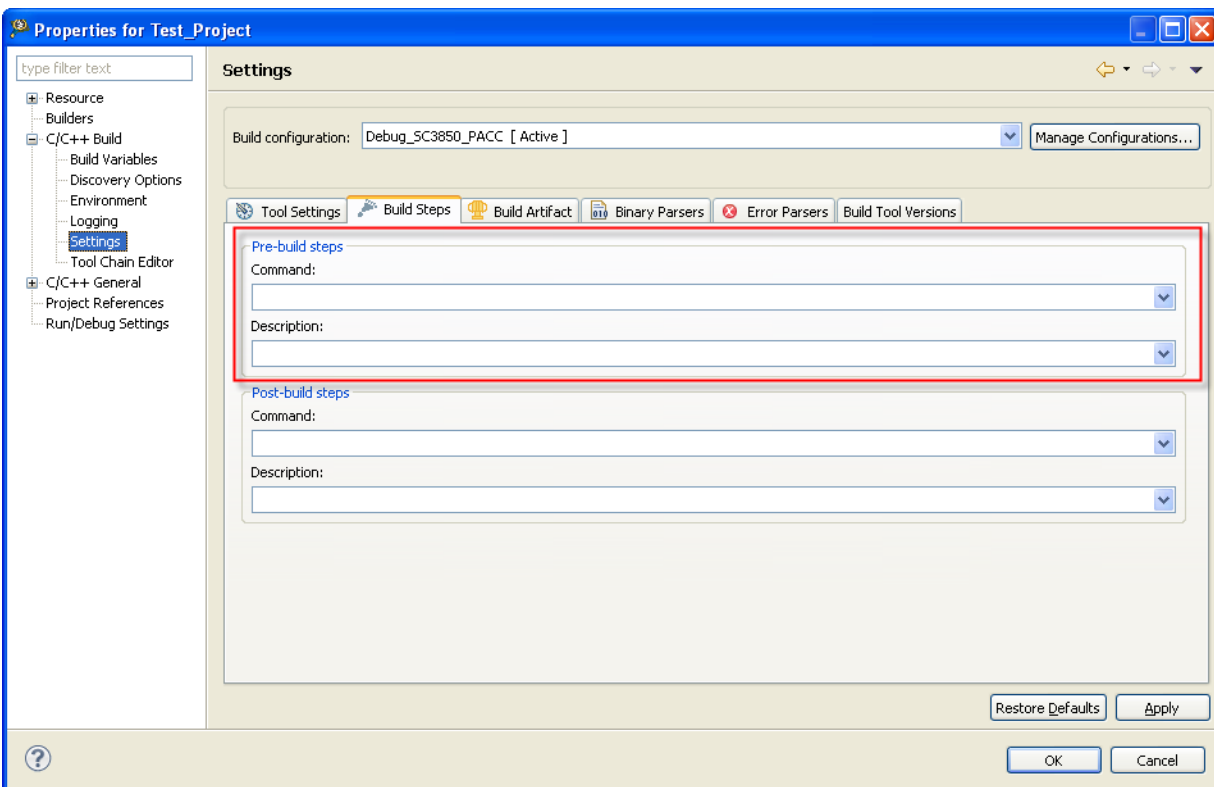


Figure 4-6. Properties Dialog Box-Settings Page

6. Click Ok.

4.1.13 How do I filter the items listed in the Problems view based on the project selected?

To filter the items listed in the **Problems** view based on the project selected:

1. Select the inverted triangle from the the top left corner of the **Problems** view.

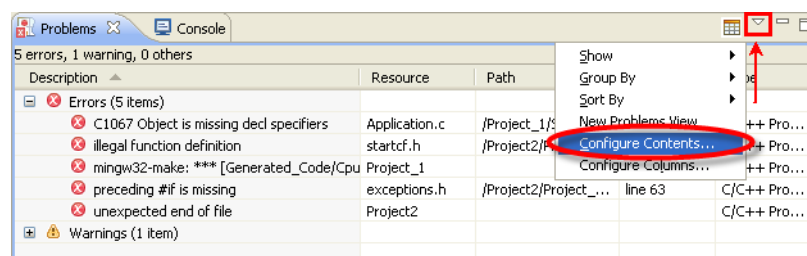


Figure 4-7. Problems View

2. Select **Configure Contents** from the menu.

The **Configure Contents** dialog box appears.

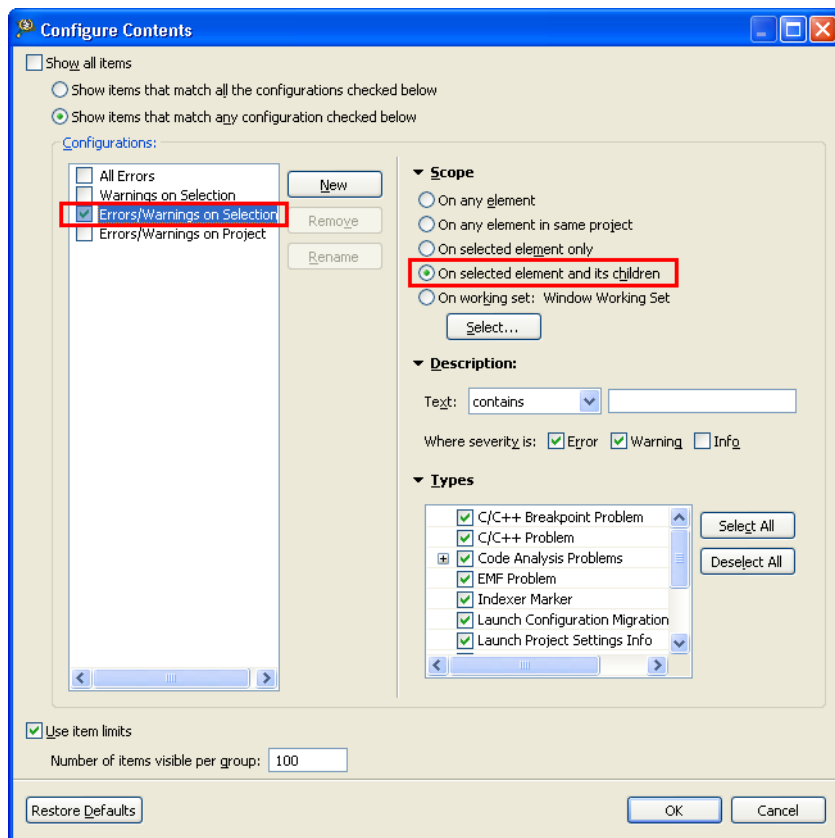


Figure 4-8. Configure Contents Dialog Box

3. Check the **Errors/Warnings on Selection** checkbox from the **Configurations** list in the left panel.
4. Make sure that the **On selected element and its children** option is selected from the **Scope** list in the right panel.
5. Click **OK** .

The **Problems** view lists the problems related to the selected project only.

Chapter 5 Compiler

In this chapter, you find Frequently Asked Questions (FAQs) related to compiler settings in the CodeWarrior IDE.

- [FAQs - Compiler](#)

5.1 FAQs - Compiler

This topic lists the compiler related FAQs.

- [How do I compile my project?](#)
- [Where do I see the warning and error messages?](#)
- [When I compile my project, where can I see the commands that are being executed?](#)

5.1.1 How do I compile my project?

In order to compile the project, follow steps below:

1. From CodeWarrior Projects, right-click on project name.

A context menu appears.

2. From context menu, select **Clean Project** .

If **Build Automatically** is checked, then when **Clean Project** executes, it creates a **Binaries** folder that contains the *.abs file that are used for debugging. If **Build Automatically** is unchecked, then **Clean Project** removes any previously-built binaries, including the *.abs file.

3. Select **Project > Build Project** to compile.

The project compiles.

5.1.2 Where do I see the warning and error messages?

The warning and errors messages appear in the **Problems** view. To display the **Problems** views select **Window > Show View > Problems** . The **Problems** view appears docked to the lower part of the screen. Double click on error message in **Problems** view to navigate to the relevant source code. The **Problems** view shows problems for all open projects. If there is more than one open project, the window may show problems not associated with the project you are building. Compiler and linker error message are also shown in the **Problems** view.

5.1.3 When I compile my project, where can I see the commands that are being executed?

The commands that are being executed from the compiler or linker can be viewed in the console view. From the IDE menu bar, select **Window > Show View > Console** . The **Console** view appears docked to the lower part of the screen.

Chapter 6

Linker

In this chapter, you find Frequently Asked Questions (FAQs) related to the linker settings in the CodeWarrior IDE.

- [FAQs - Linker](#)

6.1 FAQs - Linker

This topic lists the linker related FAQs.

- [Can I use a different linker command file than the one being used in my project?](#)

6.1.1 Can I use a different linker command file than the one being used in my project?

Yes, the linker command file can be changed by following the steps given below:

1. In the CodeWarrior Projects window, right-click on project name.

A context menu appears.

2. From context menu, select **Properties**.

The **Properties** window appears.

3. Select **C/C++ Build > Settings**.

The **Settings** options appear on right hand side of properties window.

4. Select **Power Linux Linker > Power Environment**.

The **Power Environment Settings** appear.

5. The **LCF File** text box specifies the linker command file that is currently being used.
6. Click **Browse** to select a different linker command file.
7. Click **Apply**.
8. Click **OK**.

The linker command file changes, as specified.

Chapter 7

Debugger

In this chapter, you find Frequently Asked Questions (FAQs) related to the debugger settings in the CodeWarrior IDE.

The FAQs listed in this chapter are divided into the following categories:

- [Breakpoints](#)
- [NEW Variables](#)
- [Build Configuration](#)
- [Flash Programmer](#)
- [Miscellaneous](#)

7.1 Breakpoints

In this topic, FAQs related to breakpoints are listed.

- [How do I set breakpoints?](#)
- [How do I view the breakpoints that have been set?](#)
- [How do I disable existing breakpoints?](#)
- [What does the checkmark next to the breakpoint represent?](#)
- [What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options?](#)
- [Which breakpoint option do I select if I want to add a breakpoint?](#)
- [Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list?](#)
- [What if I want to delete a breakpoint?](#)
- [How do I place a breakpoint on a given physical address?](#)
- [How do I add hardware breakpoints? Can I add them through the GUI?](#)
- [How can I view the hardware breakpoints that have been set?](#)

- What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other?
- What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window?
- Is there a way to add breakpoints to C and Assembly files that exist in a library?
- How do I set a breakpoint at a particular function?
- What is the difference between a breakpoint and a line breakpoint?
- The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing?
- **NEW** While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do?
- **NEW** Can I use software breakpoints on code running in flash memory?
- **NEW** What is a hardware breakpoint?
- **NEW** How do I select the type of breakpoint for my application in the Editor view?

7.1.1 How do I set breakpoints?

To set breakpoints in a source or assembly file, follow one of the four methods given below.

1. Right-click on the left-hand side of source file (the Gray bar)

A context menu appears.

2. Select **Toggle Breakpoint** at the line where you want the breakpoint.

-or-

1. Select **Ctrl+Shift+B** at the line where you want the breakpoint.

-or-

1. Double-click on the line where you want the breakpoint

-or-

1. Click on the line where you want the breakpoint.
2. From the IDE menu bar, select **Run > Toggle Breakpoint** .

A breakpoint appears on the left hand side of source code file.

7.1.2 How do I view the breakpoints that have been set?

To view the breakpoints that have been set, select **Window > Show View > Breakpoints** from the IDE menu bar. The **Breakpoints** window appears docked to the upper right-hand side of debugger perspective. The breakpoints window displays the breakpoints that you have enabled.

7.1.3 How do I disable existing breakpoints?

To disable existing breakpoints:

1. Click on the line that has the breakpoint enabled.
2. From the IDE menu bar, select **Run > Toggle Breakpoint**.

-or-

- Press **Ctrl+Shift+B** at the source line where you want the breakpoint disabled.

-or-

- Double-click the line where you want the breakpoint disabled.

-or-

1. Right-click on the left-hand side of source file (the Gray area)
A context menu appears.
2. Select **Toggle Breakpoint**.

-or-

1. Select the breakpoint that you want disabled in the **Breakpoints** window.
2. Right-click the breakpoint.

A context menu appears.

3. Select **Disable**.

The Breakpoint gets disabled.

7.1.4 What does the checkmark next to the breakpoint represent?

The checkmark next to a breakpoint indicates that the debugger has resolved the breakpoint and will be hit by the debugger. If the breakpoint does not have a checkmark, the debugger will not hit it.

7.1.5 What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options?

The **Toggle Breakpoint** option adds or removes breakpoints to the source and breakpoint list. The **Disable/Enable** option makes breakpoints either active or inactive. New breakpoints cannot be added using the **Disable/Enable** breakpoint option.

7.1.6 Which breakpoint option do I select if I want to add a breakpoint?

To add a breakpoint, select **Toggle Breakpoint**.

7.1.7 Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list?

If you want to skip a breakpoint, but still keeping it in your breakpoint list, then select the **Disable Breakpoint** option.

7.1.8 What if I want to delete a breakpoint?

If you want to delete a breakpoint from the source code, select the **Toggle Breakpoint** option.

7.1.9 How do I place a breakpoint on a given physical address?

To add a breakpoint on a given physical address, follow steps below.

1. In the CodeWarrior projects window, right-click on project name.

A context menu appears.

2. From context menu, select **Debug As > Debug Configurations** .

The **Debug** window appears.

3. Underneath **CodeWarrior Download** , select project name.

4. Select **Debug** .

The **Debug** perspective appears.

5. In **Debug** perspective, select **Window > Show View > Debugger Shell** .

The **Debugger Shell** window appears.

6. In **Debugger Shell** , type `bp <address>`.

This will set breakpoint at the `<address>` memory location.

7.1.10 How do I add hardware breakpoints? Can I add them through the GUI?

You can add hardware breakpoints through the GUI. To add them through the GUI, follow steps below:

1. Right-click on the left hand side of source file (the Gray bar).

A context menu appears.

2. From context menu, select **Set Special Breakpoint > Hardware**.

The Hardware breakpoint appears next to source line.

7.1.11 How can I view the hardware breakpoints that have been set?

To view the hardware breakpoints, select **Window > Show View > Breakpoints** from the IDE menu bar. The **Breakpoints** window appears.

7.1.12 What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other?

A software breakpoint inserts a debug instruction into the code, so it only works in the writable memory. If you are working in ROM, then you have to use a hardware breakpoint. The hardware breakpoints make use of the hardware debug support using OCE.

7.1.13 What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window?

This feature is basically a toggle button in the Breakpoint window that causes a new breakpoint (or a watchpoint) to get set only on the active debug context. This feature lets you set breakpoints at certain locations in different source file instead of same breakpoints being set in the same files. The behavior that happens right now is if you set a breakpoint in one file that is being shared by multiple cores it will set that breakpoint in the same file for all cores. With this feature you will be able to set different breakpoints in different files.

If no debug context exists, then the breakpoint is installed in all contexts as normal. Note that once set, the behavior of the breakpoint is the same as the existing filtering behavior, such that filtering is maintained for the individual context during a restart but is lost after a Terminate. After a terminate, the breakpoint is installed in all debug contexts.

7.1.14 Is there a way to add breakpoints to C and Assembly files that exist in a library?

Yes. You need to make sure that you compile the library with debug information. You also need to go to Project Properties and make sure that the project has been imported. Make sure that you put a checkmark on the library project.

7.1.15 How do I set a breakpoint at a particular function?

To set a breakpoint at a particular function, you can either do it through the debugger shell or the debugger GUI. If you are going to set it from the debugger shell, you would type in `bp <function_name>`. If you are going to set it from the GUI, then you right-click the function name and select **Toggle Breakpoint**.

7.1.16 What is the difference between a breakpoint and a line breakpoint?

They are practically the same thing. Eclipse introduced Method Breakpoints which we don't have in the classic CodeWarrior tools, hence the need for a distinction. The online documentation states that regular breakpoints suspend the execution of a thread before a line of code or method is executed. Regular breakpoints include:

- Line Breakpoint; that suspends thread execution when the line of code it is applied to is executed.
- Method Breakpoint; that suspends thread execution when the method that it was applied to is entered or exited (or both).
- Address Breakpoint; that breaks when the core executes code at the given address.

7.1.17 The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing?

In order to remove these warning symbols, follow the steps below in the C/C++ perspective:

1. From the IDE menu bar, select **Window > Preferences**.

The **Preferences** window appears.

2. Select **General > Editors > Text Editors > Annotations**.

The **Annotations** settings pane appears.

3. Select **Warnings**.

- In the **Show in** area, clear the **Vertical ruler** checkbox, as shown in the following figure.

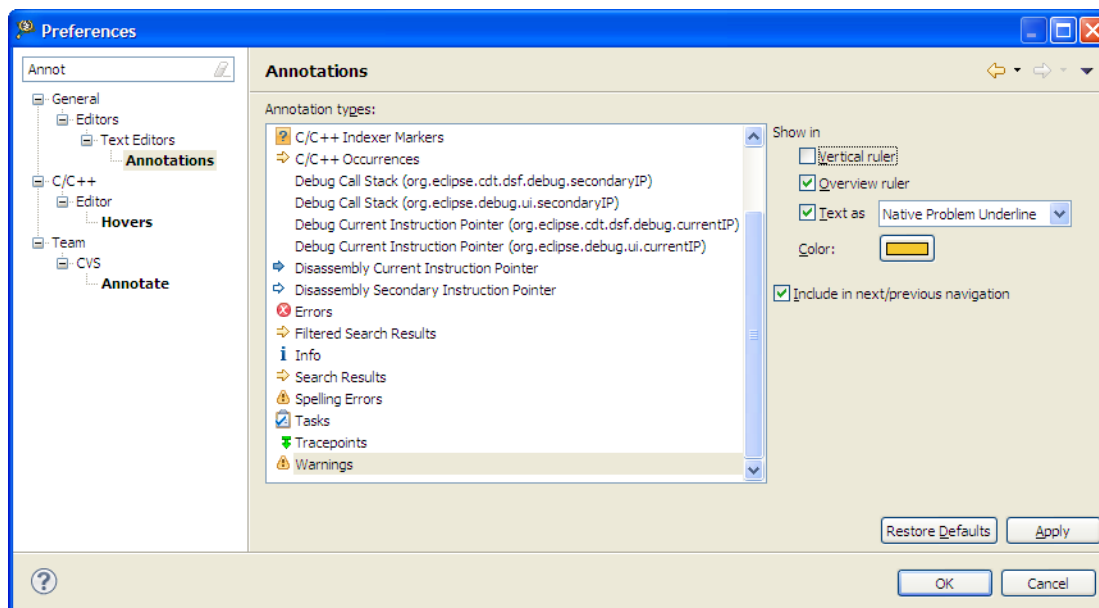


Figure 7-1. Preferences Window - Annotations Page

- Click **Apply** .
- Click **OK** .

The warning symbols are removed.

7.1.18 While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do?

If the debugger fails to restore the original instruction, then the software breakpoints remains and causes a target halt when it reaches that instruction. The problem might be caused by a board or some hardware problem. In such case, the debugger breakpoint list might gets corrupted too. To solve this problem it is best to reload the application to the target.

7.1.19 Can I use software breakpoints on code running in flash memory?

Some debuggers allows software breakpoints on code running in the flash memory. But this usually makes setting and removing breakpoints a time-consuming practice. The debugger has to erase and re-program the flash to set and remove the breakpoints. Other architectures implement a *patch list*, similar to the debugger breakpoint list in RAM. With this it is possible to set software breakpoints with hardware.

7.1.20 What is a hardware breakpoint?

A hardware breakpoint watch an internal bus or the program counter, and if it matches a certain condition, it stops the processor, or do whatever the hardware implements for that condition.

7.1.21 How do I select the type of breakpoint for my application in the Editor view?

To select the type of breakpoint in the **Editor** view, right-click on the marker bar and select the type of breakpoint from the **Breakpoint Types** option's menu list. The following figure shows the context-menu with **Breakpoint Type** option's menu list:

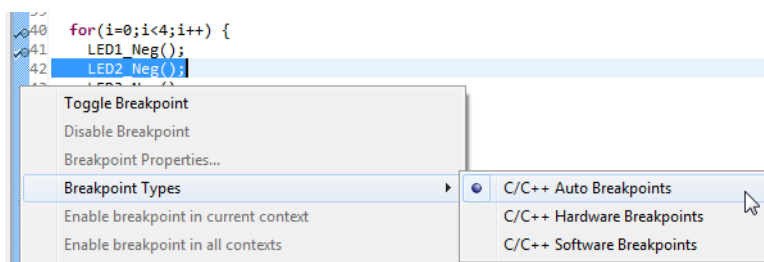


Figure 7-2. Breakpoint Type Menu

This is especially useful if you are running an application with parts in RAM and parts in FLASH, or if you want to use hardware breakpoints for code in RAM.

7.2 Variables

In this topic, FAQs related to variables are listed.

- **NEW** By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?
- **NEW** How do I debug my application using global variables?
- **NEW** How do I filter variables in files?
- **NEW** How do I enable watch on a variable?
- **NEW** What is a static local variable?
- **NEW** When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do?

7.2.1 By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?

While debugging, the debugger lists the local and global variables of the *currentthread* and stack frame only in the **Variables** view. To view the variables from other modules, you have to debug your application using global variables.

7.2.2 How do I debug my application using global variables?

You can debug your application using the variables outside of the current thread or stackframe by adding global variables from the **Variables** view.

To add a global variable:

1. Start a debugging session.
2. The IDE switches to the **Debug** perspective.
3. Click the **Add Global Variables** button from the **Variables** view toolbar. Alternatively, right-click in the **Variables** view and select **Add Global Variables** option from the context menu.

The **Add Globals** dialog box appears listing all global variables present in your application.

Tip

Select **Window > Show View > Variables** from the IDE menu bar to open the **Variables** view.

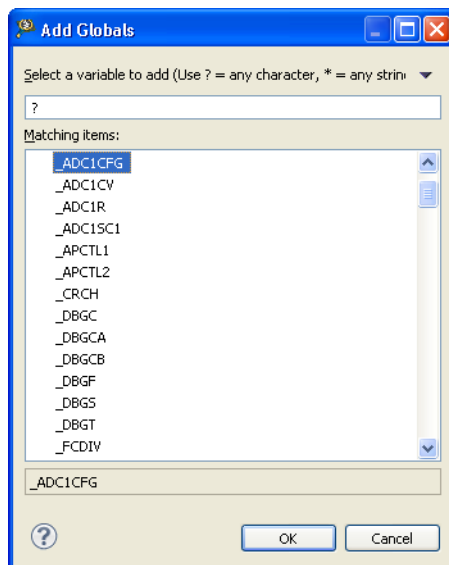


Figure 7-3. Add Globals Dialog Box

4. Enter the variable name in the text box, or you can enter a filter string and select the desired variable from the **Matching items** list.
5. Click **OK**.

The selected variable appears in the **Variables** view.

7.2.3 How do I filter variables in files?

To filter the variables in files:

1. Click the **Add Global Variables** button from the **Variables** view toolbar.
The **Add Globals** dialog box appears.
2. Use the `<variableName>#<fileName>` format in the text box to filter the file.

The desired variable appears in the **Matching items** list.

7.2.4 How do I enable watch on a variable?

To enable **Watch** on a variable:

1. Select the variable for which you want to enable watch in the **Variables** view.
2. Right-click and select **Watch** from the context menu.

The **Expressions** view appears listing the selected variable.

7.2.5 What is a static local variable?

The static local variables are like global variables, but with the function scope.

7.2.6 When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do?

To enable **Watch** on the *static* variables:

1. Click on the static variable name with error in the **Expressions** view's **Name** column.

The editing mode is enabled for the selected variable.

2. Edit the variable name to this format, "`FileName>::variableName`".

For example:

- For `static_global` variable in the `Application.c` file, enter the expression, `"static_global>::Application.c"`
- For `static_local` variable in the `Application.c` file, enter the expression, `"&static_global>::Application.c"`

The watch is now enabled for the edited static variable.

7.3 Build Configuration

In this topic, FAQs related to build configuration are listed.

- [How do I create a new build configuration for my project?](#)
- [Is it possible to exclude a single or multiple files from the project while building it?](#)
- **NEW** [Is it possible to configure build settings for a particular source code file?](#)
- [Is it possible to store output of the build configuration of my project in a separate directory?](#)
- [How can I disable the option of having the CodeWarrior IDE build my project before launching it?](#)

- How can I export the launch configurations settings and then import them back into the project?
- How do I ensure that the launch configurations are a part of the project when it is exported?
- I am unable to specify a JTAG configuration file because the option is disabled. Is there a way to specify it?

7.3.1 How do I create a new build configuration for my project?

To create a new build configuration:

1. Select the project for which you want to create the build configuration in the **CodeWarrior Projects** view.
2. From the IDE menu bar, select **Project > Build Configurations > Manage** .

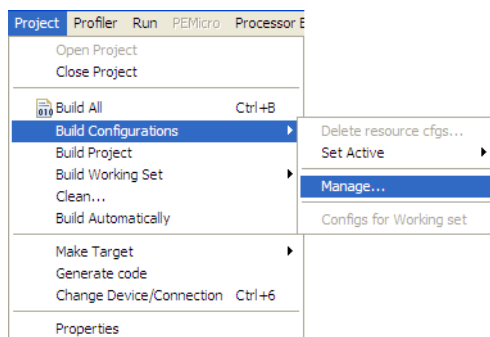


Figure 7-4. Project > Build > Configuration

3. The **Manage Configurations** dialog box appears.
4. Click the **New** button to create a new build configuration.

The **Create New Configuration** dialog box appears.

5. Enter the name and description for the build configuration.
6. Select the appropriate option from the **Copy settings from** section and select the required option. For example, you may choose to copy settings from an existing configuration and select the required configuration.
7. Click **OK** to close the **Create New Configuration** dialog box.
8. Click **OK**.

To configure build settings of the newly created build configuration:

1. Select the project with which the build configuration is associated in the **CodeWarrior Projects** view.
2. From the IDE menu bar, select **Project > Properties**.

The **Properties** dialog box appears.

3. Select **C/C++ Build > Settings**.
4. Select the newly created configuration from the **Build Configuration** drop-down list.
5. Specify the settings for the selected build configuration, as required.
6. Click **OK**.

The build settings are configured, as specified.

7.3.2 Is it possible to exclude a single or multiple files from the project while building it?

Yes, you can exclude single or multiple files from build:

1. Select the file in the **CodeWarrior Projects** view. To select multiple files, press and hold the **Ctrl** key.
2. Right-click and select **Resource Configurations > Exclude from build**

The **Exclude from build** dialog box appears.

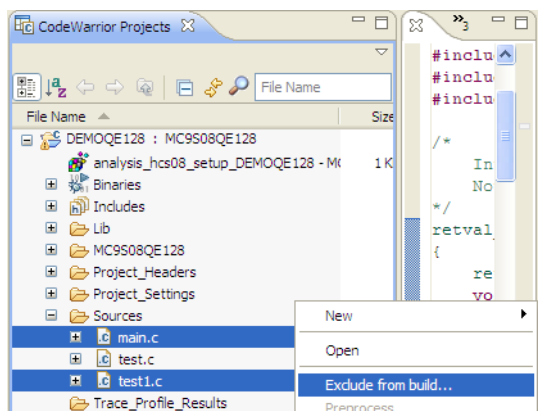


Figure 7-5. Exclude from build

3. Select the build configuration from which you want to exclude the selected file.
4. Click **OK**.

The selected files are excluded from the build configuration.

7.3.3 Is it possible to configure build settings for a particular source code file?

Yes, to set file specific build options:

1. Right-click the file in the **CodeWarrior Projects** view and select **Properties** .

The **Properties** dialog box for the selected file appears.

2. Specify the required settings. For example, you can select a specific compiler option for compiling the file.

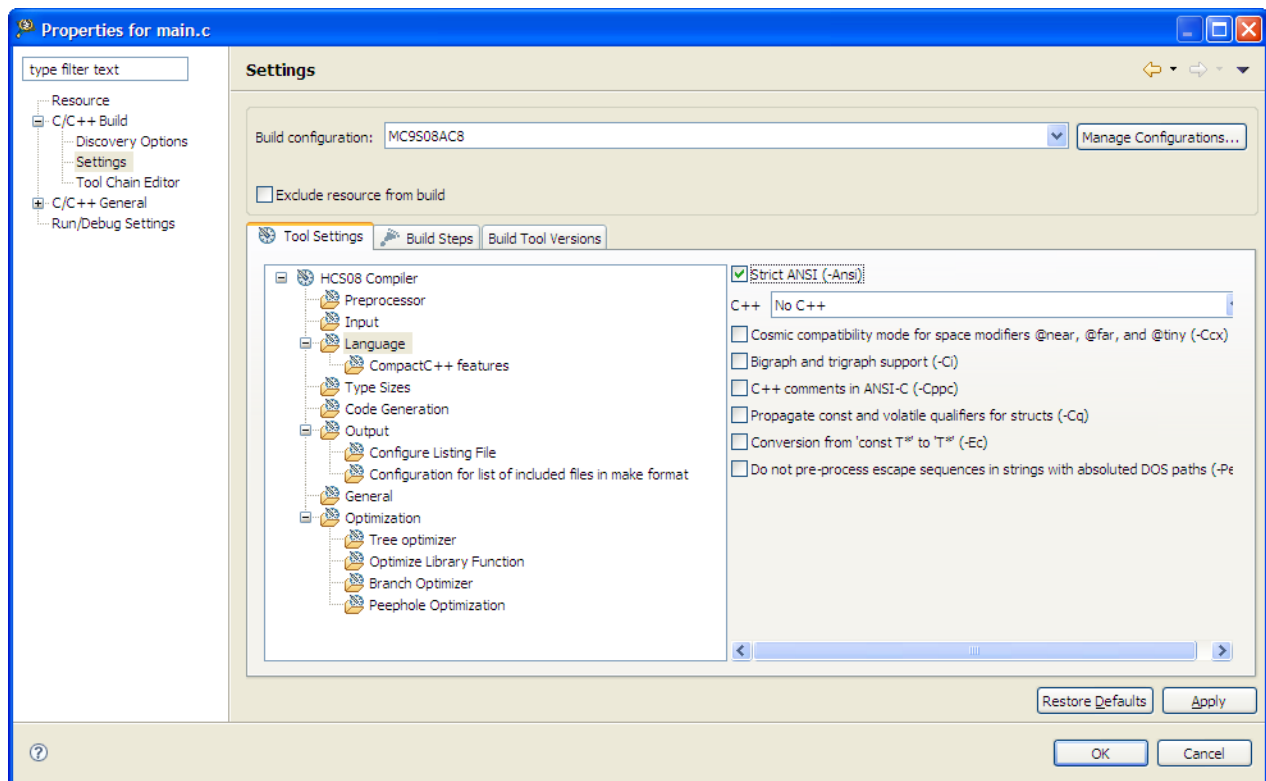


Figure 7-6. Properties Dialog Box for Selected File

3. Click **OK** .

The file specific settings are configured, as specified.

7.3.4 Is it possible to store output of the build configuration of my project in a separate directory?

Yes, you can specify output directory for the build configuration.

1. Select the project with which the build configuration is associated in the **CodeWarrior Projects** view.
2. From the IDE menu bar, select **Project > Properties**.

The **Properties** dialog box appears.

3. Select **C/C++ Build**.

The **C/C++ Build** page appears in the right panel of the **Properties** dialog box.

4. Select the build configuration from the **Configuration** drop-down list.
5. Specify the required build directory in the **Build directory** text box.
6. Click **Apply**.
7. Click **OK**.
8. Right-click the project and select **Build Configurations > Build Selected**.

The **Clean and Rebuild Configurations** dialog box appears.

9. Select the build configuration to build and click **OK**.

The project build begins and the output is stored in the specified directory.

7.3.5 How can I disable the option of having the CodeWarrior IDE build my project before launching it?

To disable build before launch:

1. From the IDE menu bar, select **Window > Preferences** .

The **Preferences** dialog box appears.

2. Expand the tree control to select **Run/Debug > Launching** .

The **Launching** page appears in the right pane.

3. Clear the **Build (if required) before launching** checkbox.
4. Click **Apply** .
5. Click **OK** .

Now, CodeWarrior IDE will not build the project before launching it.

7.3.6 How can I export the launch configurations settings and then import them back into the project?

You can export and import the launch configuration settings using the Export and Import wizards.

To export a launch configuration:

1. From the CodeWarrior menu bar, select **File > Export** .

The **Export** dialog box appears.

2. Expand **Run/Debug** tree control and select **Launch Configurations** .

The **Export Launch Configurations** wizard appears.

3. Select the launch configuration you want to export.
4. In the **Location** field, specify the location where you want to export the launch configuration.
5. Click **Finish** .

The selected launch configuration exports as `.launch` file.

Similarly, you can import the launch configuration (`.launch` file) using **File > Import** in the CodeWarrior IDE menu bar.

Alternatively, you can copy the `*.launch` files present in the `Project_Settings\Debugger` folder of a project and place it in the project where you want to export it.

7.3.7 How do I ensure that the launch configurations are a part of the project when it is exported?

To make a launch configurations be part of the project, follow steps below:

1. From the IDE menu bar, select **Run > Debug Configurations** .

The **Debug** window appears.

2. Under **CodeWarrior Download**, select **the launch configuration of the project that needs to be exported**.
3. Select the **Common** tab.

The **Common** settings pane appears.

4. Under **Save as** , select the shared file radio button. Leave the default folder name as is.
5. Click **Apply** .

Now the launch configuration is physically located in the project folder, so it will be a part of the project when zipped.

7.3.8 I am unable to specify a JTAG configuration file because the option is disabled. Is there a way to specify it?

Yes there is a way to specify this file. To specify a JTAG configuration file, follow these steps:

1. Right-click on the project name.
A context sensitive menu appears.
2. From the context menu, select **Debug As > Debug Configurations**.
The **Debug Configurations** window appears.
3. Under **CodeWarrior Download**, select project name.
4. In the **Main** tab, click **Edit** that appears next to the **Connection** drop-down list.
The **Properties for <connection launch configuration>** window appears.
5. Click the **Edit** button next to the **System** drop-down list.
The **Properties for <system launch configuration>** window appears.
6. Click the **Edit** button next to the **System type** drop-down list.
The **System Types** dialog box appears.
7. Click **Import**.
The **Import System Type** dialog box appears.
8. Select the JTAG initialization file that describes the items on the JTAG chain from this location: `<CWInstallDir>\PA\PA_Support\Initialization_Files\jtag_chains`
9. Click **Open**.
10. Click **OK**.

7.4 Flash Programmer

In this topic, FAQs related to flash programmer are listed.

- [How can I create the flash programmer, hardware diagnostics, or import/export fill memory tasks in the CodeWarrior IDE?](#)

- I need to flash a large size application (more than 64 KB) , and I also need to perform target tasks, such as secure/unsecure the device and additional verify or erase. How do I do it?
- Is there a way to quickly flash multiple boards with different S-Records?
- Is it possible to debug the target multiple times with the same binary application, without flashing the device?
- I need to enable logging for flash programming. How do I do it?
- **NEW** How can I program my boards without a project?

7.4.1 How can I create the flash programmer, hardware diagnostics, or import/export fill memory tasks in the CodeWarrior IDE?

CodeWarrior IDE provides a framework, **Target Task** view, to create tasks to:

- flash, erase, verify a device,
- perform hardware diagnostic, or
- read, write, or export memory.

To open the **Target Task** view:

1. Select **Window > Show View > Other**.

The **Show View** dialog box appears.

2. Type "Target Taks" view as the filter text to narrow down the list of the views in the dialog box.
3. Select **Debug > Target Tasks** and click **OK**.

The **Target Tasks** view opens in the Workbench window.

7.4.2 I need to flash a large size application (more than 64 KB) , and I also need to perform target tasks, such as secure/unsecure the device and additional verify or erase. How do I do it?

You can have the debugger flash the target using a target task. The flash programming target tasks are small application applets, which are downloaded to the target by the debugger. The debugger gets the application binary from the host and programs the flash.

A flash programming task is setup in the **Run > Debug Configurations > Debugger > Download** tab.

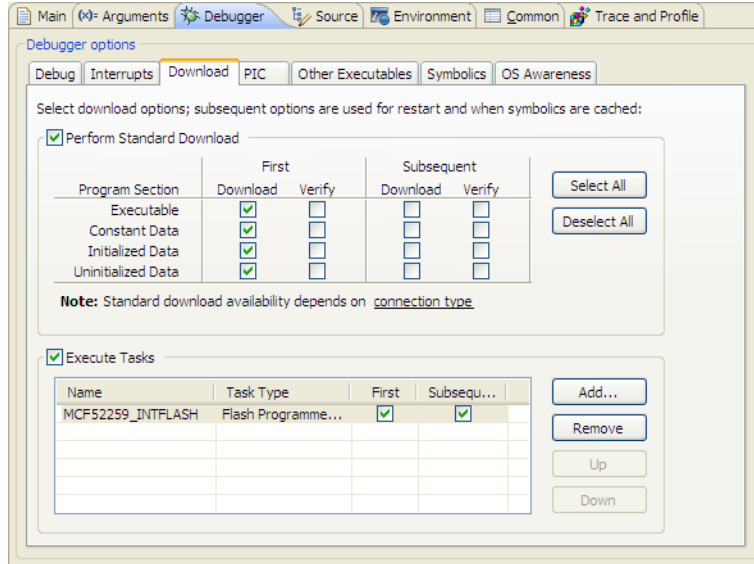


Figure 7-7. Flash Programming Target Task Associated with Debugger

This means that the debugger is using the attached target task to be executed as part of the download.

However, you can also perform flash programming as a standalone task, without combining it with the debug task.

1. Select **Flash Programmer > Open Flash Programmer** on the CodeWarrior IDE tool bar.

The **Target Tasks** view appears.

2. To create a new target task, either import a flash configuration file or duplicate an existing flash task.
3. Right-click the new target task and select **Change Run Configuration**.
4. Change the run configuration from **Active Debug Context** to a defined launch configuration.
5. Click the **Execute** button to execute the target task.

When the task is executed, the **Console** view shows the following results.

```

Flash Programmer Console
Program Command Succeeded
Downloading 0x00000018 bytes to be programmed at 0x00000400
Programming ...
Program Command Succeeded
Downloading 0x0000199C bytes to be programmed at 0x00000420
Programming ...
Program Command Succeeded
cmdwin::fl::verify
Beginning Operation ...
-----
Verifying file C:\Users\Erich Styger\Data\HTA\CW\wsp_Intro\Tow
Initializing remote connection ...
Performing target initialization ...
Auto-detection is successful.
File is of type Elf Format.

Reading 0x00000300 bytes starting from address 0x00000000
Reading 0x00000018 bytes starting from address 0x00000400
Reading 0x0000199C bytes starting from address 0x00000420
Verify Command Succeeded
    
```

Figure 7-8. Console View

You can use this information to diagnose issues with the task execution, if there are any.

You can also access the previously executed target task directly from the **Flash Programmer** button on the CodeWarrior IDE toolbar.

7.4.3 Is there a way to quickly flash multiple boards with different S-Records?

You can flash the target without the target task.

1. From the CodeWarrior IDE toolbar, select **Flash Programmer > Flash File to Target**.

The **Flash File to Target** dialog box appears.

2. Specify the run configuration that defines the connection, the flash configuration, and the file you want to flash.
3. Click the required button, **Erase Whole Device** or **Erase and Program**.

The diagnostic information is logged to the **Console** View. This helps you view the flash results in detail when flashing large applications.

7.4.4 Is it possible to debug the target multiple times with the same binary application, without flashing the device?

Yes, you can debug the target multiple times with the same application.

1. Flash the target once with either using a target task (standalone) or using **Flash Programmer > Flash File to Target**.
2. Select **Run > Debug Configurations**.

The **Debug Configurations** dialog box appears.

3. Select **Debugger > Download**.

The **Download** page appears in the **Debug Configuration** dialog box.

4. Clear the **Perform Standard Download** and **Execute Tasks** checkboxes.

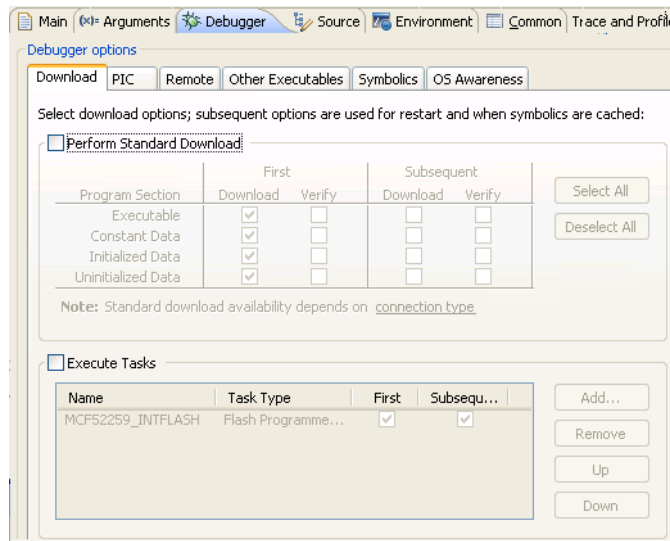


Figure 7-9. Download Page - Debug Configurations Dialog Box

Now as the debugger does not need to flash a potentially large application, you can start debugging within a few seconds, and you can do this multiple times.

5. To flash the target, click the **Flash Programmer** button followed by the **Debug** button on the CodeWarrior IDE toolbar.

Now, you can debug the target multiple times with the same binary application, without flashing the device

7.4.5 I need to enable logging for flash programming. How do I do it?

Various issues may occur while flash programming, such as the target is not stable, you might have problems with the target power, or the part might be secured.

To gather more information about such issues, you can enable logging.

1. From the CodeWarrior IDE menu bar, select **Run > Debug Configurations** .

The **Debug Configurations** dialog box appears.

2. Click the **Edit** button in the **Remote system** section.

The **Properties for <project_name>** dialog box appears.

3. Check the **Enable Logging** checkbox in the **Connection** page.
4. Click **OK** .
5. Click **Debug** .

The log results appear in the **Console** view.

7.4.6 How can I program my boards without a project?

You can program your boards without a project by using the **Flash File To Target** feature. To use this feature you need to have the binary file that you want to flash for your device. The following steps lists the procedure to program the boards without a project:

1. Click the *downward arrow key* of the **Flash Programmer** button from the IDE menu bar and select **Flash File To Target** option.

The **Flash File To Target** dialog box appears.

2. Click **New** from the **Connection** group.

The **New Connection** dialog box appears.

3. Select **Hardware or Simulator Connection** option from the **CodeWarrior Bareboard Debugging** group.
4. Click **Next**.

The **Hardware or Simulator Connection** page appears.

5. Enter the name and description in the **Name** and **Description** text boxes.
6. Click **New** in the **Target** group to add a new target.

The **New Connection** dialog box appears.

7. Select **Hardware or Simulator Target** from the CodeWarrior Bareboard Debugging group.
8. Click **Next**.

The **Hardware or Simulator Target** page appears.

9. Enter the name and description in the **Name** and **Description** text boxes.
10. Select the desired target from the **Target type** drop-down list.
11. Click **Finish**. The **Hardware or Simulator Target** page closes and the **Hardware or Simulator Connection** page appears.
12. Select the desired connection from the **Connection type** drop down list.
13. This expands the dialog for the run control specific settings.
14. Click **Finish**.

The **New Connection** dialog box closes and the **Flash File To Target** dialog box appears.

15. From the **File Options** group, click **Browse** and select the desired binary file to be programmed on the board.
16. Select the **Erase and Program** button to mass erase your device and program it with the file.

Your board is programmed with the binary file you selected without accessing a project.

7.5 Miscellaneous

In this topic, miscellaneous FAQs related to the debugger are listed.

- [How can I view only certain registers?](#)
- [How can I view addresses in memory?](#)
- [How do I open up multiple memory locations in the memory view?](#)
- [How can I view variables that I have set in my project?](#)
- [How can I change the values of variables?](#)
- [How do I view global variables that have been accessed in my project?](#)
- [Is the command-line debugger available in the Eclipse-based CodeWarrior IDE as well?](#)
- [Is it possible to specify a different name for the binary file \(.elf\)?](#)
- [Can I debug the code from the generated assembly file step-by-step?](#)
- [If I have a debugger session running on my application, how can I halt the application?](#)
- [Why do I get the following error message when I attempt to debug my project?](#)
- [Can I change the format of the Memory view?](#)
- [Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective?](#)
- [Is there a way to display the variables always in the Hexadecimal format?](#)

- Can I change the endianness that is displayed in the Memory view?
- In the Memory view, is it possible to go back to the address that the rendering was created for?
- Is there a way to view disassembly at any location?
- Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format?
- What does the Reset on Connect option do?
- Can I change the Program Counter (pc) value while in the debugger without having to restart the debugger?
- How do I specify the program entry point?
- How can I run a debug session repeatedly?
- Can I direct console output to a file instead of the console window?
- How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration?
- How can I inspect global variables in the variables view?

7.5.1 How can I view only certain registers?

To view only certain registers, follow these steps:

1. Select **Window > Show View > Registers** from the IDE menu bar while in the debugger perspective.

The **Registers** view appears docked to the upper right hand side of debugger perspective.

2. Right-click anywhere in the **Registers** tab.

A context menu appears.

3. Select Add Register Group in the context menu.

The **Register Group** window appears.

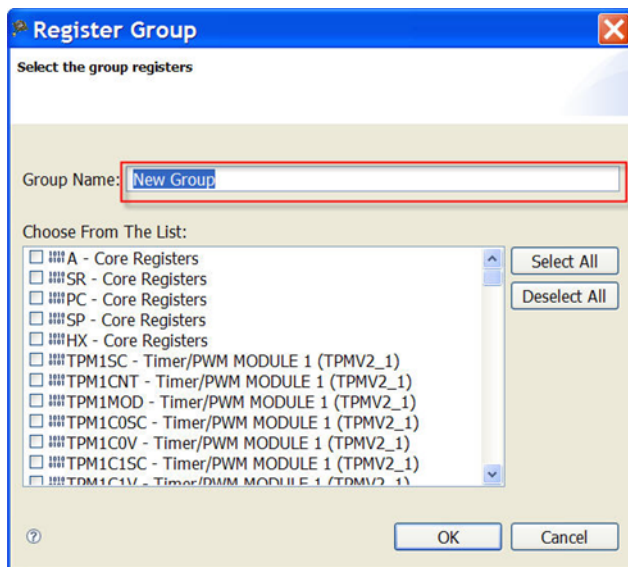


Figure 7-10. Register Group Window

4. Type the name of new register group in the **Group Name** textbox.
5. Select the checkbox next to registers that you want to be a part of this group.
6. Click **OK** .

The **Register Group** window closes.

Now you can view only the selected registers.

7.5.2 How can I view addresses in memory?

To view addresses in memory, follow these steps:

1. Select **Window > Show View > Memory** in the debug perspective.

The **Memory** window appears docked to the lower part of debugger perspective.

2. Click  sign icon.

The **Monitor Memory** window appears.

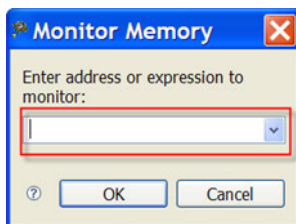


Figure 7-11. Monitor Memory Window

3. Type memory address in the **Enter address or expression to monitor** textbox.
4. Click **OK** .

The **Monitor Memory** window closes.

The memory location appears in the memory window. The memory contents can be viewed by selecting memory on the left hand side, as shown in the following figure.

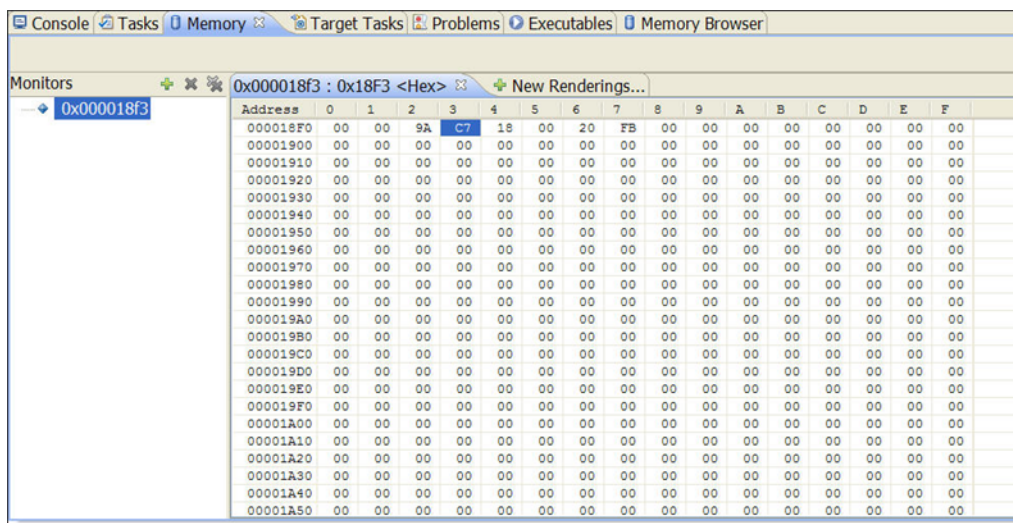


Figure 7-12. Memory View

7.5.3 How do I open up multiple memory locations in the memory view?

To open multiple memory locations in the memory view, follow these steps:

1. Select **Window > Show View > Memory** in the debug perspective.

The **Memory** window appears docked to the lower part of debugger perspective.

2. Click  sign icon.

The **Monitor Memory** window appears.

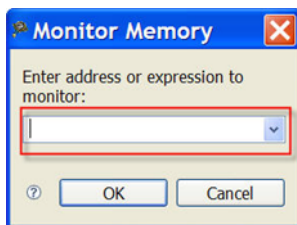



Figure 7-13. Monitor Memory Window

3. Type memory address in the **Enter address or expression to monitor** textbox.
4. Click **OK** .

The new address appears in the memory window.

5. Click  sign icon.
6. The **Monitor Memory** window appears.
7. Type another memory address in the **Enter address or expression to monitor** textbox.
8. Click **OK** .

The new address appears in the memory window. The contents can be viewed by selecting memory on the left hand side, as shown in the following figure.

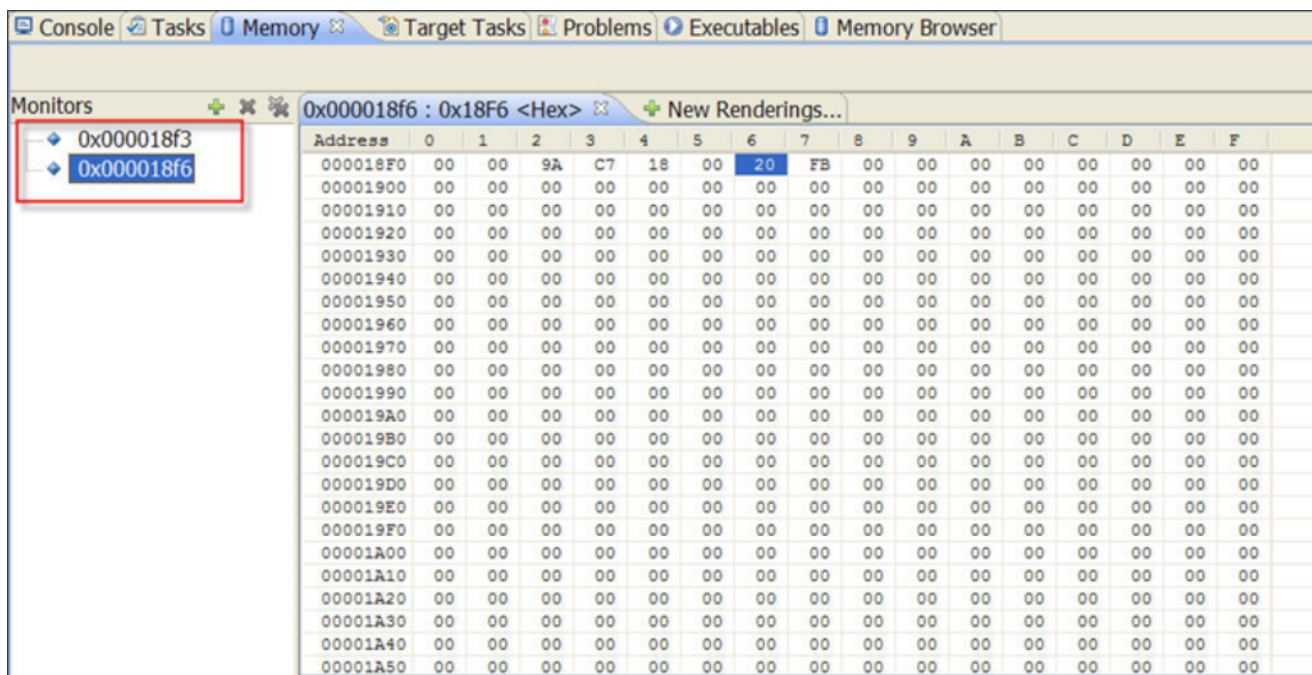


Figure 7-14. Memory View with Multiple Addresses

7.5.4 How can I view variables that I have set in my project?

To view variables from the debug perspective toolbar, select **Window > Show View > Variables** . The Variable window appears docked on the upper right hand side of debug perspective.

7.5.5 How can I change the values of variables?

To change the value of variables, follow the steps below in the debug perspective:

1. From the toolbar, select **Window > Show View > Variables** . The **Variables** window appears docked on the upper right hand side of debugger perspective.
2. Right-click on the variable.

A context menu appears.

3. From context menu, select **Change Value** .

The **Set Value** window appears.

4. In the **Enter a new value for variables** textbox, type a new variable value.
5. Click **OK** .

The **Set Value** window closes.

The variable value changes in the **Variables** window. The variable being changed is highlighted in yellow.

7.5.6 How do I view global variables that have been accessed in my project?

To view global variables that have been accessed in your project, follow these steps:

1. Select **Window > Show View > Variables** from the IDE menu bar while in the debugger perspective.

The **Variables** view appears docked to the upper right hand side of debugger perspective.

2. Right-click a variable.

A context menu appears.

3. Select **Add Global Variables** in the context menu.

The **Add Globals** dialog box appears.

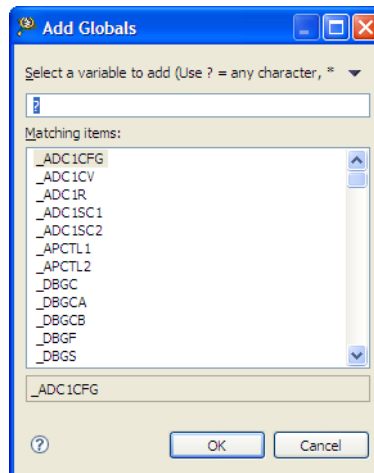


Figure 7-15. Add Globals Dialog Box

4. Select the global variables that you want to view.
5. Click **OK** .

The **Add Globals** dialog box closes.

The selected variables appear in the **Variables** view with the symbol that identifies them as global variable.

NOTE

The selected global variables will persist throughout the session and subsequent sessions, until they are removed.

7.5.7 Is the command-line debugger available in the Eclipse-based CodeWarrior IDE as well?

Yes. To use the Command Line Debugger follow steps given below in debug perspective:

1. From the IDE menu bar, select **Window > Show View > Debugger Shell** .
2. Click **OK** .

The Debugger Shell window appears docked on the lower right-hand corner of debug perspective.

7.5.8 Is it possible to specify a different name for the binary file (.elf)?

Yes this can be done by following steps below:

1. In the CodeWarrior Projects window, right-click the project name.

A context menu appears.

2. From the context menu, select **Properties**.

The **Properties** window appears.

3. Select **C/C++ Build > Settings**.

The **Settings** options appear on the right side of properties window.

4. Select **Build Artifact** tab.

5. Type the name of the executable that you want generated in the **Artifact name** textbox.

6. The **Artifact extension** textbox contains the extension of the executable that will be generated. The default extension for binary files is .elf. If another extension is desired, change the extension in the **Artifact extension** textbox.

7. Click **Apply**.

8. Click **OK**.

The **Properties** window disappears.

When the project is rebuilt a new binary with the new extension is generated.

7.5.9 Can I debug the code from the generated assembly file step-by-step?

Yes this can be done. From the debugger perspective toolbar, select **Window > Show View > Disassembly** . The **Disassembly** window appears displaying the assembly code generated from the C code. Breakpoints can be set by double-clicking on assembly line.

7.5.10 If I have a debugger session running on my application, how can I halt the application?

In the **Debug** perspective, click on the .abs being executed. From toolbar, select **Suspend** from the toolbar which halts the application temporarily. To run the application, select **Resume** from toolbar.

7.5.11 Why do I get the following error message when I attempt to debug my project?

```
Program does not exist
```

This error occurs because the project you are trying to debug has not been built yet. To build the project, right-click the project name and select the **Build Project** option.

7.5.12 Can I change the format of the Memory view?

Yes. In order to change the format of the memory window do the following steps:

1. Select **Window > Show View > Memory** in the debug perspective.

The **Memory** window appears docked to the lower part of debugger perspective.

2. Click  sign icon.

The **Monitor Memory** window appears.

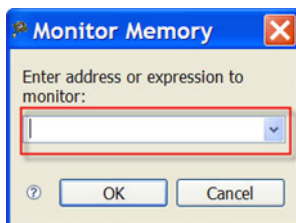


Figure 7-16. Monitor Memory Window

3. Type memory address in the **Enter address or expression to monitor** textbox.
4. Click **OK**.

The **Monitor Memory** window closes. The memory location appears in the memory window. The memory contents can be viewed by selecting memory on the left hand side, as shown in the following figure.

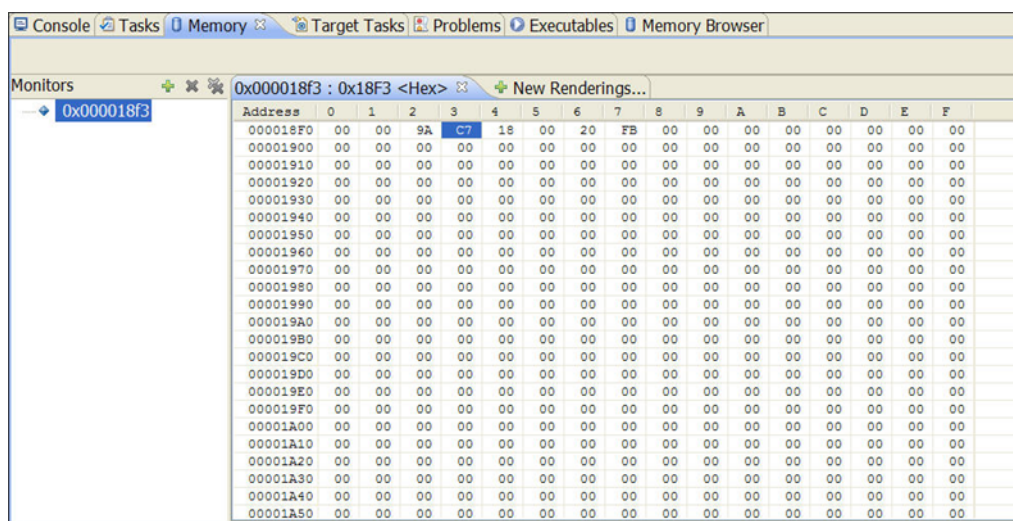


Figure 7-17. Memory View

5. Right-click anywhere in the **Memory** view

A context menu appears.

6. From context menu, select **Format**.

The **Format** window appears.

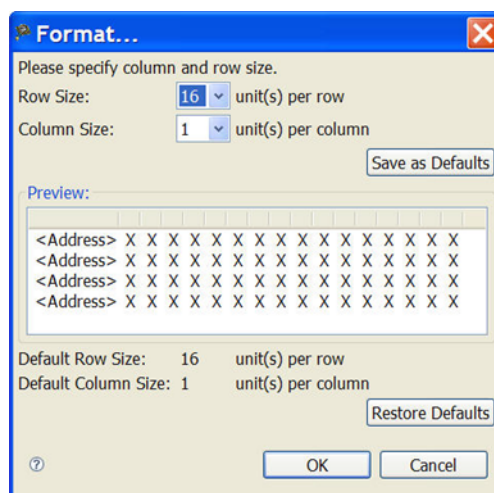


Figure 7-18. Format Dialog Box

Row Size indicates how the addresses will increase. **Column Size** displays the number of bytes in that address space. For this example, the address will increase by 16 bytes and each row will display 4 bytes in each column. The **Preview** window changes and displays address every 2 bytes by 2 bytes per column.

7. Click **OK**.

The **Format** window closes.

The memory view changes and displays the addresses using new format, as the following image shows.

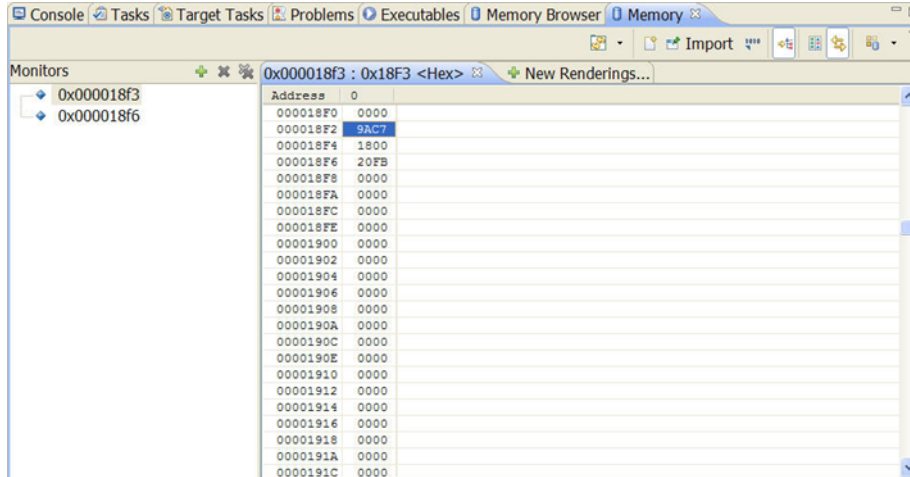


Figure 7-19. Updated Memory View

7.5.13 Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective?

The Debugger Engine (DE) is loaded when it is first needed in one of the following situations:

1. Debugger is launched
2. Debugger Shell is opened

The DE.exe is terminated only when CodeWarrior is terminated.

7.5.14 Is there a way to display the variables always in the Hexadecimal format?

Yes. There is a way to display the variables always in Hexadecimal instead of the default decimal format. In order to accomplish this, follow steps below.

1. Select **Window > Preferences** from C/C++ perspective toolbar.

The **Preferences** window appears.

2. Select **C/C++ > Debug**.

The **Debug** settings appear.

3. Change **Default variable format** to **Hexadecimal**. By default, it is set to **Natural** (for more information, refer to the image listed below).
4. Click **Apply**.
5. Click **OK**.
6. Restart CodeWarrior Development Studio.

When **Variables** view appears in the debug perspective, the variables appear in the Hexadecimal format.

NOTE

This setting is a global setting.

7.5.15 Can I change the endianness that is displayed in the Memory view?

Yes. To change the endianness that is displayed in the memory view, follow these steps:

1. Select **Window > Show View > Memory** in the debug perspective.

The **Memory** window appears docked to the lower part of debugger perspective.

2. Click  sign icon.

The **Monitor Memory** window appears.

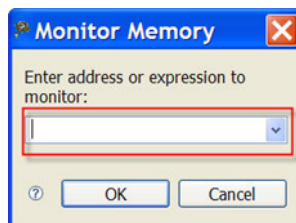


Figure 7-20. Monitor Memory Window

3. Type memory address in the **Enter address or expression to monitor** textbox.
4. Click **OK**.

The new address appears in the memory window. The contents can be viewed by selecting memory on the left hand side, as the following figure shows.

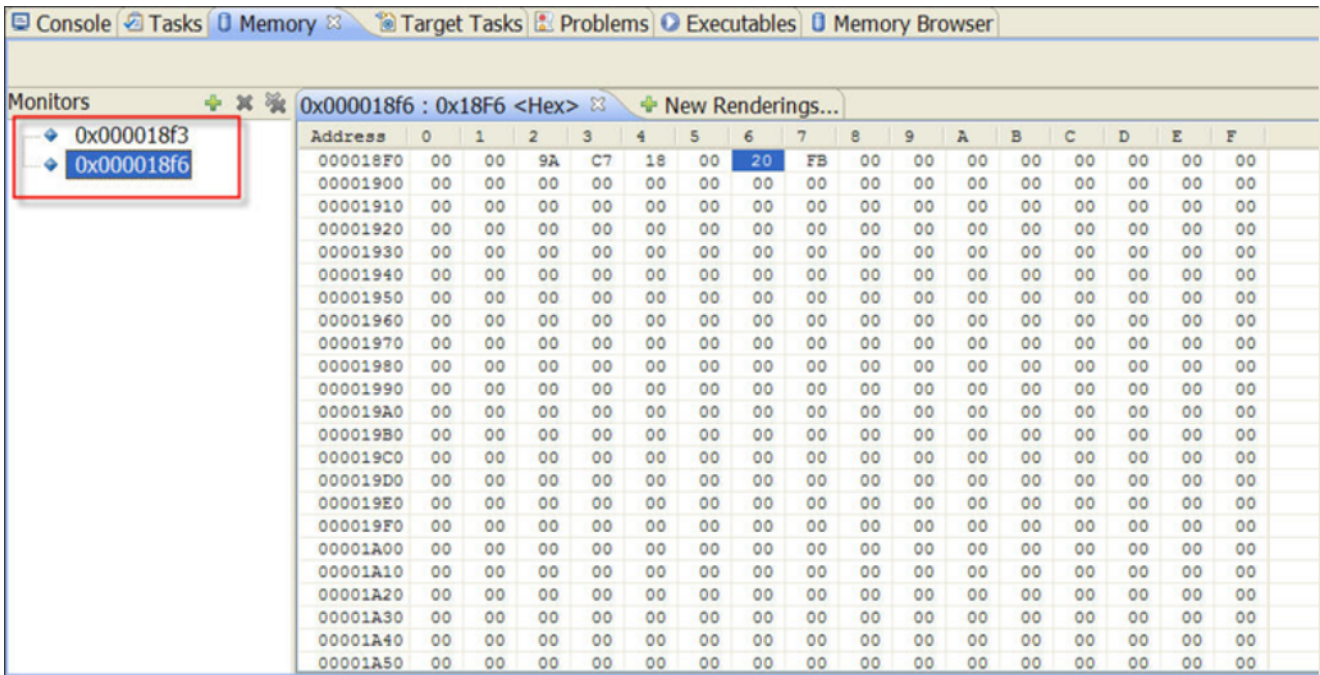


Figure 7-21. Memory View with Multiple Addresses

5. From the **Memory** Toolbar, select **New Renderings...** .

The **New Renderings** window appears.

6. Select **Traditional**.

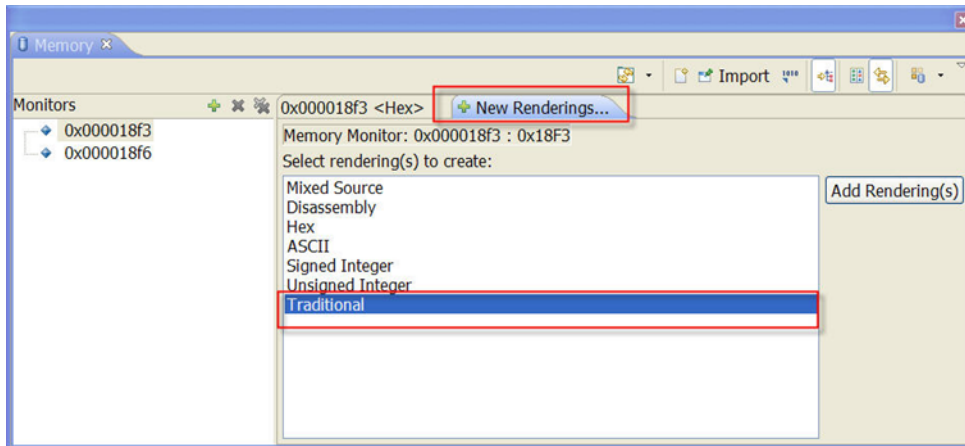


Figure 7-22. Memory View-New Renderings

7. Click **Add Renderings**.

8. Click **OK**.

The **Traditional** view appears in Memory view.

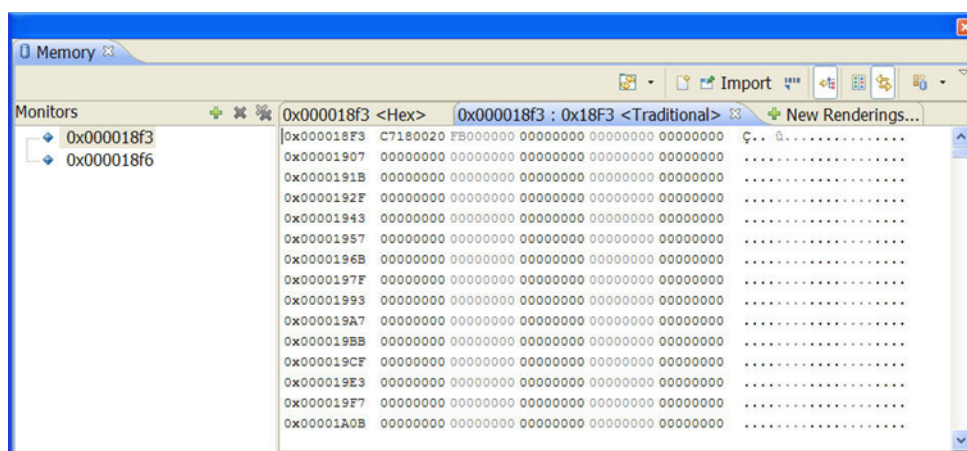


Figure 7-23. Memory View-Traditional Tab

9. Right-click anywhere in the **Traditional** tab.

A context menu appears.

10. From the context menu, select **Endian > Little**. By default it is **Big**.

The memory view displays little endianness.

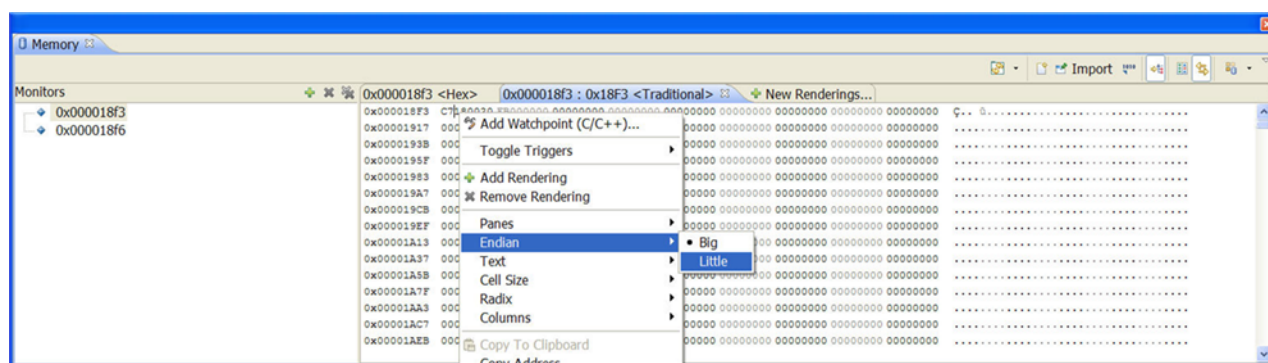


Figure 7-24. Memory View-Little Endian Addresses

NOTE

To change the endianness back to **Big**, right-click anywhere in the **Traditional** tab, and select **Endian > Big** from the context menu.

7.5.16 In the Memory view, is it possible to go back to the address that the rendering was created for?

Yes. In order to get back to the address that the rendering was created for, follow steps below:

1. Right-click anywhere in the **Memory** view.
A context window appears.
2. From context menu, select **Reset to Base Address** .

The view changes to display the base address that the rendering was initially created for.

7.5.17 Is there a way to view disassembly at any location?

Yes it is possible to view disassembly at any location. To do so, follow steps below:

1. Right-click anywhere in the **Memory** view.
A context window appears.
2. From context menu, select **Go To Address**.
The **Go To Address** textbox appears.
3. Enter the address in the Address textbox. If the **Input as Hex** checkbox is selected, then the address can be entered as either a hexadecimal or a decimal address.
4. Click **OK**.

The assembly instructions located at the specified address appears in the **Memory** view (for more information, refer to the image listed below).

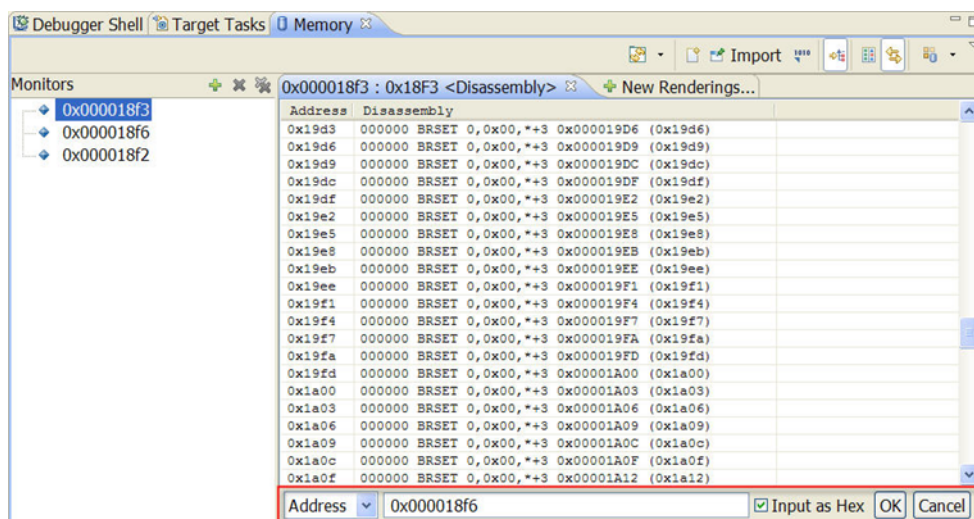


Figure 7-25. Memory View-Assembly Instructions

7.5.18 Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format?

Yes. It is possible to display the expressions always in Hexadecimal format instead of the default decimal format. In order to accomplish this follow steps below:

1. Select **Window > Preferences** from C/C++ perspective toolbar.

The **Preferences** window appears.

2. Select **C/C++ > Debug**.

The **Debug** settings appear.

3. Change **Default expression format** to **Hexadecimal**. By default, it is set to **Natural** (for more information, refer to the image listed below).

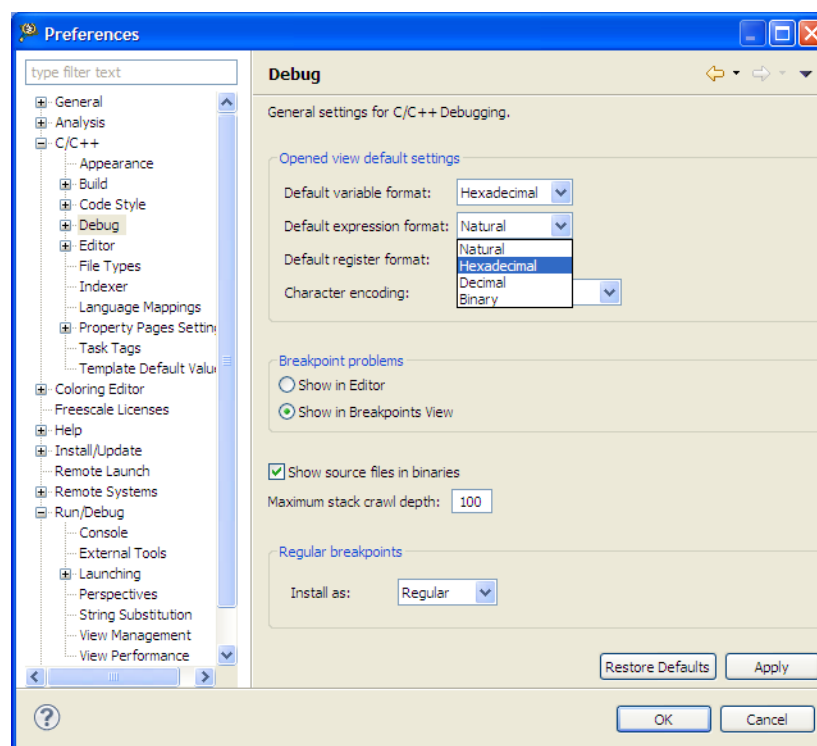


Figure 7-26. Preferences Window-Debug Page

4. Click **Apply**.
5. Click **OK**.
6. Restart CodeWarrior Development Studio.

When **Variables** view appears in the debug perspective, the variables appear in the Hexadecimal format.

NOTE

This setting is a global setting.

7.5.19 What does the Reset on Connect option do?

The **Reset on Connect** option resets all the cores.

7.5.20 Can I change the Program Counter (pc) value while in the debugger without having to restart the debugger?

Yes. In order to change the Program Counter (pc) without restarting the debugger, follow steps below in the debug perspective:

1. Start a debug session.

In the editor view, right-click on the source line where you want the pc to move to.

2. A context menu appears.
3. From context menu, select **Move To Line**.

The pc moves to line. The debugger moves the program counter to the location you specified. The editor view shows the new location.

7.5.21 How do I specify the program entry point?

To specify the program entry point, follow steps below in C/C++ perspective:

1. In the CodeWarrior projects window, right-click on project name.

A context menu appears.

2. From context menu, right-click on project name, select **Debug As > Debug Configurations**.

The **Debug** window appears.

3. Under **CodeWarrior Download**, select project name.
4. Select **Debugger** tab.

The **Debugger options** page appears.

5. Under Program Execution, select the Program Entry point option.
6. Click **Apply**.
7. Click **Debug**.

7.5.22 How can I run a debug session repeatedly?

There are several of ways to run a debug session repeatedly. Below is a list of possible ways to run a debug session repeatedly:

- Select the **Debug** button to invoke the last debug session.

-or-

- Right-click the stack window and select **Relaunch** from context menu.

-or-

- If still in debug mode, click **Terminate and Relaunch** .

7.5.23 Can I direct console output to a file instead of the console window?

Yes, you can direct the console output to a file instead of the console window. To accomplish this, follow steps given below:

1. In the **CodeWarrior Projects** view, right-click on project name.

A context menu appears.

2. From context menu, right-click on project name, select **Debug As > Debug Configurations**.

The **Debug** window appears.

3. Under **CodeWarrior**, select project name.
4. Select **Common** tab.

The **Common** page appears (for more information, refer to the image *Debug Configurations-Common Page* listed below).

5. Select **File** and specify the file to which the console output is directed.

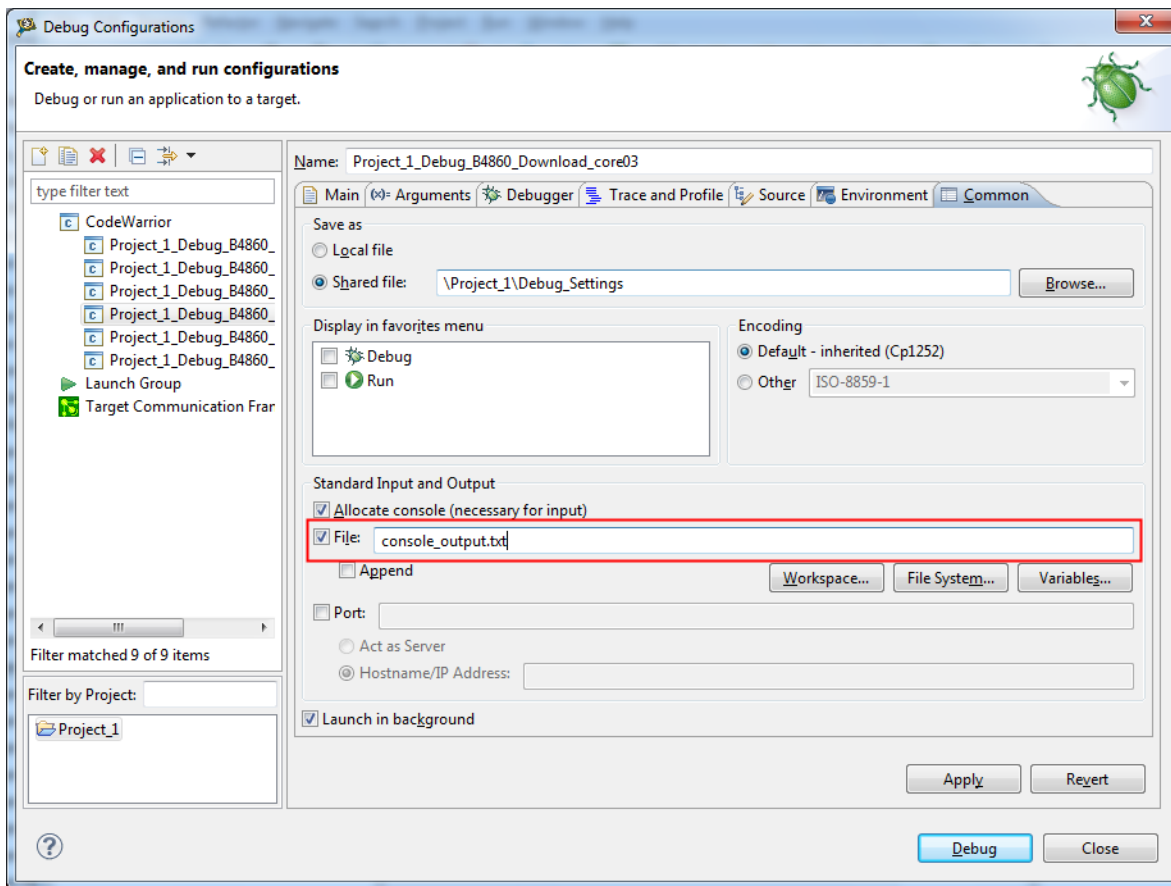


Figure 7-27. Debug Configurations-Common Page

6. Click **Apply**.
7. Click **Debug**.

The console output will be directed to the specified file.

7.5.24 How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration?

```
Could not find symbol xx in index.
```

To resolve this error message, follow the steps below:

1. In the CodeWarrior project window, right-click on project name.
A context menu appears.
2. From context menu, select **Properties** .

The **Properties** window appears.

3. Select **C/C++ General > Indexer** .

The **Indexer Settings** appear.

4. Check the **Enable project specific settings** checkbox.
5. From the **Select Indexer** drop-down list, select **Full C/C++ Indexer (complete parse)** .
6. Click **Apply** .
7. Click **OK** .

The issue will be resolved. Now, you can use Ctrl+click to open the variable declaration.

7.5.25 How can I inspect global variables in the variables view?

You can add or view global variables using the **Add Globals** dialog box. To open the **Add Globals** dialog box:

1. Right click the **Variables** view in the Debug perspective.
2. From the context menu, select **Add Global Variables**.

The **Add Globals** dialog box appears.

3. Select the variable you want to add as global variable.
4. Click **OK**.



Chapter 8

Debugger Shell

In this chapter, you find Frequently Asked Questions (FAQs) related to the Debugger Shell.

- [FAQs - Debugger Shell](#)

8.1 FAQs - Debugger Shell

This topic lists the FAQs related to Debugger Shell.

- [How can I find the default value of memory width? How can I change the value if I want to?](#)
- [Can I change the Program Counter \(pc\) value?](#)
- [Does the config page off option still available in the Eclipse-based CodeWarrior IDE?](#)
- [Is there a way to accelerate the execution speed and turn off the printing when I use a lot of mem commands in the debugger shell?](#)

8.1.1 How can I find the default value of memory width? How can I change the value if I want to?

To find the default value for memory width, follow steps below in debug perspective:

1. From Debug Perspective toolbar, select **Window > Show View > Debugger Shell** .

The **Debugger Shell** window appears docked to the lower right hand side of debug perspective.

2. In debugger shell, type `config`.

The current configurations settings appear. If you want to change the memory width, type `config MemWidth <value>`. To make sure settings were set, type `config`. The current configuration settings appear.

8.1.2 Can I change the Program Counter (pc) value?

Yes. The pc value can be changed through the debugger shell. In order to change the pc value through the debugger shell follow steps below in debug perspective:

1. From the debug perspective, select **Window > Show View > Debugger Shell** .

The **Debugger Shell** window appears docked to the lower right hand side of debug perspective.

2. In debugger shell, type `reg pc`.

The present value of pc is displayed.

3. In debugger shell, type `reg pc=0x10000000`.

This command changes pc value to `0x10000000`.

4. In debugger shell, type `reg pc`.

The current value of pc is displayed; which in this case should be `PC=0x10000000`.

8.1.3 Does the config page off option still available in the Eclipse-based CodeWarrior IDE?

No. This option is not implemented. If a tcl script needs to be run and you want to advance the debugger shell display, select **Enabling Page** icon from the toolbar. By selecting this option, the debugger shell automatically advances the display without having to press the space bar.

8.1.4 Is there a way to accelerate the execution speed and turn off the printing when I use a lot of mem commands in the debugger shell?

Yes. In order to turn off the printing add the `-np` to the `mem` command in your script file. It should look like the following:

```
mem -np
```



Chapter 9

Profiling and Analysis Tool

In this chapter, you find Frequently Asked Questions (FAQs) related to the CodeWarrior for Power Architecture Tracing and Analysis Tools.

- [FAQs -Tracing and Analysis](#)

9.1 FAQs -Tracing and Analysis

This topic lists tracing and analysis related FAQs.

- [How is tracing useful for the user?](#)
- [How do I enable trace and profiling?](#)
- [How do I collect trace data?](#)
- [How do I view trace data?](#)
- [What are the different types of fields displayed in the Trace viewer and what type of information is provided in them?](#)
- [Where does the trace data get saved?](#)
- [How do I import a trace data file?](#)
- [How do I configure Trace profile\(s\)?](#)
- [What are the various Trace debug connection types I can choose from?](#)
- [What are tracepoints?](#)
- [What are software and hardware tracepoints?](#)
- [What are start and stop tracepoints?](#)
- [How do I view the tracepoints that I have set?](#)
- [What is Performance Analysis?](#)
- [How do I create working sets?](#)
- [How do we add scenarios to a Performance Analysis configuration?](#)
- [What is Performance Analysis perspective used for?](#)
- [What are scenarios?](#)
- **NEW** [What are Nexus Messages?](#)

- **NEW** What is Exclude Symbols feature used for?
- **NEW** What are Custom Scenarios?

9.1.1 How is tracing useful for the user?

Tracing is a technique to obtain diagnostic and performance information about a program's execution. This information is typically used for debugging purposes, and additionally, depending on the type and detail of information contained in a trace log, helps in diagnosing common problems with the software.

9.1.2 How do I enable trace and profiling?

To enable trace and profiling:

1. Open the **Debug Configurations** dialog box.
2. Select your project in the tree structure on the left-hand side.
3. Select the **Trace and Profile** tab.
4. Check the **Enable Trace and Profile** checkbox.

This enables the trace and profiling of your project.

9.1.3 How do I collect trace data?

To collect trace data:

1. Open the **Debug Configurations** dialog box by choosing Run > Debug Configurations from the IDE menu bar.
2. Enable trace and profiling in the **Trace and Profile** tab.
3. Select the desired configuration in the **Trace and Profile** tab.
4. If you want to modify the selected configuration, click **Edit**.

The **Trace Configurations** dialog box appears. You can use this dialog box to create, delete, rename, and modify trace configuration.

5. Debug your application.
6. In the **Debug** view, click **Resume** to start data collection, resume execution and begin measurement.

7. Let the application run until all trace data have been captured for several seconds.
8. Click **Suspend** .

This is how you collect trace data of your application.

9.1.4 How do I view trace data?

To view trace data:

1. From the IDE menu bar, select **Window > Show View > Software Analysis** to open the **Software Analysis** view.
2. Expand the project name.

All trace collections performed for the project are displayed.

3. Click the **Trace** hyperlink.

The trace data appears in the **Trace** viewer. Similarly, you can view Critical Code, Timeline, Performance, and Call Tree data in the respective viewers.

9.1.5 What are the different types of fields displayed in the Trace viewer and what type of information is provided in them?

The **Trace** viewer displays the source code and corresponding linear instructions, or non-executed branch instructions, which have occurred since the last program trace event of the same trace source.

The table below describes the fields of the **Trace** viewer.

Table 9-1. Trace viewer fields

Field	Description
Offset	The first number in the offset specifies the Nexus Message from which the trace event has been decoded. The second number specifies the number of the decoded event, since each Nexus Message can generate 0 to n decoded events.
Event Source	Specifies the trace source that produced the event, such as the Nexus client and trace probe device.
Description	Displays detailed information about the trace line.
Source	Displays the source function of the trace line if it is a call or a branch.

Table continues on the next page...

Table 9-1. Trace viewer fields (continued)

Field	Description
Target	Displays the target function of the trace line if it is a call or a branch.
Type	Specifies the type of event that has occurred, such as Info, Branch, Error, Function Return and Function Call.
Timestamp	Specifies the timestamp value that is expressed as clock ticks. Depending on the configuration of the trace source, this can be relative to the event's trace source or relative to all events logged from all sources

9.1.6 Where does the trace data get saved?

The trace data gets saved in the **.Analysis Data** folder which gets automatically created in the workspace after trace collection.

9.1.7 How do I import a trace data file?

A trace data file is imported from an existing project into another project. You can view the data of the imported file in the **Trace** viewer.

To import trace data:

1. From the IDE menu bar, select **File > Import** .

The **Import** wizard appears.

2. Expand the **Software Analysis** tree node and choose **Trace**. Click **Next**.

The **Import Trace** page appears.

3. Click **Browse** next to the **Trace data file** field to select the trace data file to be imported. The trace data file is located in the **.Analysis Data** folder of the project in the workspace.
4. Select one of the following options:
 - **Existing project** - Imports the trace data file to an existing project present in your current workspace. Selecting this option opens the **Select an Existing Project** page where you can select the project you wish to import the trace data file to.
 - **New project** - Imports the trace data file to a new project.

5. Select the **Copy the file to the workspace** checkbox if you want to maintain a copy of the trace data file in your current workspace.
6. Select the **View the trace data on finish** checkbox to open the **Trace** viewer automatically after clicking the **Finish** button.
7. Click **Next** .

The **Import Trace to Existing Project** page appears.

8. Select the project in which you want to import the trace data, in the **Import trace in project** group.
9. Enter a new file name for the trace data file in the **New trace file name will be** text box. This field is optional.
10. Click **Next**.

The **Import Trace Configuration** page appears.

11. Select the desired hardware for the trace from the **System** pop-up menu.
12. Click **Finish**.

The trace data file is imported to the `.Analysis Data` folder in your project and the **Trace** viewer appears with the imported trace data.

The trace data file is imported to the project.

9.1.8 How do I configure Trace profile(s)?

To configure trace profiles:

1. Open the **Debug Configurations** dialog box.
2. Click **Trace and Profile** to view the trace and profile configuration options.
3. Click **Edit** to view the **Trace Configurations** dialog box in order to modify the selected configuration.

9.1.9 What are the various Trace debug connection types I can choose from?

You can select from the following options available for Gigabit TAP on the **Hardware** page of the **New Project Wizard** :

- **JTAG over JTAG cable** - to send JTAG command over the JTAG cable.

- **JTAG over Aurora cable** - to send JTAG commands over the Aurora cable.
- **Aurora** - to send Aurora commands over the Aurora cable.

NOTE

You can select from **USB TAP** or **Ethernet TAP** connections if you do not wish to collect trace data using Aurora/Nexus.

9.1.10 What are tracepoints?

Tracepoint is a point in the target program where start or stop tracepoints are set at a line of the source code or the assembly code. The start and stop tracepoints are triggers for enabling and disabling the trace output. The advantage of setting start and stop tracepoints is to capture the trace data from the specific part of the program and limiting the collection of trace data to only designated portions of the entire source code.

9.1.11 What are software and hardware tracepoints?

There are two types of tracepoints, Hardware and Software, that can be set on the source code of the target hardware. Based on the available resources a tracepoint may be a hardware or a software tracepoint. By default, CodeWarrior attempts to set a software tracepoint but switch to set a hardware tracepoint if necessary or if directed by the user. Hardware tracepoints use hardware resources to control the trace collection, so no modification is required inside the project. You can set only two hardware tracepoints but multiple software tracepoints. Software tracepoints use interrupts to enable or disable the trace.

9.1.12 What are start and stop tracepoints?

A start tracepoint begins the trace collection from the instruction where it is set. A stop tracepoint ends the trace collection at the instruction where it is set.

9.1.13 How do I view the tracepoints that I have set?

To view tracepoints:

1. From the IDE menu bar, choose **Window > Show View > Other** .
2. In the Show View dialog, expand the **Software Analysis** node and choose **Analysispoints**.
3. Click **OK** to close the **Show View** dialog.

The **Analysispoints** view appears displaying the tracepoints that are set on the program code.

9.1.14 What is Performance Analysis?

The Performance Analysis tool enables you to configure, collect, and analyze scenarios in the QorIQ processors: P4080, P4040, P2040/41, P3041, P5010, P5040, and P5020/21 as well as in the Bx and Tx devices. The tool helps you optimize your bareboard application by using core and platform counters to provide enhanced levels of visibility to the application. This tool is integrated in the CodeWarrior IDE to handle configuration and communication with the target system.

9.1.15 How do I create working sets?

To create working sets:

1. From the IDE menu bar, select **Project > Build Working Set > Select Working Set** .

The **Select Working Set** dialog box appears.

2. Click **New** .

The **Select Working Set Type** dialog box appears.

3. Select the desired working set type and click **Next** .

The **New Working Set** dialog box appears.

4. Enter the name of the working set in the **Working Set Name** text box and click **Finish** .

A new working set is created and is displayed in the **Select Working Set** dialog box.

9.1.16 How do we add scenarios to a Performance Analysis configuration?

To add a scenario to the Performance Analysis configuration:

1. Click **Add a Scenario** in the **Scenarios** pane of the **Configuration** view.

The **Add <Processor> Scenarios** screen appears in which only those scenarios that are applicable to the selected processor are displayed.

2. Select the subsystem that you want to analyze by selecting the corresponding checkbox.
3. Click **OK** to add the selected scenarios to the configuration.
4. Click **File > Save** to save your configuration.

9.1.17 What is Performance Analysis perspective used for?

The Performance Analysis perspective provides a user-friendly interface to perform the scenario measurement activities, such as adding a configuration, configuring connection, defining scenarios, and collecting data for QorIQ processors as well as B4 and T4 devices.

9.1.18 What are scenarios?

A scenario contains the following:

- **Events** - Hardware occurrences that can be counted to any event in the processor, including the output of other counters in a feedback loop.
- **Counters** - Counters that can be connected to any event in the processor, including the output of other counters in a feedback loop.
- **Metrics** - A mathematical combination of counters (which count events) to provide an additional answer.

9.1.19 What are Nexus Messages?

For QorIQ processors, trace is stored in the form of Nexus messages. These messages are translated to higher-level trace events that are displayed in the **Trace** viewer. The Nexus messages can be seen in the **Nexus Messages** view, which is accessible through **Software Analysis** view.

9.1.20 What is Exclude Symbols feature used for?

The Exclude Symbols feature allows you to configure the statistics before or after the data is processed in the **Critical Code**, **Performance**, and **Call Tree** viewers. You can configure the statistics using the **Configure** hyperlink available in the **Software Analysis** view or using the **Exclude Symbols** button available on the toolbar of the **Critical Code**, **Performance**, and **Call Tree** viewers.

This feature excludes the symbols and libraries from the data results. The old statistics will be deleted and the new statistics will be computed based on the configuration, and the viewer will be repopulated with the new results.

9.1.21 What are Custom Scenarios?

The Performance Analysis tool lets you create your own scenarios called custom scenarios by editing the predefined stock scenarios. You can add events, assign counters for them, and compute predefined metrics using the **Custom Scenario Editor**. You can then run the event-counter configuration to generate reports of the event and metric measurements that you have added to the custom scenario.



Index

A

Analysis [183](#)

B

Breakpoints [135](#)

Build Configuration [146](#)

building project [123](#)

By default, the debugger shows me the global variables of the currently debugged source file.

How can I view my variables from another module?
[144](#)

C

Can I change the endianness that is displayed in the Memory view? [169](#)

Can I change the format of the Memory view? [166](#)

Can I change the Program Counter (pc) value? [180](#)

Can I change the Program Counter (pc) value while in the debugger without having to restart the debugger? [174](#)

Can I debug the code from the generated assembly file step-by-step? [165](#)

Can I direct console output to a file instead of the console window? [175](#)

Can I disassemble my source code file? [92](#)

Can I still use Ctrl+Tab keys to navigate between open windows? [87](#)

Can I use a different linker command file than the one being used in my project? [133](#)

Can I use software breakpoints on code running in flash memory? [142](#)

Can the CodeWarrior IDE assist me in writing the source code? [27](#)

Can the icon decorators be combined? [108](#)

Classic CodeWarrior IDE [126](#)

CodeWarrior IDE [115](#)

CodeWarrior project [111](#)

compare [114](#), [124](#), [125](#)

Compiler [131](#)

contents [113](#)

Ctrl + - [126](#)

current workspace [114](#)

Custom Scenarios [191](#)

CVS [58](#)

D

Debugger [135](#)

Debugger Shell [179](#)

directories [125](#)

Does the CodeWarrior IDE support version control systems? [58](#)

Does the config page off option still available in the Eclipse-based CodeWarrior IDE? [180](#)

E

Editor [20](#)

ELF file [127](#)

Exclude Symbols [191](#)

export [122](#)

F

FAQs - Compiler [131](#)

FAQs - Project [119](#)

FAQs - Project Management [111](#)

FAQs -Tracing and Analysis [183](#)

Flash Programmer [152](#)

folder [113](#)

Folder [120](#)

G

GUI [139](#)

H

header [112](#)

How are the PARENT-COUNT-MyVariable definitions defined in the CodeWarrior IDE? [99](#)

How can I add my project to CVS from within the CodeWarrior IDE? [60](#)

How can I add to my project a folder that exists outside the current workspace? [120](#)

How can I can filter the projects listed in the CodeWarrior Projects view based on Working Sets? [103](#)

How can I change a key binding? [80](#)

How can I change the color that highlights the current line in a source code file? [25](#)

How can I change the debugger key bindings to the ones that I used to have in the Classic CodeWarrior IDE? [88](#)

How can I change the name of the executable that is generated when I build my project? [90](#)

How can I change the tab width/size? [29](#)

How can I change the values of variables? [163](#)

How can I comment/uncomment a block of code in the editor? [43](#)

How can I compare files in CodeWarrior IDE? [45](#)

- How can I configure predefined macros in the CodeWarrior IDE? [32](#)
- How can I create the flash programmer, hardware diagnostics, or import/export fill memory tasks in the CodeWarrior IDE? [153](#)
- How can I determine which header files are associated with my source code file? [85](#)
- How can I disable the option of having the CodeWarrior IDE build my project before launching it? [150](#)
- How can I export the launch configurations settings and then import them back into the project? [150](#)
- How can I find out if certain files contain debug information? [95](#)
- How can I find out where a function is declared in my source code? [37](#)
- How can I find the default value of memory width? How can I change the value if I want to? [179](#)
- How can I find the version of the CodeWarrior that I am using? [77](#)
- How can I go back to the previous location if the Editor takes me to another source or header file when, for example, I select the Open Declaration option? [72](#)
- How can I import/export a Working Set in the CodeWarrior eclipse IDE? [104](#)
- How can I improve the performance of the CodeWarrior IDE I am working with? [76](#)
- How can I inspect global variables in the variables view? [177](#)
- How can I modify and save the files in GBK encoding using the CodeWarrior IDE? [98](#)
- How can I open .tcl extension files in the Editor window? [38](#)
- How can I open an existing project in the CodeWarrior IDE? [96](#)
- How can I open definition of a function or header file from within the editor? [22](#)
- How can I program my boards without a project? [157](#)
- How can I quickly open declaration of any variable, function name, macro, or header file from within Editor? [21](#)
- How can I quickly switch to a header file from within the source code file? [32](#)
- How can I resolve the following error message that I get when I start the CodeWarrior IDE? [95](#)
- How can I run a debug session repeatedly? [175](#)
- How can I search for specific files in a workspace? [69](#)
- How can I share same installation of the CodeWarrior software with multiple users? [42](#)
- How can I start the post-build steps in the CodeWarrior IDE? [97](#)
- How can I switch to another workspace? [93](#)
- How can I view addresses in memory? [160](#)
- How can I view a graphical representation of source code in the CodeWarrior IDE using Dot and Doxygen? [83](#)
- How can I view all keyboard shortcuts available in the CodeWarrior IDE? [56](#)
- How can I view only certain registers? [159](#)
- How can I view the hardware breakpoints that have been set? [139](#)
- How can I view variables that I have set in my project? [162](#)
- How does the CodeWarrior IDE differentiate between enabled and disabled macros? [32](#)
- How do I add hardware breakpoints? Can I add them through the GUI? [139](#)
- How do I add to my project a new file that exists outside the current workspace? [122](#)
- How do I collect trace data? [184](#)
- How do I compile my project? [131](#)
- How do I configure the Scalability settings? [109](#)
- How do I configure Trace profile(s)? [187](#)
- How do I convert Line Delimiters to the Unix format? [36](#)
- How do I create a new build configuration for my project? [147](#)
- How do I create working sets? [189](#)
- How do I customize my Workspace perspectives? [52](#)
- How do I debug my application using global variables? [144](#)
- How do I disable existing breakpoints? [137](#)
- How do I edit the keyboard shortcuts in the CodeWarrior IDE? [57](#)
- How do I enable full screen mode in my CodeWarrior IDE? [53](#)
- How do I enable or disable an icon decorator? [107](#)
- How do I enable the incremental build on my project? [127](#)
- How do I enable trace and profiling? [184](#)
- How do I enable watch on a variable? [145](#)
- How do I ensure that the launch configurations are a part of the project when it is exported? [151](#)
- How do I filter variables in files? [145](#)
- How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with? [55](#)
- How do I import a trace data file? [186](#)
- How do I manage the Working Sets? [102](#)
- How do I minimize a Workbench window? [52](#)
- How do I open up multiple memory locations in the memory view? [161](#)
- How do I perform a simple search in the CodeWarrior IDE? [65](#)
- How do I place a breakpoint on a given physical address? [139](#)
- How do I remove blank lines in the Editor view? [49](#)
- How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration? [176](#)

How do I search within the Working Set? [101](#)
 How do I select the type of breakpoint for my application in the Editor view? [143](#)
 How do I set a breakpoint at a particular function? [141](#)
 How do I set breakpoints? [136](#)
 How do I specify the program entry point? [174](#)
 How do I switch between the source files in the Editor view? [48](#)
 How do I uninstall the third party plugins? [19](#)
 How do I update my local history settings? [78](#)
 How do I view call hierarchy in my source code? [86](#)
 How do I view global variables that have been accessed in my project? [163](#)
 How do I view the breakpoints that have been set? [137](#)
 How do I view the tracepoints that I have set? [189](#)
 How do I view trace data? [185](#)
 How is tracing useful for the user? [184](#)
 How the CodeWarrior Project Importer handles recursive access paths? [91](#)
 How to have projects from different workspaces or locations in a workspace? [93](#)

I

I am facing issues in accessing the updater sites. How do I resolve this? [18](#)
 I am unable to specify a JTAG configuration file because the option is disabled. Is there a way to specify it? [152](#)
 I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why? [87](#)
 If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations? [24](#)
 If I have a debugger session running on my application, how can I halt the application? [165](#)
 I need to enable logging for flash programming. How do I do it? [156](#)
 I need to flash a large size application (more than 64 KB) , and I also need to perform target tasks, such as secure/unsecure the device and additional verify or erase. How do I do it? [153](#)
 Installation [17](#)
 instruct [123](#)
 In the Memory view, is it possible to go back to the address that the rendering was created for? [171](#)
 Is it necessary to have the project name identical to the name of the directory that contains the .project file? [97](#)
 Is it possible to change the default code format settings? [30](#)
 Is it possible to condense a large source file by folding/unfolding different portions of code? [40](#)
 Is it possible to configure build settings for a particular source code file? [148](#)
 Is it possible to debug the target multiple times with the same binary application, without flashing the device? [155](#)
 Is it possible to display line numbers in Editor? [30](#)
 Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format? [173](#)
 Is it possible to exclude a single or multiple files from the project while building it? [148](#)
 Is it possible to preview the changes of the Replace action? [65](#)
 Is it possible to restrict the number of resources that appear on my Workbench window? [50](#)
 Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project? [77](#)
 Is it possible to roll back the changes I did to my source code? [34](#)
 Is it possible to specify a different name for the binary file? [165](#)
 Is it possible to store output of the build configuration of my project in a separate directory? [149](#)
 Is it possible to view definition of a macro or a variable in the source code file? [26](#)
 Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE? [33](#)
 Is it possible to view the change history of a source code file? [82](#)
 Is there a file that contains all the launch configuration settings? [91](#)
 Is there a shortcut to open the search window for a specified text that is equivalent to the Ctrl + Shift + M shortcut key available in the Classic CodeWarrior IDE? [72](#)
 Is there a way to accelerate the execution speed and turn off the printing when I use a lot of mem commands in the debugger shell? [180](#)
 Is there a way to add breakpoints to C and Assembly files that exist in a library? [140](#), [164](#)
 Is there a way to change the default color of comments that span across multiple lines? [35](#)
 Is there a way to change the variable or the function name in a CodeWarrior project? [67](#)
 Is there a way to compare files that are not in project? [47](#)
 Is there a way to determine in which workspace I am working currently? [41](#)
 Is there a way to display the variables always in the Hexadecimal format? [168](#)
 Is there a way to filter the settings in the Preferences window to find a particular setting? [80](#)
 Is there a way to instruct the CodeWarrior IDE to use relative paths instead of absolute ones to store the project file location in a workspace? [99](#)
 Is there a way to quickly flash multiple boards with different S-Records? [155](#)

Is there a way to replace a variable easily? [70](#)
Is there a way to revert to the original contents of a source code file? [36](#)
Is there a way to view disassembly at any location? [172](#)
I want to edit my source files, but all the other views are using the space as well. How do I maximize the editor view to have better working area? [48](#)
I want to install updates for the CodeWarrior software. However, when I check for updates, no updates are found. What do I do? [17](#)

L

Linker [133](#)
local history [125](#)

M

Miscellaneous [74](#), [158](#)
modifications [115](#)
My attempt to install the CodeWarrior tool fails with the error "Another version of this product is already installed". What do I do? [19](#)

N

New FAQs [14](#)
Nexus Messages [191](#)

O

object code [126](#)

P

Perspective [114](#)
Profiling [183](#)
project [112](#), [113](#), [120](#)
Project Management [111](#)
project properties [115](#)
projects [114](#)
Property Settings [114](#)

R

related [111](#)
Remote System [115](#)
resize [114](#)

S

Search and Replace Action [64](#)
shortcut key [126](#)
source [112](#)

Source [120](#)
source files [111](#), [113](#), [123–125](#)

T

The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing? [141](#)

U

Updating and Installing Software [17](#)

V

Variables [143](#)
views [114](#)

W

What are Icon decorators? [107](#)
What are scenarios? [190](#)
What are software and hardware tracepoints? [188](#)
What are start and stop tracepoints? [188](#)
What are the different types of fields displayed in Trace Events viewer and what type of information is provided in them? [185](#)
What are the various Trace debug connection types I can choose from? [187](#)
What are tracepoints? [188](#)
What controls the initial layout of a view in a Workbench window? [50](#)
What does an asterisk on the Editor title bar mean? [37](#)
What does the checkmark next to the breakpoint represent? [138](#)
What does the Reset on Connect option do? [174](#)
What if I want to delete a breakpoint? [138](#)
What is a hardware breakpoint? [143](#)
What is a perspective? [77](#)
What is a static local variable? [146](#)
What is a Working Set? How do I build/compile a Working Set? [100](#)
What is Performance Analysis? [189](#)
What is Performance Analysis perspective used for? [190](#)
What is the difference between a breakpoint and a line breakpoint? [141](#)
What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other? [140](#)
What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options? [138](#)
What is the first thing that I see when I start the CodeWarrior IDE? [76](#)

- What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button? [96](#)
- What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window? [140](#)
- What is the purpose of the Outline view? [95](#)
- What is the purpose of the Properties view? [95](#)
- What is the purpose of the Tasks view? [93](#)
- When I compile my project, where can I see the commands that are being executed? [132](#)
- When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do? [146](#)
- When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, "You are opening a large file. Scalability mode has been turned on for this editor to help improve performance by disabling features requiring to parse the source file language." What does it mean? [109](#)
- When I save my launch configurations as a local file and then delete the project, all the local configurations get deleted as well. How can I resolve this issue? [98](#)
- Where does the CodeWarrior IDE save the debug configuration as a local file by default? [97](#)
- Where does the trace data get saved? [186](#)
- Where do I see the warning and error messages? [132](#)
- Where exactly the disassemble file gets created? [92](#)
- Which breakpoint option do I select if I want to add a breakpoint? [138](#)
- Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list? [138](#)
- Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings? [73](#)
- While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do? [142](#)
- While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug? [126](#)
- Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective? [168](#)
- Why does the Rename option appear grayed out? [87](#)
- Why do I get the following error message when I attempt to debug my project? [166](#)
- Why do I get the following error message when I try to edit a source code file? [37](#)
- Why duplicating a configuration in the debugger perspective does not duplicate the run configuration? [87](#)
- Why the.metadata folder in my workspace stores a huge history? [92](#)
- Why the console view does not display all the warning and error messages by default? [90](#)
- Why the project that I just created is not visible in the CodeWarrior Projects view? [86](#)
- Workbench Window [49](#)
- workspace [113](#), [114](#)



How to Reach Us:

Home Page:

freescale.com

Web Support:

freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. Freescale reserves the right to make changes without further notice to any products herein.

Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale 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 that may be provided in Freescale 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. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, CodeWarrior, QorIQ, StarCore are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. QorIQ Qonverge is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2011–2016 Freescale Semiconductor, Inc.

