# Contents

<table>
<thead>
<tr>
<th>Section number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Contents of this Manual</td>
<td>13</td>
</tr>
<tr>
<td>1.2</td>
<td>Accompanying Documentation</td>
<td>14</td>
</tr>
<tr>
<td>1.3</td>
<td>New FAQs</td>
<td>14</td>
</tr>
<tr>
<td>2.1</td>
<td>Editor</td>
<td>17</td>
</tr>
<tr>
<td>2.1.1</td>
<td>How can I quickly open declaration of any variable, function name, macro, or header file from within Editor?</td>
<td>18</td>
</tr>
<tr>
<td>2.1.2</td>
<td>If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations?</td>
<td>18</td>
</tr>
<tr>
<td>2.1.3</td>
<td>How can I change the color that highlights the current line in a source code file?</td>
<td>20</td>
</tr>
<tr>
<td>2.1.4</td>
<td>Is it possible to view definition of a macro or a variable in the source code file?</td>
<td>21</td>
</tr>
<tr>
<td>2.1.5</td>
<td>Can the CodeWarrior IDE assist me in writing the source code?</td>
<td>22</td>
</tr>
<tr>
<td>2.1.6</td>
<td>How can I change the tab width/size?</td>
<td>23</td>
</tr>
<tr>
<td>2.1.7</td>
<td>Is it possible to display line numbers in Editor?</td>
<td>24</td>
</tr>
<tr>
<td>2.1.8</td>
<td>Is it possible to change the default code format settings?</td>
<td>25</td>
</tr>
<tr>
<td>2.1.9</td>
<td>How can I quickly switch to a header file from within the source code file?</td>
<td>26</td>
</tr>
<tr>
<td>2.1.10</td>
<td>How does the CodeWarrior IDE differentiate between enabled and disabled macros?</td>
<td>26</td>
</tr>
<tr>
<td>2.1.11</td>
<td>How can I configure predefined macros in the CodeWarrior IDE?</td>
<td>26</td>
</tr>
<tr>
<td>2.1.12</td>
<td>Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE?</td>
<td>27</td>
</tr>
<tr>
<td>2.1.13</td>
<td>Is it possible to roll back the changes I did to my source code?</td>
<td>28</td>
</tr>
<tr>
<td>2.1.14</td>
<td>Is there a way to change the default color of comments that span across multiple lines?</td>
<td>29</td>
</tr>
<tr>
<td>2.1.15</td>
<td>Is there a way to revert to the original contents of a source code file?</td>
<td>30</td>
</tr>
<tr>
<td>2.1.16</td>
<td>How do I convert Line Delimiters to the Unix format?</td>
<td>31</td>
</tr>
<tr>
<td>2.1.17</td>
<td>Why do I get the following error message when I try to edit a source code file?</td>
<td>31</td>
</tr>
<tr>
<td>2.1.18</td>
<td>What does an asterisk on the Editor title bar mean?</td>
<td>32</td>
</tr>
<tr>
<td>2.1.19</td>
<td>How can I find out where a function is declared in my source code?</td>
<td>32</td>
</tr>
</tbody>
</table>
Section number | Title | Page
--- | --- | ---
2.1.20 | How can I open .tcl extension files in the Editor window? | 32
2.1.21 | How do I switch between the source files in the Editor view? | 34
2.1.22 | 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? | 35
2.1.23 | How do I remove blank lines in the Editor view? | 35
2.2 | Workbench Window | 36
2.2.1 | What controls the initial layout of a view in a Workbench window? | 37
2.2.2 | Is it possible to restrict the number of resources that appear on my Workbench window? | 37
2.2.3 | How do I minimize a Workbench window? | 38
2.2.4 | How do I customize my Workspace perspectives? | 39
2.2.5 | How do I enable full screen mode in my CodeWarrior IDE? | 40
2.2.6 | How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with? | 42
2.2.7 | How can I view all keyboard shortcuts available in the CodeWarrior IDE? | 42
2.2.8 | How do I edit the keyboard shortcuts in the CodeWarrior IDE? | 43
2.3 | CVS | 44
2.3.1 | Does the CodeWarrior IDE support version control systems? | 44
2.3.2 | How can I add my project to CVS from within the CodeWarrior IDE? | 46
2.4 | Search and Replace Action | 50
2.4.1 | How do I perform a simple search in the CodeWarrior IDE? | 50
2.4.2 | Is it possible to preview the changes of the Replace action? | 51
2.4.3 | Is there a way to change the variable or the function name in a CodeWarrior project? | 53
2.4.4 | How can I search for specific files in a workspace? | 55
2.4.5 | Is there a way to replace a variable easily? | 56
2.4.6 | 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? | 57
2.4.7 | 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? | 57
2.4.8 | Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings? | 58
<table>
<thead>
<tr>
<th>Section number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>Miscellaneous...................................................................................</td>
<td>59</td>
</tr>
<tr>
<td>2.5.1</td>
<td>How can I improve the performance of the CodeWarrior IDE I am working with?</td>
<td>61</td>
</tr>
<tr>
<td>2.5.2</td>
<td>What is the first thing that I see when I start the CodeWarrior IDE?</td>
<td>61</td>
</tr>
<tr>
<td>2.5.3</td>
<td>What is a perspective?</td>
<td>62</td>
</tr>
<tr>
<td>2.5.4</td>
<td>How can I find the version of the CodeWarrior that I am using?</td>
<td>62</td>
</tr>
<tr>
<td>2.5.5</td>
<td>Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project?</td>
<td>62</td>
</tr>
<tr>
<td>2.5.6</td>
<td>How do I update my local history settings?</td>
<td>63</td>
</tr>
<tr>
<td>2.5.7</td>
<td>Is there a way to filter the settings in the Preferences window to find a particular setting?</td>
<td>64</td>
</tr>
<tr>
<td>2.5.8</td>
<td>How can I change a key binding?</td>
<td>65</td>
</tr>
<tr>
<td>2.5.9</td>
<td>Is it possible to view the change history of a source code file?</td>
<td>67</td>
</tr>
<tr>
<td>2.5.10</td>
<td>How can I determine which header files are associated with my source code file?</td>
<td>68</td>
</tr>
<tr>
<td>2.5.11</td>
<td>How do I view call hierarchy in my source code?</td>
<td>69</td>
</tr>
<tr>
<td>2.5.12</td>
<td>Why the project that I just created is not visible in the CodeWarrior Projects view?</td>
<td>70</td>
</tr>
<tr>
<td>2.5.13</td>
<td>I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why?</td>
<td>70</td>
</tr>
<tr>
<td>2.5.14</td>
<td>Why does the Rename option appear grayed out?</td>
<td>70</td>
</tr>
<tr>
<td>2.5.15</td>
<td>Why duplicating a configuration in the debugger perspective does not duplicate the run configuration?</td>
<td>71</td>
</tr>
<tr>
<td>2.5.16</td>
<td>Can I still use Ctrl+Tab keys to navigate between open windows?</td>
<td>71</td>
</tr>
<tr>
<td>2.5.17</td>
<td>How can I change the debugger key bindings to the ones that I used to have in the Classic CodeWarrior IDE?</td>
<td>71</td>
</tr>
<tr>
<td>2.5.18</td>
<td>How can I change the name of the executable that is generated when I build my project?</td>
<td>73</td>
</tr>
<tr>
<td>2.5.19</td>
<td>Why the console view does not display all the warning and error messages by default?</td>
<td>73</td>
</tr>
<tr>
<td>2.5.20</td>
<td>Is there a file that contains all the launch configuration settings?</td>
<td>74</td>
</tr>
<tr>
<td>2.5.21</td>
<td>How the CodeWarrior Project Importer handles recursive access paths?</td>
<td>74</td>
</tr>
<tr>
<td>2.5.22</td>
<td>Can I disassemble my source code file?</td>
<td>75</td>
</tr>
<tr>
<td>2.5.23</td>
<td>Where exactly the disassemble file gets created?</td>
<td>75</td>
</tr>
<tr>
<td>2.5.24</td>
<td>Why the.metadata folder in my workspace stores a huge history?</td>
<td>75</td>
</tr>
<tr>
<td>2.5.25</td>
<td>How to have projects from different workspaces or locations in a workspace?</td>
<td>76</td>
</tr>
<tr>
<td>2.5.26</td>
<td>How can I switch to another workspace?</td>
<td>76</td>
</tr>
<tr>
<td>Section number</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>2.5.27</td>
<td>What is the purpose of the Tasks view?</td>
<td>76</td>
</tr>
<tr>
<td>2.5.28</td>
<td>What is the purpose of the Properties view?</td>
<td>78</td>
</tr>
<tr>
<td>2.5.29</td>
<td>What is the purpose of the Outline view?</td>
<td>78</td>
</tr>
<tr>
<td>2.5.30</td>
<td>How can I find out if certain files contain debug information?</td>
<td>78</td>
</tr>
<tr>
<td>2.5.31</td>
<td>How can I resolve the following error message that I get when I start the CodeWarrior IDE?</td>
<td>78</td>
</tr>
<tr>
<td>2.5.32</td>
<td>How can I open an existing project in the CodeWarrior IDE?</td>
<td>78</td>
</tr>
<tr>
<td>2.5.33</td>
<td>What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button?</td>
<td>79</td>
</tr>
<tr>
<td>2.5.34</td>
<td>How can I start the post-build steps in the CodeWarrior IDE?</td>
<td>79</td>
</tr>
<tr>
<td>2.5.35</td>
<td>Is it necessary to have the project name identical to the name of the directory that contains the .project file?</td>
<td>80</td>
</tr>
<tr>
<td>2.5.36</td>
<td>Where does the CodeWarrior IDE save the debug configuration as a local file by default?</td>
<td>80</td>
</tr>
<tr>
<td>2.5.37</td>
<td>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?</td>
<td>80</td>
</tr>
<tr>
<td>2.5.38</td>
<td>How can I modify and save the files in GBK encoding using the CodeWarrior IDE?</td>
<td>81</td>
</tr>
<tr>
<td>2.5.39</td>
<td>How are the PARENT-COUNT-MyVariable definitions defined in the CodeWarrior IDE?</td>
<td>82</td>
</tr>
<tr>
<td>2.5.40</td>
<td>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?</td>
<td>82</td>
</tr>
<tr>
<td>2.5.41</td>
<td>What is a Working Set? How do I build/compile a Working Set?</td>
<td>82</td>
</tr>
<tr>
<td>2.5.42</td>
<td>How do I search within the Working Set?</td>
<td>84</td>
</tr>
<tr>
<td>2.5.43</td>
<td>How do I manage the Working Sets?</td>
<td>85</td>
</tr>
<tr>
<td>2.5.44</td>
<td>How can I filter the projects listed in the CodeWarrior Projects view based on Working Sets?</td>
<td>86</td>
</tr>
<tr>
<td>2.5.45</td>
<td>How can I import/export a Working Set in the CodeWarrior eclipse IDE?</td>
<td>87</td>
</tr>
<tr>
<td>2.5.46</td>
<td>What are Icon decorators?</td>
<td>89</td>
</tr>
<tr>
<td>2.5.47</td>
<td>How do I enable or disable an icon decorator?</td>
<td>90</td>
</tr>
<tr>
<td>2.5.48</td>
<td>Can the icon decorators be combined?</td>
<td>91</td>
</tr>
<tr>
<td>2.5.49</td>
<td>When I open a large file in CodeWarrior IDE, an Editor Scalability dialog box appears, listing the message, &quot;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.&quot; What does it mean?</td>
<td>91</td>
</tr>
<tr>
<td>2.5.50</td>
<td>How do I configure the Scalability settings?</td>
<td>92</td>
</tr>
</tbody>
</table>
### Chapter 3
Project Management

3.1 FAQs - Project Management

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.1</td>
<td>How can I find out where the source files related to a CodeWarrior project are saved?</td>
<td>95</td>
</tr>
<tr>
<td>3.1.2</td>
<td>How can I add another source or header file to my project?</td>
<td>96</td>
</tr>
<tr>
<td>3.1.3</td>
<td>What is a workspace?</td>
<td>97</td>
</tr>
<tr>
<td>3.1.4</td>
<td>What is a project?</td>
<td>97</td>
</tr>
<tr>
<td>3.1.5</td>
<td>How can I view the contents of various types of source code files that appear in my project folder?</td>
<td>97</td>
</tr>
<tr>
<td>3.1.6</td>
<td>Is there a way to compare the Property Settings of two different projects?</td>
<td>97</td>
</tr>
<tr>
<td>3.1.7</td>
<td>How do I resize the various views in a Perspective?</td>
<td>98</td>
</tr>
<tr>
<td>3.1.8</td>
<td>How can I add a project from some other workspace to my current workspace?</td>
<td>98</td>
</tr>
<tr>
<td>3.1.9</td>
<td>How do I ensure that the CodeWarrior IDE saves all modifications that I make to the project properties?</td>
<td>99</td>
</tr>
</tbody>
</table>

### Chapter 4
Project

4.1 FAQs - Project

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.1</td>
<td>How do I add a new Source Folder to my project?</td>
<td>102</td>
</tr>
<tr>
<td>4.1.2</td>
<td>How can I add to my project a folder that exists outside the current workspace?</td>
<td>102</td>
</tr>
<tr>
<td>4.1.3</td>
<td>How do I add to my project a new file that exists outside the current workspace?</td>
<td>103</td>
</tr>
<tr>
<td>4.1.4</td>
<td>How can I export a project that I could send to someone else?</td>
<td>103</td>
</tr>
<tr>
<td>4.1.5</td>
<td>How do I instruct the CodeWarrior IDE to save changes to my source code files automatically before building the project?</td>
<td>104</td>
</tr>
<tr>
<td>4.1.6</td>
<td>How do I compare two source code files?</td>
<td>105</td>
</tr>
<tr>
<td>4.1.7</td>
<td>How do I compare two directories?</td>
<td>105</td>
</tr>
<tr>
<td>4.1.8</td>
<td>How can I specify the number of days to keep local history for source code files?</td>
<td>106</td>
</tr>
<tr>
<td>4.1.9</td>
<td>How can I see or change the preprocessor macros defined for my project?</td>
<td>106</td>
</tr>
<tr>
<td>4.1.10</td>
<td>Is there a shortcut to remove object code that is equivalent to the Ctrl + - shortcut key available in the Classic CodeWarrior IDE?</td>
<td>108</td>
</tr>
<tr>
<td>4.1.11</td>
<td>While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?</td>
<td>108</td>
</tr>
<tr>
<td>4.1.12</td>
<td>While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?</td>
<td>109</td>
</tr>
</tbody>
</table>
### Chapter 5
**Compiler**

#### 5.1 FAQs - Compiler

- **5.1.1** How do I compile my project? .......................... 113
- **5.1.2** Where do I see the warning and error messages? .......................................................... 114
- **5.1.3** When I compile my project, where can I see the commands that are being executed? ........ 114

### Chapter 6
**Linker**

#### 6.1 FAQs - Linker

- **6.1.1** Can I use a different linker command file than the one being used in my project? .............. 115

### Chapter 7
**Debugger**

#### 7.1 Breakpoints

- **7.1.1** How do I set breakpoints? ............................... 118
- **7.1.2** How do I view the breakpoints that have been set? .......................................................... 118
- **7.1.3** How do I disable existing breakpoints? .......................................................... 119
- **7.1.4** What does the checkmark next to the breakpoint represent? ........................................ 119
- **7.1.5** What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options? ......................... 120
- **7.1.6** Which breakpoint option do I select if I want to add a breakpoint? .................................. 120
- **7.1.7** Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list? .......................................................................................................................... 120
- **7.1.8** What if I want to delete a breakpoint? ........................................................................ 120
- **7.1.9** How do I place a breakpoint on a given physical address? ................................................. 120
- **7.1.10** How do I add hardware breakpoints? Can I add them through the GUI? .......................... 121
- **7.1.11** How can I view the hardware breakpoints that have been set? ...................................... 121
- **7.1.12** What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other? .................................................................................. 122
<table>
<thead>
<tr>
<th>Section number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1.13</td>
<td>What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window?</td>
<td>122</td>
</tr>
<tr>
<td>7.1.14</td>
<td>Is there a way to add breakpoints to C and Assembly files that exist in a library?</td>
<td>122</td>
</tr>
<tr>
<td>7.1.15</td>
<td>How do I set a breakpoint at a particular function?</td>
<td>122</td>
</tr>
<tr>
<td>7.1.16</td>
<td>What is the difference between a breakpoint and a line breakpoint?</td>
<td>123</td>
</tr>
<tr>
<td>7.1.17</td>
<td>The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing?</td>
<td>123</td>
</tr>
<tr>
<td>7.1.18</td>
<td>While debugging in RAM my debugger halts and reports an &quot;illegal breakpoint&quot; error. What do I do?</td>
<td>124</td>
</tr>
<tr>
<td>7.1.19</td>
<td>Can I use software breakpoints on code running in flash memory?</td>
<td>124</td>
</tr>
<tr>
<td>7.1.20</td>
<td>What is a hardware breakpoint?</td>
<td>125</td>
</tr>
<tr>
<td>7.1.21</td>
<td>How do I select the type of breakpoint for my application in the Editor view?</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>7.2 Variables</td>
<td>125</td>
</tr>
<tr>
<td>7.2.1</td>
<td>By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?</td>
<td>126</td>
</tr>
<tr>
<td>7.2.2</td>
<td>How do I debug my application using global variables?</td>
<td>126</td>
</tr>
<tr>
<td>7.2.3</td>
<td>How do I filter variables in files?</td>
<td>127</td>
</tr>
<tr>
<td>7.2.4</td>
<td>How do I enable watch on a variable?</td>
<td>127</td>
</tr>
<tr>
<td>7.2.5</td>
<td>What is a static local variable?</td>
<td>128</td>
</tr>
<tr>
<td>7.2.6</td>
<td>When I enable watch on a static variable, it shows the error, &quot;&lt;error(s) during the evaluation&gt;&quot;. What do I do?</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>7.3 Build Configuration</td>
<td>128</td>
</tr>
<tr>
<td>7.3.1</td>
<td>How do I create a new build configuration for my project?</td>
<td>129</td>
</tr>
<tr>
<td>7.3.2</td>
<td>Is it possible to exclude a single or multiple files from the project while building it?</td>
<td>130</td>
</tr>
<tr>
<td>7.3.3</td>
<td>Is it possible to configure build settings for a particular source code file?</td>
<td>130</td>
</tr>
<tr>
<td>7.3.4</td>
<td>Is it possible to store output of the build configuration of my project in a separate directory?</td>
<td>131</td>
</tr>
<tr>
<td>7.3.5</td>
<td>How can I disable the option of having the CodeWarrior IDE build my project before launching it?</td>
<td>132</td>
</tr>
<tr>
<td>7.3.6</td>
<td>How can I export the launch configurations settings and then import them back into the project?</td>
<td>132</td>
</tr>
<tr>
<td>7.3.7</td>
<td>How do I ensure that the launch configurations are a part of the project when it is exported?</td>
<td>133</td>
</tr>
<tr>
<td>7.3.8</td>
<td>Is it possible to debug the target multiple times with the same binary application, without flashing the device?</td>
<td>133</td>
</tr>
<tr>
<td>Section number</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>7.3.9</td>
<td>I need to enable logging for flash programming. How do I do it?</td>
<td>134</td>
</tr>
<tr>
<td>7.4</td>
<td>Miscellaneous</td>
<td>135</td>
</tr>
<tr>
<td>7.4.1</td>
<td>How can I view only certain registers?</td>
<td>136</td>
</tr>
<tr>
<td>7.4.2</td>
<td>How can I view addresses in memory?</td>
<td>137</td>
</tr>
<tr>
<td>7.4.3</td>
<td>How do I open up multiple memory locations in the memory view?</td>
<td>138</td>
</tr>
<tr>
<td>7.4.4</td>
<td>How can I view variables that I have set in my project?</td>
<td>139</td>
</tr>
<tr>
<td>7.4.5</td>
<td>How can I change the values of variables?</td>
<td>140</td>
</tr>
<tr>
<td>7.4.6</td>
<td>How do I view global variables that have been accessed in my project?</td>
<td>140</td>
</tr>
<tr>
<td>7.4.7</td>
<td>Is the command-line debugger available in the Eclipse-based CodeWarrior IDE as well?</td>
<td>141</td>
</tr>
<tr>
<td>7.4.8</td>
<td>Is it possible to specify a different name for the binary file (.eld)?</td>
<td>141</td>
</tr>
<tr>
<td>7.4.9</td>
<td>Can I debug the code from the generated assembly file step-by-step?</td>
<td>142</td>
</tr>
<tr>
<td>7.4.10</td>
<td>If I have a debugger session running on my application, how can I halt the application?</td>
<td>142</td>
</tr>
<tr>
<td>7.4.11</td>
<td>Why do I get the following error message when I attempt to debug my project?</td>
<td>143</td>
</tr>
<tr>
<td>7.4.12</td>
<td>Can I change the format of the Memory view?</td>
<td>143</td>
</tr>
<tr>
<td>7.4.13</td>
<td>Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective?</td>
<td>145</td>
</tr>
<tr>
<td>7.4.14</td>
<td>Is there a way to display the variables always in the Hexadecimal format?</td>
<td>145</td>
</tr>
<tr>
<td>7.4.15</td>
<td>Can I change the endianness that is displayed in the Memory view?</td>
<td>146</td>
</tr>
<tr>
<td>7.4.16</td>
<td>In the Memory view, is it possible to go back to the address that the rendering was created for?</td>
<td>148</td>
</tr>
<tr>
<td>7.4.17</td>
<td>Is there a way to view disassembly at any location?</td>
<td>149</td>
</tr>
<tr>
<td>7.4.18</td>
<td>Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format?</td>
<td>149</td>
</tr>
<tr>
<td>7.4.19</td>
<td>What does the Reset on Connect option do?</td>
<td>151</td>
</tr>
<tr>
<td>7.4.20</td>
<td>Can I change the Program Counter (pc) value while in the debugger without having to restart the debugger?</td>
<td>151</td>
</tr>
<tr>
<td>7.4.21</td>
<td>How do I specify the program entry point?</td>
<td>151</td>
</tr>
<tr>
<td>7.4.22</td>
<td>How can I run a debug session repeatedly?</td>
<td>152</td>
</tr>
<tr>
<td>7.4.23</td>
<td>Can I direct console output to a file instead of the console window?</td>
<td>152</td>
</tr>
<tr>
<td>7.4.24</td>
<td>How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration?</td>
<td>153</td>
</tr>
</tbody>
</table>
# Chapter 8
## Debugger Shell

8.1 FAQs - Debugger Shell

<table>
<thead>
<tr>
<th>Section number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1.1</td>
<td>How can I find the default value of memory width? How can I change the value if I want to?</td>
<td>155</td>
</tr>
<tr>
<td>8.1.2</td>
<td>Can I change the Program Counter (pc) value?</td>
<td>156</td>
</tr>
<tr>
<td>8.1.3</td>
<td>Does the config page off option still available in the Eclipse-based CodeWarrior IDE?</td>
<td>156</td>
</tr>
<tr>
<td>8.1.4</td>
<td>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?</td>
<td>156</td>
</tr>
</tbody>
</table>

# Chapter 9
## Profiling and Analysis Tool

9.1 FAQs - Profiling

<table>
<thead>
<tr>
<th>Section number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1.1</td>
<td>How do I enable trace and profiling?</td>
<td>160</td>
</tr>
<tr>
<td>9.1.2</td>
<td>How do I view trace and other data?</td>
<td>161</td>
</tr>
<tr>
<td>9.1.3</td>
<td>How can I save my trace results for later use?</td>
<td>161</td>
</tr>
<tr>
<td>9.1.4</td>
<td>How can I open the DataFile Editor?</td>
<td>161</td>
</tr>
<tr>
<td>9.1.5</td>
<td>What are tracepoints?</td>
<td>161</td>
</tr>
<tr>
<td>9.1.6</td>
<td>What are software and hardware tracepoints?</td>
<td>162</td>
</tr>
<tr>
<td>9.1.7</td>
<td>What are start and stop tracepoints?</td>
<td>162</td>
</tr>
<tr>
<td>9.1.8</td>
<td>Which option do I select if I want to add a tracepoint?</td>
<td>162</td>
</tr>
<tr>
<td>9.1.9</td>
<td>How do I view the tracepoints that I have set?</td>
<td>162</td>
</tr>
<tr>
<td>9.1.10</td>
<td>What is Performance Analysis perspective used for?</td>
<td>162</td>
</tr>
<tr>
<td>9.1.11</td>
<td>What are scenarios?</td>
<td>163</td>
</tr>
<tr>
<td>9.1.12</td>
<td>Is the Performance Analysis tool available in two different versions?</td>
<td>163</td>
</tr>
<tr>
<td>9.1.13</td>
<td>What are counterpoints?</td>
<td>163</td>
</tr>
<tr>
<td>9.1.14</td>
<td>How can I configure DPU counters?</td>
<td>164</td>
</tr>
<tr>
<td>9.1.15</td>
<td>How can I change the value of the OCE registers based on the OCE configurator settings?</td>
<td>164</td>
</tr>
<tr>
<td>9.1.16</td>
<td>What all registers will be affected by the settings in the Profiler Configuration panel?</td>
<td>164</td>
</tr>
<tr>
<td>9.1.17</td>
<td>What is the difference between counted events and count cycles using the OCE counters?</td>
<td>165</td>
</tr>
<tr>
<td>9.1.18</td>
<td>When tracing with VTB (DPU/OCE) is done all writes go directly out on the bus without passing the MMU but does not go directly to the memory. Is privilege signal required while writing on the bus?</td>
<td>165</td>
</tr>
<tr>
<td>Section number</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>9.1.19</td>
<td>How can you check the profiling results during a debug session?</td>
<td>165</td>
</tr>
<tr>
<td>9.1.20</td>
<td>What are Register Analyzers?</td>
<td>165</td>
</tr>
<tr>
<td>9.1.21</td>
<td>What are different types of Register Analyzer?</td>
<td>166</td>
</tr>
<tr>
<td>9.1.22</td>
<td>What does the number of VLES executed represents in the result?</td>
<td>166</td>
</tr>
<tr>
<td>9.1.23</td>
<td>What is the use of remote launch in CodeWarrior scripts?</td>
<td>166</td>
</tr>
<tr>
<td>9.1.24</td>
<td>When using default 8144ADS stationary project all DPU registers display hex number &quot;FF&quot;. Also, when I try to change the value of register it does not allow to change. Why I can not change the value of DPU registers?</td>
<td>167</td>
</tr>
<tr>
<td>9.1.25</td>
<td>Does Trace and Profile work with SmartDSP OS? I tried to use the Trace and Profile tools to see the PC Trace Subroutine and interrupt calls and returns, but I was not able to. Is there a limitation or setting that I need to set?</td>
<td>167</td>
</tr>
<tr>
<td>9.1.26</td>
<td>What is the speed of SmartDSP HEAT? Can the SmartDSP HEAT provide higher speed than HSST when doing trace offload?</td>
<td>167</td>
</tr>
<tr>
<td>9.1.27</td>
<td>How to get VLES and Cycle Count Numbers using hardware DPU Counters in CodeWarrior?</td>
<td>168</td>
</tr>
<tr>
<td>9.1.28</td>
<td>What is Linker Command File and how to enable trace in it?</td>
<td>171</td>
</tr>
<tr>
<td>9.1.29</td>
<td>How the type of memory and the size of VTB is determined?</td>
<td>172</td>
</tr>
<tr>
<td>9.1.30</td>
<td>When you should use VTB write modes: Overwrite, One Address, or Trace Event Request?</td>
<td>174</td>
</tr>
<tr>
<td>9.1.31</td>
<td>How the trace is extracted using JTAG, HSST, HEAT, Ethernet method and how they work?</td>
<td>175</td>
</tr>
<tr>
<td>9.1.32</td>
<td>How many hardware and software tracepoints are available?</td>
<td>175</td>
</tr>
<tr>
<td>9.1.33</td>
<td>How can a tracepoint affect the application?</td>
<td>176</td>
</tr>
<tr>
<td>9.1.34</td>
<td>Does trace collection affect my application performance and how?</td>
<td>176</td>
</tr>
<tr>
<td>9.1.35</td>
<td>What does it mean to collect events with counters?</td>
<td>177</td>
</tr>
<tr>
<td>9.1.36</td>
<td>What is the relationship between DPU and trace?</td>
<td>177</td>
</tr>
<tr>
<td>9.1.37</td>
<td>Why it is important to import offline trace data?</td>
<td>178</td>
</tr>
<tr>
<td>9.1.38</td>
<td>What trace configuration method do I use automatic or manual?</td>
<td>179</td>
</tr>
<tr>
<td>9.1.39</td>
<td>How can you view the latest trace without terminating the debug session?</td>
<td>180</td>
</tr>
</tbody>
</table>
Chapter 1
Introduction

This FAQ Guide lists most frequently asked or most anticipated questions about CodeWarrior Development Studio for StarCore 3900FP DSP 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 and Ethernet TAP.

In this chapter:

- Contents of this Manual - Describes the contents of this manual.
- New FAQs - 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>
<thead>
<tr>
<th>Chapter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDE</td>
<td>Lists IDE related FAQs</td>
</tr>
<tr>
<td>Project Management</td>
<td>Lists project management related FAQs</td>
</tr>
<tr>
<td>Project</td>
<td>Lists project related FAQs</td>
</tr>
<tr>
<td>Compiler</td>
<td>Lists compiler related FAQs</td>
</tr>
<tr>
<td>Linker</td>
<td>Lists linker related FAQs</td>
</tr>
<tr>
<td>Debugger</td>
<td>Lists debugger related FAQs</td>
</tr>
<tr>
<td>Debugger Shell</td>
<td>Lists debugger shell related FAQs</td>
</tr>
<tr>
<td>Profiling and Analysis Tool</td>
<td>Lists profiler related FAQs</td>
</tr>
</tbody>
</table>
1.2 Accompanying Documentation

The Documentation page describes the documentation included in this version of CodeWarrior Development Studio for StarCore 3900FP DSP Architectures.

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\SC\Help folder.

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

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

How do I enable the incremental build on my project?

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

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

What is a hardware breakpoint?

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

By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?

How do I debug my application using global variables?

How do I filter variables in files?

How do I enable watch on a variable?

What is a static local variable?

When I enable watch on a static variable, it shows the error, "<error(s)_during_the_evaluation>". What do I do?
Chapter 2
IDE

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:

• Editor
• Workbench Window
• CVS
• Search and Replace Action
• Miscellaneous

2.1 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?
• 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 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?
How do I switch between the source files in the Editor view?
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?
How do I remove blank lines in the Editor view?

2.1.1 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.

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 2-1. Open Declaration
2.1.2 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:

![Figure 2-2. Hyperlink in Source Files](image)

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

The modifier key changes, as specified.

2.1.3 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.
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.

### 2.1.4 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.
2.1.5 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.

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.
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.

You can also view the list of bitfields using Content Assist, as shown in the figure listed below.
2.1.6 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.

   ![Text Editors Preferences Page](image)

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

The default tab size changes.

2.1.7 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.

### 2.1.8 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 2-11. Code Style Preference Page](image)

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.

### 2.1.9 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.

### 2.1.10 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.

```
if PL_RDD_PULLUP_RESISTOM /* then check on bit cleared */
#define RBD_TEST_MACRO(a,b) (a&b)
else
#define RBD_TEST_MACRO(a,b) (a&b)
#endif
```

**Figure 2-12. Disabled Macros Marked in Gray**
2.1.11 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.

2.1.12 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).

Figure 2-13. Pop-up Window Displaying Macro Expansion

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

2.1.13 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.
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.

### 2.1.14 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**.
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.

### 2.1.15 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.

**2.1.16 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.

**2.1.17 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.
2.1.18 What does an asterisk on the Editor title bar mean?

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

2.1.19 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.

2.1.20 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.
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.

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

![Preferences Dialog Box - File Association](image)

**Figure 2-20. Preferences Dialog Box - File Association**

9. Click **OK**.

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

### 2.1.21 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.
2. Press **Ctrl+Tab** repeatedly to iterate through the source files list.

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

### 2.1.22 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.

### 2.1.23 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.

  ![Find/Replace Dialog Box](image)

  **Figure 2-22. 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.

### 2.2 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?
- How do I customize my Workspace perspectives?
- How do I enable full screen mode in my CodeWarrior IDE?
- How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with?
• How can I view all keyboard shortcuts available in the CodeWarrior IDE?
• How do I edit the keyboard shortcuts in the CodeWarrior IDE?

2.2.1 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.

2.2.2 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).
8. Click **Finish** to close the **New Working Set** wizard.

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

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.

### 2.2.3 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.

2.2.4 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 - \(<\text{perspective name}'>\) dialog box appears.

   ![Customize Perspective - <perspective name> Dialog Box](image)

   **Figure 2-25.** Customize Perspective - \(<\text{perspective name}'>\) 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 - \(<\text{perspective name}'>\) 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.
5. Select **OK**.
6. The **Overwrite Perspective** dialog box appears.
7. Click **Yes** to overwrite the previous default setting with the changes you made.

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

To change your CodeWarrior IDE to full screen mode:

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 2-27. 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 2-28. 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.

2.2.6 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 2-29. 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 2-30. Show Toolbar - Menu Option

2.2.7 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.

Figure 2-31. Keyboard Shortcuts Drop-down List

2.2.8 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.
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.

2.3 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?

2.3.1 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.

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).
You can now use CVS with the CodeWarrior IDE.

2.3.2 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 Project with 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.
3. Specify name of the module in the CVS repository and click Next.

The Share Project Resources page appears.

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.

6. 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 2-37. Specify How to Store Unknown Files](image)

7. Click **Next**.

The **Commit** page appears.

![Figure 2-38. Commit Page](image)

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.
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).

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).
The Synchronizer view displays what has been changed either locally or remotely. For example, in the Synchronizer 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.

![Synchronizer View Displaying File Revisions](image)

Figure 2-42. Synchronizer View Displaying File Revisions

### 2.4 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 do I need to backup or store in a version control system to completely recover project settings and related debug settings?

#### 2.4.1 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**.

### 2.4.2 Is it possible to preview the changes of the Replace action?

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

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).

   ![Search Menu](image)
   
   **Figure 2-43. Search Menu**

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

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.
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.

### 2.4.3 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.
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.

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

The specified changes are performed.
2.4.4 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).

   ![Search Dialog Box](image)

   **Figure 2-50. 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.
2.4.5 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.

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

3. Click **Replace**.

   The **Replace Text Matches** dialog box appears.
4. Click **Preview**.

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

2.4.6 **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.
2.4.7 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 2-55. Preferences Dialog Box - Keys Page](image)

**NOTE**

For more information on key binding, refer to the topic How can I change a key binding?

2.4.8 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.

2.5 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 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?
• What is a Working Set? How do I build/compile a Working Set?
• How do I search within the Working Set?
• How do I manage the Working Sets?
• How can I can filter the projects listed in the CodeWarrior Projects view based on Working Sets?
• How can I import/export a Working Set in the CodeWarrior eclipse IDE?
• What are Icon decorators?
• How do I enable or disable an icon decorator?
• 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?

2.5.1 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.
2.5.2 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.

2.5.3 What is a perspective?

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

2.5.4 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.

2.5.5 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.

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

   The selected files are restored.

### 2.5.6 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 2-56. Preferences Dialog Box - Local History Page](image)
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.

2.5.7 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 2-57. 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

2.5.8 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.
3. Select **General > Editors > Keys**.

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

4. From the **Scheme** drop-down list select the required scheme, such as **Microsoft Visual Studio** or **Emacs**.

5. Type **content assist** as the filter text to filter the list of keys.
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.

### 2.5.9 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.
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.

2.5.10 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 2-63. Include Browser View](image)

### 2.5.11 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 2-64. Call Hierarchy View](image)

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

   The callees appear in the view.
2.5.12 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.

2.5.13 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.

2.5.14 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.

2.5.15 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.

2.5.16 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.

2.5.17 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.
   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**.

![Preferences Dialog Box-Debugger Keys](image)

**Figure 2-66. 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:

1. Select **Step Over** in the **Command** column.
2. From the **Binding** textbox, remove F6 by deleting it from the binding text box.
3. 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.

![Preferences Dialog Box-New Key Binding](image)

**Figure 2-67. Preferences Dialog Box-New Key Binding**
d. 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.

### 2.5.18 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.
2.5.19 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.

2.5.20 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.

2.5.21 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:
2.5.22 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 `sourcefilename`, where represent random numbers.

2.5.23 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 `sourcefilename` file.
2.5.24 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?

2.5.25 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.

2.5.26 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.
2.5.27 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.

   ![Properties Dialog Box](image)

   **Figure 2-68. 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.
2.5.28  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.

2.5.29  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.

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

2.5.31  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.

2.5.32  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** 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.

**2.5.33** 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.

**2.5.34** 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`.

### 2.5.35 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.

### 2.5.36 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.

### 2.5.37 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.

![Preferences Dialog Box - Launch Configuration](image)

Figure 2-69. Preferences Dialog Box - Launch Configuration

4. Click **OK**.

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

### 2.5.38 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**.
2.5.39 **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).

2.5.40 **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.

2.5.41 **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.

![Select Working Set Dialog Box](image)

**Figure 2-70. 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.

![New Working Set - C/C++ Working Set Page](image)

**Figure 2-71. 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.

![Select Working Set Dialog Box with Newly Created Working Set](image)

**Figure 2-72. Select Working Set Dialog Box with Newly Created Working Set**

7. Click **OK**.

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

### 2.5.42 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.
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.

### 2.5.43 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.
2.5.44 How can I 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:
2.5.45 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.
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:
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:
2.5.46 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 2-79. Example - Decorator for Application.c File

2.5.47 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:
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.

---

**2.5.48 Can the icon decorators be combined?**

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

![Example - Combined Icon Decorators](image)

---

**2.5.49 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.

### 2.5.50 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.
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.

In this topic:

- 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?

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.
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?
- How can I see or change the preprocessor macros defined for my project?
- Is there a shortcut to remove object code that is equivalent to the Ctrl + - shortcut key available in the Classic CodeWarrior IDE?
- While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug?
- While building a project, the CodeWarrior fails to write an ELF file. How do I resolve this issue?
- How do I enable the incremental build on my project?
- 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.
5. Click Browse.
   The Browse for Folder window appears.
6. Select the folder that you want to add to the project.
7. Click OK.
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.
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 How can I see or change the preprocessor macros defined for my project?

To see or change the preprocessor macros defined for your project, follow steps below:

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 panel appears on the right-hand side.

   
   The Macros panel appears on the right-hand side displaying #define preprocessor macros.

5. Make the required changes and click Apply.

6. Click OK.
The preprocessor macros defined for your project changes as specified.

4.1.10 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.

4.1.11 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.
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 StarCore:

- 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.

**4.1.12 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.
4.1.13 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.

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.

6. Click Ok.

### 4.1.14 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.
2. Select **Configure Contents** from the menu.

   The **Configure Contents** dialog box appears.

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.
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. Click Apply.

5. 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
- Variables
- Build Configuration
- 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?
• While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do?
• Can I use software breakpoints on code running in flash memory?
• What is a hardware breakpoint?
• 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?
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 **Breakpoint Type > C/C++ Hardware Breakpoints**.
3. Select **Toggle Breakpoint** at the line where you want the breakpoint.

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 `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.

4. In the Show in area, clear the Vertical ruler checkbox, as shown in the following figure.
5. Click **Apply**.
6. 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 allow 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](image)

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.
variables

By default, the debugger shows me the global variables of the currently debugged source file. How can I view my variables from another module?

How do I debug my application using global variables?

How do I filter variables in files?

How do I enable watch on a variable?

What is a static local variable?

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 current thread 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.
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?
- 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?

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.

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](image)

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.

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 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.


   ![Figure 7-7. Download Page - Debug Configurations Dialog Box](image)

   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.
7.3.9 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 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 (.eld)?
- 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?

7.4.1 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.
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.

### 7.4.2 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 7-9. Monitor Memory Window](image)
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 7-10. Memory View](image)

### 7.4.3 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.
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 7-12. Memory View with Multiple Addresses](image)

### 7.4.4 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.

### 7.4.5 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.

### 7.4.6 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.

![Add Globals Dialog Box](image)

**Figure 7-13. 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.

### 7.4.7 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.
7.4.8 Is it possible to specify a different name for the binary file (.eld)?

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 .eld. 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.

7.4.9 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.

7.4.10 If I have a debugger session running on my application, how can I halt the application?
In the **Debug** perspective, click on the .eld being executed. From toolbar, select **Suspend** from the toolbar which halts the application temporarily. To run the application, select **Resume** from toolbar.

### 7.4.11 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.

### 7.4.12 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.

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.
5. Right-click anywhere in the **Memory** view
   
   A context menu appears.

6. From context menu, select **Format**.
   
   The **Format** window appears.

   ![Format Dialog Box](image)

   **Figure 7-16. 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.

![Updated Memory View](image)

**Figure 7-17. Updated Memory View**

### 7.4.13 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.

### 7.4.14 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.

When Variables view appears in the debug perspective, the variables appear in the Hexadecimal format.

NOTE

This setting is a global setting.

7.4.15 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.

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 7-19. Memory View with Multiple Addresses

5. From the Memory Toolbar, select New Renderings.

The New Renderings window appears.

6. Select Traditional.

Figure 7-20. Memory View-New Renderings

7. Click Add Renderings.
8. Click OK.

The Traditional view appears in Memory view.
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 7-22. Memory View-Little Endian Addresses](image)

**NOTE**

To change the endianness back to **Big**, right-click anywhere in the **Traditional** tab, and select **Endian > Big** from the context menu.

7.4.16 **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.

### 7.4.17 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 7-23. Memory View-Assembly Instructions](image-url)
7.4.18 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).

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

   When **Variables** view appears in the debug perspective, the variables appear in the Hexadecimal format.

![Figure 7-24. Preferences Window-Debug Page](image-url)
NOTE
This setting is a global setting.

7.4.19 What does the Reset on Connect option do?

The Reset on Connect option resets all the cores.

7.4.20 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.

7.4.21 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 option.
6. Click **Apply**.
7. Click **Debug**.

### 7.4.22 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**.

### 7.4.23 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.
6. Click **Apply**.
7. Click **Debug**.

The console output will be directed to the specified file.

### 7.4.24 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.


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.
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 Development for StarCore v10.x.

• FAQs - Profiling

9.1 FAQs - Profiling

This topic lists tracing and analysis related FAQs.

• How do I enable trace and profiling?
• How do I view trace and other data?
• How can I save my trace results for later use?
• How can I open the DataFile Editor?
• What are tracepoints?
• What are software and hardware tracepoints?
• What are start and stop tracepoints?
• Which option do I select if I want to add a tracepoint?
• How do I view the tracepoints that I have set?
• What is Performance Analysis perspective used for?
• What are scenarios?
• Is the Performance Analysis tool available in two different versions?
• What are counterpoints?
• How can I configure DPU counters?
• How can I change the value of the OCE registers based on the OCE configurator settings?
• What all registers will be affected by the settings in the Profiler Configuration panel?
• What is the difference between counted events and count cycles using the OCE counters?
9.1.1 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.

### 9.1.2 How do I view trace and other data?

To view trace and other data:

1. From the IDE menu bar, select **Profiler > All Results** to open the **Trace and Profile Results** view.
2. Expand the data source under the project name.
3. Expand the **Current Results** data file to view the Trace and other data sets for the collected data.
4. Double-click the Trace data set node to view the results in the Trace Data viewer.

This is how you can view trace and other data.

### 9.1.3 How can I save my trace results for later use?

To save the trace results, right-click on the **Current Results** node, in the Trace and Profile Results viewer and select **Save Current Results**.

### 9.1.4 How can I open the DataFile Editor?

To open the DataFile Editor, double-click on the data file, for example, **Current Results**.

### 9.1.5 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.
9.1.6 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. To determine whether a tracepoint is software or hardware, it is based on OCE resources availability, trace configuration and target program code. You can set maximum six hardware tracepoints. Software tracepoints use interrupts to enable or disable the trace and you can have any number of software tracepoints.

9.1.7 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.

9.1.8 Which option do I select if I want to add a tracepoint?

If you want to skip a tracepoint, but still want to have it in your list, right-click on the Marker bar where Start or Stop tracepoint is already set and enabled, select the Disable Tracepoint option from the tracepoint context menu.

9.1.9 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 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.
9.1.10 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 B4 and T4 devices.

9.1.11 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.

9.1.12 Is the Performance Analysis tool available in two different versions?

Yes, the performance analysis tool is available in two different versions: one is used for configuring, collecting, and analyzing events in the QorIQ processors: P4080, P4040, P2040/41, P3041, P5010, P5040, and P5020/21.

And the other one is used for configuring, collecting, and analyzing scenarios in the B4 and T4 devices.

9.1.13 What are counterpoints?

Counterpoints allow you to count events between two points of the executed code. The results are computed based on the collected trace data, the counterpoints are searched in trace by their addresses.
9.1.14 How can I configure DPU counters?

The DPU counters are configured in the Debug Configuration window. To configure counters:

1. Right-click on your project name, context menu appears.
2. From the context menu, select **Debug As > Debug Configurations**. The Debug Configurations window appears.
3. Select the project name executable from the CodeWarrior Download.
4. Select the **Trace and Profile** tab.
5. Click the **Advanced Settings** button. In the Preferences screen, select **DPU settings** and you should be able to configure DPU counters.

   This is how you can configure DPU counters.

9.1.15 How can I change the value of the OCE registers based on the OCE configurator settings?

To set the OCE30 registers to the value specified in the OCE Configuration, you need to click on the **Change** button after you have initialized the EDCAn registers. It may happen that the **Change** button is not visible when you open the OCE Configurator (depending on the layout and screen resolution you are using). To activate the changes:

1. Set up EDCAn triggers in the OCE configurator
2. Maximize the OCE Configurator view
3. Restore OCE configurator to its original size

   You should see the updated register values. Note that you can alternatively resize the OCE Configurator window and make it a little bit higher.

9.1.16 What all registers will be affected by the settings in the Profiler Configuration panel?
The settings in the profiler configuration panel will change the register settings of OCE, DPU, and MMU (only the M_CR Register).

**9.1.17 What is the difference between counted events and count cycles using the OCE counters?**

Counted events means profiling the cycle information either from the OCE counters or from the DPU counters. When the profiler configuration is set to Count Cycles using the OCE, the cycle count information will come only from the OCE counters.

**9.1.18 When tracing with VTB (DPU/OCE) is done all writes go directly out on the bus without passing the MMU but does not go directly to the memory. Is privilege signal required while writing on the bus?**

The privileged signal is propagated on to the bus as there are parts of the SOC which can only be written to in privileged mode (primarily CCSR). The DPU bypassed the MMU. Although there is no use for it as the main memories used for trace are not protected in general.

**9.1.19 How can you check the profiling results during a debug session?**

Yes, the trace data is collected only after the debug session has terminated. This means that you may have to wait for a long period of time depending upon the size and complexity of the application.

CodeWarrior for StarCore offers the ability to check the DPU (Debugging and Profiling Unit) results at run-time in the DPU Analyzer view. To open this view, select **Window > Show view > Other > Analysis > DPU Analyzer**. The DPU Analyzer shows the data when the program is halted. This can be used to trace the performance of the source code.
9.1.20 What are Register Analyzers?

Register Analyzer analyzes the state of the registers on the target. It helps you to see if there are errors in the register configuration, if there are disabled hardware units or to view informative messages. This feature is supported on both 8144ADS and 8156ADS targets. The two working modes of register analyzer are online and offline.

9.1.21 What are different types of Register Analyzer?

The two types of Register Analyzer are online and offline. The registers on the target can be analyzed online in an active debug session. The debug session is identified by the name of the launch configuration. The registers can be analyzed offline by loading a file that contains the register values. The registers can be analyzed based on a file that contains the information of all the registers.

9.1.22 What does the number of VLES executed represents in the result?

The number of VLES executed depends on various profiler configurations:

1. In case of PACC simulator, the value represents the number of VLES that are executed.
2. In case of profiling on hardware using OCE and tracing all instruction set, the value represents the number of VLES that are executed.
3. In case of profiling on hardware using OCE and tracing COF, the value represent the number of COF and not the number of VLES that are executed.
4. In case of profiling on hardware using DPU counters and tracing call/return, the value represent the number of call/return and not the number of VLES that are executed.

This is how you can view VLES in the result.

9.1.23 What is the use of remote launch in CodeWarrior scripts?


You can execute your own scripts in CodeWarrior IDE. The scripts are launched in remote launch and debugger shell. The Python scripts are executed in remote launch and TCL scripts are executed in debugger shell. The remote launch feature of CodeWarrior allows launch configurations to be executed remotely. You can use the debugger shell to execute commands in a command-line environment.

9.1.24 When using default 8144ADS stationary project all DPU registers display hex number "FF". Also, when I try to change the value of register it does not allow to change. Why I can not change the value of DPU registers?

If you are using a stationary project, you can not write to the DPU registers. The VTB and the OCE registers must be set up appropriately to modify the DPU registers.

9.1.25 Does Trace and Profile work with SmartDSP OS? I tried to use the Trace and Profile tools to see the PC Trace Subroutine and interrupt calls and returns, but I was not able to. Is there a limitation or setting that I need to set?

The SmartDSP OS Linker Command File does not include a definition for _VTB_start and _VTB_end. When these settings are not set, you cannot use the Compute VTB Location Automatically option. You need to define the _VTB_start and _VTB_End addresses manually.

9.1.26 What is the speed of SmartDSP HEAT? Can the SmartDSP HEAT provide higher speed than HSST when doing trace offload?

HSST uses the JTAG connection which is significantly slower than HEAT that uses Ethernet connection. The SmartDSP HEAT blocks the program while preparing the packet. During the I/O, it gives packet to the QUICC Engine for transmission with core intervention. The speed of transmission is 11.6Mbps/core on the MSC8156ADS. This is the limitation of the server capabilities and not the DSP offloading capabilities.
9.1.27 How to get VLES and Cycle Count Numbers using hardware DPU Counters in CodeWarrior?

To view VLES and Cycle count numbers in the result:

1. In the debug launch configuration, check the **Enable Trace and Profile** checkbox to enable the tracing and profiling.
2. In the Counted Events panel, select **Use DPU Counters** radio button and set the value of **DPU Configuration Settings** to **Custom** and then click **Advanced Settings**.

The Preferences screen appears.

3. In the Preferences screen, select DPU settings from the list and then select **DPU Triad A Settings** tab. Select **Use Predefined Counter settings** radio button and set
the value **Breakdown of application cycles - Group 1** from the pull-down menu.

4. In the DPU Triad B Settings tab, select **Use Predefined Counter settings** radio button and set the value **BTB Characterization Group 1** from the pull-down menu.

Counter B0 of this group counts the total number of execution set.

5. Click **OK** to close the Preferences window.
6. Click **Apply** and **Debug** to debug an application.
7. Click **Resume** to start the measurement. When you terminate or suspend the debug session, the tool generates the profiler reports.

If you are interested in the exclusive numbers then you need to look at the critical code data. In the critical code data, you should be able to view:

- Application cycle column shows the exclusive total number of cycles count per function.
• No Bubble column shows exclusive time spent executing function instructions. This number must be equal or more than "Total Number of execution set" and equal or less than "Application cycles".

• Total number of execution set shows the total number of instruction set (VLES) per function. You can export the data to CSV format.

If you need more detailed information such as inclusive, minimum, and maximum then you need to look at the performance data. Based on the selected metric, the tool outputs the data. For example, if you want to know the minimum and maximum number of execution set for a given function, you need to set **Metrics** to **Total number of execution set** and then browse the data to locate the desired function. You can also export the data to CSV format.
A Linker Command File (LCF) contains all the memory configurations and settings on which the linker relies to assemble the executable file (*.eld or *.elb). Using a linker command file (LCF), you can instruct the linker to store portions of your executable in different areas of memory. The linker relocates and binds symbols to addresses according to the directives in your LCF.

NOTE
The following scenario assumes that Enable Trace and Profile option is not selected in the Software Analysis Trace and Profile page while creating the project. If this option is already selected, you can still change the default settings from LCF.

To enable trace in LCF:

1. Expand the LCF folder of your project in the CodeWarrior Projects view.
2. Open the mmu_attr.l3k file.
3. Scroll down to the end of file, and set the value of _ENABLE_VTB as follows: 

   ```
   _ENABLE_VTB = 3;
   ```

   This will enable trace and profile in LCF and will reserve VTB in DDR memory.
NOTE
You can set the value of _ENABLE_VTB as 1, 2, or 3 depending on whether you want to reserve VTB in M1, M2, or DDR memory respectively. Setting any other value will disable the trace and profile.

4. Save the file.

This is how you can enable trace in the Linker Command File.

NOTE
The size of the reserved VTB can be increased or decreased in the common.13k file. _M2_VTB_size= 0x1000; //4K for each core
_M3_VTB_size= 0x8000; //32K for each core
_DDR_VTB_size=0x1000; //4k for each core

9.1.29 How the type of memory and the size of VTB is determined?

You can use MemoryMapViewer to identify free memory zones. To view a contiguous memory area that is not reserved for another purpose, click Window > Show View > Other > MemoryMapViewer. You can set the size of the memory depending on the size of the M1, M2, or DDR memory and VTB start address in the common.13k file.

NOTE
The common.13k file available in the LCF folder of your project.

To check the available free memory block:

1. Open the Properties page of your project.
2. Expand C/C++ Build > Settings.
3. In the Tools Settings tab, expand StarCore C/C++ Linker Application > Linker Settings.
4. In the **Map File** text box on the right, type a name for the map file.

5. Click **Apply**.

6. Close the Properties page.

7. Build your project.

8. The map file appears in the C_Debug_8156_HW folder of your project.

9. Open the map file.

10. Scroll down or search until you find the Memory Map. This is the sample:

    **Memory Map:**

    ```plaintext
    |0x30000000..0x30007fff| memory reserved (physical memory) (32768 bytes)
    rw- |0x30008000..0x30019fff| memory m2_private_data_c_wb (73728 bytes)
    |0x3001a000..0x30037fff| memory free space (122880 bytes)
    rw- |0x30038000..0x3003ffff| memory data_boot_c (32768 bytes)
    |0x40000000..0x5fffffff| memory free space (536870912 bytes)
    |0x80000000..0x9fffffff| memory free space (536870912 bytes)
    r-x |0xc0000000..0xc00093ff| memory m3_shared_text_c (37888 bytes)
    ```
When you should use VTB write modes: Overwrite, One Address, or Trace Event Request?

The order in which VTB write modes are arranged below is according to their usefulness.

Trace Event Request mode - In this mode, when the trace buffer is full, a debug request or an interrupt is generated, which allows the tools to collect the trace and prevent data from getting overwritten. This mode specifies the address within the VTB range where an interrupt or debug request will be generated when Write Mode is set to Trace Event Request. The address at which the interrupt will occur, the data collected till that address will be transferred to the computer.

After the trace is collected, the application is automatically resumed making this process transparent to the user.

Overwrite mode - This mode is useful for investigating crashes, as it captures the information about the code that was executed just before the crash. In this mode, the last state of application is retrieved, it means the last VTB (Virtual Trace Buffer) filled by the
application. Because of small size of VTB, data is overwritten every time when it gets full. You can collect and view the trace when the debug session is suspended by the user or due to a breakpoint or where the bug appears.

One Address mode - This mode is used when you select Trace Configuration Method as Manual settings done by user code. To control which data is downloaded to the computer, this option is used. It can be used if the trace information is written to a peripheral device.

9.1.31 How the trace is extracted using JTAG, HSST, HEAT, Ethernet method and how they work?

All these methods use an interrupt instead of a debug instruction. When the trace buffer is full, which means core will not stop when the trace is being collected.

It means the core will not stop while the trace is being collected. These download methods are differentiated based on how the trace is downloaded from the target.

High Speed Simultaneous Transfer (HSST) is a method to facilitate data transfer between an application that runs on a target (for example, MSC8156 target) and a host side CodeWarrior application. HSST download method uses the same physical communication layer as JTAG download method, but the speed of the HSST method is 12.82 times faster than the JTAG download method. While the JTAG download method transfers data by halting the target, exchanging the data followed by resuming the target, the HSST download method works faster because the transfer between target and host computer occurs without stopping the target. JTAG is very slow and does not require additional hardware support.

HEAT requires an ethernet physical connection, an operating system that can manage file I/O (Freescale SDOS) and SmartDSP OS server on the host, but it provides the highest speed. It is used for streaming large data to and from PC. The transfer of data that takes hours on JTAG takes seconds on HEAT.

Ethernet uses the physical support and the NetIO software libraries. It is as fast as HEAT and unlike HEAT, it does not require any specific operating system.

9.1.32 How many hardware and software tracepoints are available?
Hardware tracepoints use OCE channels, which are non-intrusive but are in limited number (only 6 channels) and are also included in the resources shared with the debugger. Software tracepoints use interrupts, which means they might affect execution speed and you may have to modify the application in order to use them. The advantage is that you can use any number of software tracepoints.

Both hardware and software tracepoints can be set on the source code of the target hardware. Hardware tracepoints are non-intrusive and does not affect the execution speed of an application. There can be maximum of six hardware tracepoints because there are only six OCE channels present on the chip. If you have set the start and stop hardware tracepoint, debug the application and open the OCE Configurator view. In the OCE Configurator view, there are six EDCA0-5 channels showing you whether the EDCA channel is enabled or disabled and also the addresses. This address is passed through the data bus to the processor. EDCA0 channel will check this address in data bus and if start tracepoint is set on that address, tracing will start. EDCA1 channel will check for the second address in data bus and encounters stop tracepoint at that address, tracing will stop.

Software tracepoints use interrupts to enable or disable the trace. You can have any number of start and stop tracepoints.

A demo on how software tracepoints works can be found at: <InstallDir>\StarCore_Support\SoftwareAnalysis\swandemo_81xxads_tracepoints

9.1.33 How can a tracepoint affect the application?

Hardware tracepoints does not affect the application and they are non-intrusive. Software tracepoints can affect the execution speed of an application and they are intrusive. CPU allocates resources to generate and execute the software interrupts.

9.1.34 Does trace collection affect my application performance and how?

To minimize the effects of tracing on application performance, consider the following options:

- Count only the events of interest. Multiple debug sessions can be carried out automatically to obtain trace data for other events. Tracing the events individually
means that Trace Write Buffer (TWB) fills slowly and less writes to VTB are required.

- Use VTB of the appropriate size for the download method.
- Recall that the core stalls while the trace is downloaded. A larger-sized VTB generates less debug interrupts when transferring its content to the host PC. Therefore, the goal is to reduce the number of times the VTB must be emptied onto the host PC. In an ideal scenario, the VTB should have enough capacity to save all the events from a debug session, so that a VTB download occurs only when the debug session terminates. The goal is to minimize the number of downloads required to transfer the trace data.
- Use M2 or M3 memory when available.
- Use start or stop tracepoints to take the measurements only of the critical code data. This results in less data that must be saved in the VTB.

### 9.1.35 What does it mean to collect events with counters?

The specialized unit called DPU has 6 counters that are used to measure different events, such as execution cycles, stalls, cache events, holds due to VTB writes, bus load, BTB characterization and others. All the events that can be measured are available in the Software Analysis GUI configurator.

When such event occurs, an interrupt occurs inside the DPU and the actual value of the counter associated with that event is saved. This snapshot of the counter is then saved into the VTB along with the address that has generated the event (e.g. calling a function cause a jump to a particular address. Returning from the function is causing another jump to another address. The two snapshots of the counter associated with the jumps helps to compute the time needed for the execution of a function).

This mechanism that consists of associating an address with a counter value is the basis of collecting the event.

### 9.1.36 What is the relationship between DPU and trace?

At hardware level, the unit responsible for data or code events collection is the DPU. The DPU writes the trace information that is generated either by itself or the On Chip Emulator (OCE), to a Virtual Trace Buffer (VTB) as shown in figure below. The VTB can reside in system memory or, in internal or external memory.
The DPU write accesses are buffered in the Trace Write Buffer (TWB), and then written through the main data bus interface in bursts. The lowest priority bus requests are used when writing into the first unwritten address of the VTB. When TWB becomes full, it raises the priority of its bus request. At the first cycle after this priority change, the DPU generates a core stall request. When the writes into the VTB reaches the VTB Trace Event Request Address, it triggers a debug interrupt and the data is downloaded into the host PC.

### 9.1.37 Why it is important to import offline trace data?

The offline trace data is used to diagnose the issue or reason for crash in the application. You do not require large memory for collecting the trace data since VTB acts as a circular buffer that is overwritten until the application crashes. The buffer will contain the last data when the application was crashed. This data will help you to diagnose the reason for crashing of an application. When an error occurs, the target is halted and the data from the VTB is saved from the target into a file. If CodeWarrior debugger connection is not available, an alternate way to save the data should be used. For example, the Ethernet connection can also be used to dump the data to a PC host.

After the file has been saved onto the host, the data can be examined to identify the cause for failure.

To save the VTB data in .rawtrace file:

1. Select **Windows > Show View > Other > Debug > Memory** to open the **Memory** view.
2. Click the **Export** icon present on the toolbar of this view.
3. **Export Memory** screen appears.

![Export Memory dialog box](image)

4. Enter the VTB Start and End addresses and also the format in which you want the data to be saved.

5. Browse the location you want the data to be saved and click **OK**.

This is how you can import offline trace data.

9.1.38 **What trace configuration method do I use automatic or manual?**

It depends on your needs that how do you want to configure the tracing using CodeWarrior GUI or from the code of your application.

This setting affects only the way the configuration is set and applied to the target. For both options, you will have tracing and profiling results available.

If you select Auto - settings done by CW, all the settings made in the GUI configurator are applied to the trace hardware modules by CodeWarrior when the execution of the application is started. All DPU registers settings are done automatically by CodeWarrior as shown in figure below.

![Configuration dialog box](image)
In case of Manual - settings done by user code, you need to ensure that the application sets the DPU registers correctly from the code before capturing any trace. The register values must be set manually from source code before debugging your application.

9.1.39 How can you view the latest trace without terminating the debug session?

In case of large applications, you can see the results for a particular function without terminating the debug session. This helps you to view the results without waiting for long time.
Index

A
Analysis 159

B
Breakpoints 117
Build Configuration 128
building project 104
By default, the debugger shows me the global
variables of the currently debugged source file.
How can I view my variables from another module? 126

C
Can I change the endianness that is displayed in the
Memory view? 146
Can I change the format of the Memory view? 143
Can I change the Program Counter (pc) value? 156
Can I change the Program Counter (pc) value while
in the debugger without having to restart the
debugger? 151
Can I debug the code from the generated assembly
file step-by-step? 142
Can I direct console output to a file instead of the
console window? 152
Can I disassemble my source code file? 75
Can I still use Ctrl+Tab keys to navigate between open windows? 71
Can I use a different linker command file than the one being used in my project? 115
Can I use software breakpoints on code running in flash memory? 124
Can the CodeWarrior IDE assist me in writing the source code? 22
Can the icon decorators be combined? 91
Classic CodeWarrior IDE 108
CodeWarrior project 95
compare 98, 105, 106
Compiler 113
contents 97
Ctrl + - 108
current workspace 98
CVS 44

D
Debugger 117
Debugger Shell 155
directories 106
Does the CodeWarrior IDE support version control
systems? 44
Does the config page off option still available in the
Eclipse-based CodeWarrior IDE? 156
Does Trace and Profile work with SmartDSP OS? I tried to use the Trace and Profile tools to see the PC
Trace Subroutine and interrupt calls and returns, but
I was not able to. Is there a limitation or setting that
I need to set? 167
Does trace collection affect my application
performance and how? 176

E
Editor 17
ELF file 109
export 103

F
FAQs - Compiler 113
FAQs - Profiling 159
FAQs - Project 101
FAQs - Project Management 95
folder 97
Folder 102

G
GUI 121

H
header 96
How are the PARENT-COUNT-MyVariable
definitions defined in the CodeWarrior IDE? 82
How can a tracepoint affect the application? 176
How can I add my project to CVS from within the
CodeWarrior IDE? 46
How can I add to my project a folder that exists
outside the current workspace? 102
How can I filter the projects listed in the
CodeWarrior Projects view based on Working Sets? 86
How can I change a key binding? 65
How can I change the color that highlights the
current line in a source code file? 20
How can I change the debugger key bindings to the
ones that I used to have in the Classic CodeWarrior
IDE? 71
How can I change the name of the executable that is
generated when I build my project? 73
How can I change the tab width/size? 24
How can I change the value of the OCE registers based on the OCE configurator settings? 164
How can I change the values of variables? 140
How can I configure DPU counters? 164
How can I configure predefined macros in the CodeWarrior IDE? 27
How can I determine which header files are associated with my source code file? 68
How can I disable the option of having the CodeWarrior IDE build my project before launching it? 132
How can I export the launch configurations settings and then import them back into the project? 132
How can I find out if certain files contain debug information? 78
How can I find out where a function is declared in my source code? 32
How can I find the default value of memory width? 89
How can I change the value if I want to? 155
How can I find the version of the CodeWarrior that I am using? 62
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? 57
How can I import/export a Working Set in the CodeWarrior eclipse IDE? 87
How can I improve the performance of the CodeWarrior IDE I am working with? 61
How can I modify and save the files in GBK encoding using the CodeWarrior IDE? 81
How can I open .tcl extension files in the Editor window? 32
How can I open an existing project in the CodeWarrior IDE? 78
How can I open the DataFile Editor? 161
How can I quickly open declaration of any variable, function name, macro, or header file from within Editor? 18
How can I quickly switch to a header file from within the source code file? 26
How can I resolve the following error message that I get when I start the CodeWarrior IDE? 78
How can I run a debug session repeatedly? 152
How can I save my trace results for later use? 161
How can I search for specific files in a workspace? 55
How can I see or change the preprocessor macros defined for my project? 107
How can I start the post-build steps in the CodeWarrior IDE? 79
How can I switch to another workspace? 76
How can I view addresses in memory? 137
How can I view all keyboard shortcuts available in the CodeWarrior IDE? 42
How can I view only certain registers? 136
How can I view the hardware breakpoints that have been set? 121
How can I view variables that I have set in my project? 139
How can you check the profiling results during a debug session? 165
How can you view the latest trace without terminating the debug session? 180
How does the CodeWarrior IDE differentiate between enabled and disabled macros? 26
How do I add hardware breakpoints? Can I add them through the GUI? 121
How do I add to my project a new file that exists outside the current workspace? 103
How do I compile my project? 113
How do I configure the Scalability settings? 92
How do I convert Line Delimiters to the Unix format? 31
How do I create a new build configuration for my project? 129
How do I customize my Workspace perspectives? 39
How do I debug my application using global variables? 126
How do I disable existing breakpoints? 119
How do I edit the keyboard shortcuts in the CodeWarrior IDE? 43
How do I enable full screen mode in my CodeWarrior IDE? 40
How do I enable or disable an icon decorator? 90
How do I enable the incremental build on my project? 110
How do I enable trace and profiling? 160
How do I enable watch on a variable? 127
How do I ensure that the launch configurations are a part of the project when it is exported? 133
How do I filter variables in files? 127
How do I hide the toolbar from the IDE for better visibility of other views that I am currently working with? 42
How do I manage the Working Sets? 85
How do I minimize a Workbench window? 38
How do I open up multiple memory locations in the memory view? 138
How do I perform a simple search in the CodeWarrior IDE? 50
How do I place a breakpoint on a given physical address? 120
How do I remove blank lines in the Editor view? 35
How do I resolve the following error message that I get when I use Ctrl-Click to open variable declaration? 153
How do I search within the Working Set? 84
How do I select the type of breakpoint for my application in the Editor view? 125
How do I set a breakpoint at a particular function? 123
How do I set breakpoints? 118
How do I specify the program entry point? 151
How do I switch between the source files in the Editor view? 34
How do I update my local history settings? 63
How do I view call hierarchy in my source code? 69
How do I view global variables that have been accessed in my project? 140
How do I view the breakpoints that have been set? 118
How do I view the tracepoints that I have set? 162
How do I view trace and other data? 161
How many hardware and software tracepoints are available? 175
How the CodeWarrior Project Importer handles recursive access paths? 74
How the trace is extracted using JTAG, HSST, HEAT, Ethernet method and how they work? 175
How the type of memory and the size of VTB is determined? 172
How to get VLES and Cycle Count Numbers using hardware DPU Counters in CodeWarrior? 168
How to have projects from different workspaces or locations in a workspace? 76

I created a new file in Windows Explorer but the file does not appear in the CodeWarrior Projects view. Why? 70
If I am using a Macintosh computer, what is the alternate to using the F3 key in Editor to open declarations? 19
If I have a debugger session running on my application, how can I halt the application? 142
I need to enable logging for flash programming. How do I do it? 135
Installation 17
Instruc 104
In the Memory view, is it possible to go back to the address that the rendering was created for? 148
Is it necessary to have the project name identical to the name of the directory that contains the .project file? 80
Is it possible to change the default code format settings? 25
Is it possible to configure build settings for a particular source code file? 130
Is it possible to debug the target multiple times with the same binary application, without flashing the device? 134
Is it possible to display line numbers in Editor? 24
Is it possible to display the expressions always in the Hexadecimal format instead of the default decimal format? 150
Is it possible to exclude a single or multiple files from the project while building it? 130
Is it possible to preview the changes of the Replace action? 51

Is it possible to restrict the number of resources that appear on my Workbench window? 37
Is it possible to retrieve a file that I deleted accidentally from my CodeWarrior project? 62
Is it possible to roll back the changes I did to my source code? 28
Is it possible to specify a different name for the binary file? 142
Is it possible to store output of the build configuration of my project in a separate directory? 131
Is it possible to view definition of a macro or a variable in the source code file? 21
Is it possible to view evaluated expansion of a macro in the CodeWarrior IDE? 27
Is it possible to view the change history of a source code file? 67
Is the Performance Analysis tool available in two different versions? 163
Is there a file that contains all the launch configuration settings? 74
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? 58
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? 156
Is there a way to add breakpoints to C and Assembly files that exist in a library? 122, 141
Is there a way to change the default color of comments that span across multiple lines? 29
Is there a way to change the variable or the function name in a CodeWarrior project? 53
Is there a way to display the variables always in the Hexadecimal format? 145
Is there a way to filter the settings in the Preferences window to find a particular setting? 64
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? 82
Is there a way to replace a variable easily? 56
Is there a way to revert to the original contents of a source code file? 30
Is there a way to view disassembly at any location? 149
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? 35

L
Linker 115
Local history 106
M
Miscellaneous 59, 135
modifications 99

N
New FAQs 14

O
object code 108

P
Perspective 98
Profiling 159
project 96, 97, 102
Project Management 95
project properties 99
projects 98
Property Settings 98

R
related 95
resize 98

S
Search and Replace Action 50
shortcut key 108
source 96
Source 102
source files 95, 97, 104–106

T
The warnings symbols in Editor distort visibility of breakpoints. How can I stop these warning symbols from appearing? 123

V
Variables 125
views 98

W
What all registers will be affected by the settings in the Profiler Configuration panel? 164
What are counterpoints? 163
What are different types of Register Analyzer? 166
What are Icon decorators? 90
What are Register Analyzers? 166

What are scenarios? 163
What are software and hardware tracepoints? 162
What are start and stop tracepoints? 162
What are tracepoints? 161
What controls the initial layout of a view in a Workbench window? 37
What does an asterisk on the Editor title bar mean? 32
What does it mean to collect events with counters? 177
What does the checkmark next to the breakpoint represent? 119
What does the number of VLES executed represents in the result? 166
What does the Reset on Connect option do? 151
What if I want to delete a breakpoint? 120
What is a hardware breakpoint? 125
What is a perspective? 62
What is a static local variable? 128
What is a Working Set? How do I build/compile a Working Set? 82
What is Linker Command File and how to enable trace in it? 171
What is Performance Analysis perspective used for? 163
What is the difference between a breakpoint and a line breakpoint? 123
What is the difference between counted events and count cycles using the OCE counters? 165
What is the difference between the hardware breakpoint and the software breakpoint? Why would I use one over the other? 122
What is the difference between the Toggle Breakpoint and Enable/Disable breakpoint options? 120
What is the first thing that I see when I start the CodeWarrior IDE? 62
What is the Manage Configurations button in the C/C++ perspective toolbar used for? Could it be replaced by the Properties button? 79
What is the purpose of the Limit New Breakpoint to Active Debug Context button in the Breakpoint window? 122
What is the purpose of the Outline view? 78
What is the purpose of the Properties view? 78
What is the purpose of the Tasks view? 77
What is the relationship between DPU and trace? 177
What is the speed of SmartDSP HEAT? Can the SmartDSP HEAT provide higher speed than HSST when doing trace offload? 167
What is the use of remote launch in CodeWarrior scripts? 166
What trace configuration method do I use automatic or manual? 179
When I compile my project, where can I see the commands that are being executed? 114
When I enable watch on a static variable, it shows the error, "<error(s)> during the evaluation". What do I do? 128
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? 92
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? 80
When tracing with VTB (DPU/OCE) is done all writes go directly out on the bus without passing the MMU but does not go directly to the memory. Is privilege signal required while writing on the bus? 165
When using default 8144ADS stationary project all DPU registers display hex number "FF". Also, when I try to change the value of register it does not allow to change. Why I can not change the value of DPU registers? 167
When you should use VTB write modes: Overwrite, One Address, or Trace Event Request? 174
Where does the CodeWarrior IDE save the debug configuration as a local file by default? 80
Where do I see the warning and error messages? 114
Where exactly the disassemble file gets created? 75
Which breakpoint option do I select if I want to add a breakpoint? 120
Which breakpoint option do I select if I want to skip a breakpoint while still keeping it in my breakpoint list? 120
Which option do I select if I want to add a tracepoint? 162
Which set of files I need to backup or store in a version control system to completely recover project settings and related debug settings? 58
While debugging in RAM my debugger halts and reports an "illegal breakpoint" error. What do I do? 124
While launching a project, the CodeWarrior software automatically builds the current active target. How can I specify the build configuration I want to debug? 108
Why does the DE.exe appear in the Windows Task Manager even after terminating and closing the debugger perspective? 145
Why does the Rename option appear grayed out? 70
Why do I get the following error message when I attempt to debug my project? 143
Why do I get the following error message when I try to edit a source code file? 31

Why duplicating a configuration in the debugger perspective does not duplicate the run configuration? 71
Why it is important to import offline trace data? 178
Why the.metadata folder in my workspace stores a huge history? 76
Why the console view does not display all the warning and error messages by default? 74
Why the project that I just created is not visible in the CodeWarrior Projects view? 70
Workbench Window 36
workspace 97, 98
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.

© 2010–2015 Freescale Semiconductor, Inc. All rights reserved.