Release Notes

CodeWarrior™ Development Studio for Microcontrollers V10.2

TABLE OF CONTENTS

1 What's New

Freescale's CodeWarrior for Microcontrollers v10.2 integrates the development tools for the ColdFire®, ColdFire+, DSC, Kinetis, Qorivva, PX, RS08, S08, and S12Z architectures into a single product based on the Eclipse open development platform. Eclipse offers an excellent framework for building software development environments and is becoming a standard framework used by many embedded software vendors.

Architectures supported in this version of the tools for the first time are:

- DSC
- S12Z

Many new derivatives for other architectures have been added, with details below.

Major new features of this release include:

- Support for subprojects within the project manager
- Multi-threaded builds for host systems with multiple processors, reducing build times
- Migrated the IDE to Eclipse 3.6.1, with many enhancements as a result
- Debug support for Kinetis low power and very low power modes
- Real-time trace/profiling support for ColdFire v2-v4, based on external trace collection with P&E TraceLink probe
- Real time (non-intrusive) trace/profiling support for Kinetis, based on external trace collection with P&E TraceLink and J-Trace

The new product features include the following:

1.1 IDE

- Migrated to Eclipse 3.6.1
- Added subproject support.
- A project can be modified via a script. The script can copy the project, add additional sources files to the project, add/change build tool options, and add/change launch configurations and debugger related configurations.
- Multi-threaded builds are now supported for host systems with multiple processors. The option is ON by default and applies to all projects in a workspace. A per-project option is also available and overrides the workspace option. The per-project option is OFF by default.
- Default settings for "Project Menu->Clean" have changed to the typical use case.
    - "Clean projects selected below" is selected by default
    - Only the current project is selected by default
- Whenever a user is prompted for a file or path and path variables are used, the dialog now displays the absolute address.
- Ability to import multiple projects is improved.
- "File Requestor" usability is improved to allow easier use of variables in paths.
- The ECD (Command Line Driver) interface can build projects from different locations regardless of whether the projects belong to a workspace or not**.**
- The Black Box Crash tool allows users to gather information, which can be provided to the CodeWarrior support team to debug CodeWarrior crashes. This information includes the following:
    - Debug Engine crash dumps
    - eclipse logs
    - protocol logs (ccs/ccssim)
    - console logs
- Pressing the keyboard "Enter" key in the New Project Wizard is equivalent to selecting "Next" instead of "Finish."

- The command-line make file generation no longer launches the CodeWarrior IDE during the generation process.
- To access product documentation, select “Help->Documentation Roadmap" instead of “Help->User Assistance Map.”
- To access CodeWarrior videos, select “Help->Videos.” The link plays the videos in a standalone browser when clicked.
- To access CodeWarrior videos from the “Welcome Screen”, select “Tutorials->CodeWarrior Video Tutorials.” The link plays the videos in a standalone browser when clicked.
- Added support for Ubuntu 10.04 host-operating system

1.2 Wizards

1.2.1 New Project Wizard, Change Connection Wizard and Executable (Elf) Importer Wizard

- ColdFire+ – added support for new derivative families
  - MCF51JF
  - MCF51JG
  - MCF51JU
  - MCF51QF
  - MCF51QH
  - MCF51QM
  - MCF51QU

- DSC – added support for new architecture and derivative families
  - MC56F83xx
  - MC56F800x
  - MC56F801x
  - MC56F802x
  - MC56F803x
  - MC56F824x
  - MC56F825x
  - MC56F844xx
  - MC56F845xx
  - MC56F847xx

- Kinetis – added support for new Kinetis part numbers and derivative families
  - K10 Family
    - K10D (50 MHz)
    - K10D (100 MHz)
    - K10F (120 MHz)
  - K20 Family
    - K20D (50 MHz)
    - K20D (100 MHz)
    - K20F (120 MHz)
  - K30 Family
    - K30D (100 MHz)
  - K40 Family
    - K40D (100 MHz)
  - K50 Family
    - K50D (100 MHz)
    - K51D (100 MHz)
    - K52D (100 MHz)
    - K53D (100 MHz)
  - K60 Family

- K60D (100 MHz)
- K60F (120 MHz)
- K60F (150 MHz)
- K61F (120 MHz)
- K61F (150 MHz)

▪ K70 Family
- K70F (120 MHz)
- K70F (150 MHz)

o Qorivva – added support for new derivative families
▪ MPC56xxF
▪ MPC56xxK
▪ MPC56xxL
▪ MPC56xxP
▪ MPC56xxS
▪ MPC5668E/G

o PX– added support for new derivative families
▪ PXD
▪ PXN
▪ PXR
▪ PXS

o S12Z – added support for new architecture and derivative families
▪ S12ZVM

1.2.3 Added New Project Wizard Library project support for RS08/S08

1.2.4 Added New Project Wizard floating point support for S12Z

1.2.5 Added wizard support for new connections

o Universal Multilink FX - Kinetis, RS08/S08, S12Z, DSC, V1 ColdFire/ColdFire+, V2-V4 ColdFire, Qorivva, PX, S12Z, DSC
o TraceLink (USB/Ethernet) – Kinetis, V2-V4 ColdFire
o Cyclone MAX (USB/Ethernet/Serial) – Kinetis, Qorivva, PX
o Cyclone PRO (USB/Ethernet/Serial) – S12Z
o Open Source JTAG – DSC
o Open Source BDM – S12Z
o Segger J-Trace / SWO (SWD based) – Kinetis
o CodeWarrior USB TAP BDM for – DSC

1.3 Build Tools

1.3.1 Kinetis

o Improved optimizations for speed and size. About 17% better speed in EEMBC Automotive kernels.
o Added alias_by_type control to optimizations
o Added strength reduction and redundant load-store optimization
o Added support for Hardware Single Precision Vector Floating Point
o Added support for half-precision (16-bit) floating point type '__fp16'
o Added support for #pragma pack() to align data objects
o Improved generated debug info

1.3.2 S08

o S08 compiler optimizes successive calls to the same __far function via the same __far function pointer.
o Improved support for computing stack consumption for modules specified in the ENTRIES block in the linker parameter file.

- New assembler option (-DefLabel) to improve support for data allocation directives.

1.3.3 ColdFire

- Added linker support for the new naming convention used for object files in MCU 10.x
- Improved debug information generation for optimized code
- Added assembler support for absolute short addressing
- Added inline assembler support for absolute short addressing
- Improved optimization behavior for better use of bitmask operations with memory access

1.3.4 Qorivva/PX

- Improved optimization for leaf functions
- Enhanced SPE and SPE2 Intrinsics support and documentation
- Added linker support to fill unused bytes in a section with a defined byte pattern
- Improved project import experience with respect to standard libraries
- Improved optimizations, enabling 16 bit load/store and fused MAC instruction generation
- Improved generated debug info

1.3.5 DSC

- Assembler support for 56800EX core
- ELF/DWARF2 output format, including some DWARF3 extensions
- Library support for single precision trigonometric functions

1.3.6 S12Z

- Added compiler support
  - C and inline assembly parsing
  - Object code generation (ELF)
  - Debug information generation (DWARF2)
  - 32-bit floating point support
  - High-level optimization support (with impact on –O1, –O2 and –O3 optimization level)
  - Memory model support (-model <mem_model>: 14-bit addressing (small) / 18-bit addressing (medium) / 24-bit addressing (large))
  - Improved memory accesses code size (by using best-fit offsets)
  - minimum source level compatibility with the other HC(S)12(X) processors (due to different compiler / front-end technologies)
- Added assembler support
- Added linker support
- Added decoder /disassembler support

1.4 Debugger

1.4.1 ColdFire+

- Added support for the following families of derivatives
  - MCF51JF
  - MCF51JG
  - MCF51JU
  - MCF51QF
  - MCF51QU
  - MCF51QH
  - MCF51QM

1.4.2 DSC

- Standard run-control (run, stop, step in, step out, go and setting breakpoints)
- Ability to display all available processor general purpose, special purpose and memory mapped registers with bit-level information
- Project Importer support for CW DSC v8.3 projects
- Support for the following families of derivatives

- MC56F83xx
- MC56F800x
- MC56F801x
- MC56F802x
- MC56F803x
- MC56F824x
- MC56F825x
- MC56F844xx
- MC56F845xx
- MC56F847xx

1.4.3 Kinetis

- Added support for the following derivatives
  - MK10DX32
  - MK10DX64
  - MK10DX128
  - MK10DN32
  - MK10DN64
  - MK10DN128
  - MK10FX512
  - MK10FN1M0
  - MK20DX32
  - MK20DX64
  - MK20DX128
  - MK20DN32
  - MK20DN64
  - MK20DN128
  - MK20FX512
  - MK20FN1M0
  - MK60FX512
  - MK60FN1M0
  - MK61FX512
  - MK61FN1M0
  - MK60FX512
  - MK60FN1M0
  - MK61FX512
  - MK61FN1M0
  - MK70FX512
  - MK70FN1M0
  - MK70FX512
  - MK70FN1M0
- Added cache viewer
- Added support for hardware floating point unit
- Added support for NAND Flash Controller
- Added low power support
- Added custom trim support

1.4.4 Qorivva

- Added support for the following families of derivatives
  - MPC56xxF
  - MPC56xxK
  - MPC56xxL
  - MPC56xxP
  - MPC56xxS
  - MPC5668E/G

1.4.5 PX

- Added support for the following families of derivatives

- PXD
- PXN
- PXR
- PXS

1.4.6 S12Z

- Standard run-control (run, stop, step in, step out, go and setting breakpoints)
- Ability to display all available processor general purpose, special purpose and memory mapped registers with bit-level information
- Support for the following families of derivatives
  - MC9S12ZVM

1.4.7 General

- Update to Eclipse 3.6:
  - Breadcrumb Debug view – Minimize debug window to a toolbar state
  - Columns in Expression view – [Expression view now displays expressions in a similar column way as variables view]
  - C/C++ refactoring support
  - C/C++ Editor enhancement for C++ types including templates
  - Memory Browser to quickly inspect any processor memory from different memory spaces
  - Per-launch build before debug setting with higher degree of control
- Default Debug Console presents the output of scripts executed during the launch phase of a debug session
- Debugger offers to close current debug session if new one is started
- Target Task Framework – multiple tasks execution
  - Tasks can be canceled individually but still run sequentially.
  - In case of successive hits of the run button, when tasks from previous session are still running, the new tasks will be aligned in the queue and will wait after previous session to finish.
- Redesign Import/Export/Fill Memory Target Task
- Breakpoints can now be set before the PIC load address is changed.
- The debugger presents the values of the unreadable memory in a distinct way, no longer using the *reservedchar* from memory configuration file.
- Kernel Awareness SDK: Collect relocation information for multiple ELF debug
- Able to debug an executable (ELF) without creating a dedicated project in a workspace with ELF Import Wizard. Now users can specify an executable to be debugged from any place on the host machine
- Cast to Type – present possible choices
- Cast to Type – casting of array elements
- Find Register - does not require register group to be expanded
- Display Linux task name in debug view
- Black Box Crash Recorder - Diagnostic Information wizard exports debugger engine crash details.
- Breakpoint Annotations – User can customize how an editor will draw breakpoint marker, e.g. highlight the whole line
- Progress bar feedback is improved. Progress bar is more accurate to the action currently being executed as part of the download step
- CodeWarrior/Eclipse startup time is improved. CodeWarrior startup is twice as fast on usual tasks. User interface is also not locked while background initialization work is being done.
- Offline Registers View – Registers dump file editor displays bitfield register details in the same way as the register view.
- Show registers in running mode in Register Window – All register groups will be displayed while target is running so user can monitor the register of interest.
- Debugger shell: scripting support for Save/Restore Registers
- DWARF 3.0 Enhancement - support for Non-Contiguous Address Ranges
- DWARF2 Symbolics support for DSC Memory Architecture

- Project Importer support for CW MCU v10.0 and CW MCU v10.1 projects

1.5 Flash Programming

1.5.1 Flash Programmer

- Added "Verify after program" option
- Added a new button in the actions area - "Duplicate Action", which will create a clone of the selected action.
- Added multicore support
- Added diagnostics to report bad blocks for NAND flashes
- Improved Verify Operation Performance

1.5.2 General

- Added “Simple Flash” to allow a program to be flashed to a device without a project

1.6 New Target Connections

1.6.1 V1 ColdFire/ColdFire+

- Universal Multilink FX

1.6.2 V2-V4 ColdFire

- TraceLink (USB/Ethernet)
- Universal Multilink FX

1.6.3 DSC

- Universal Multilink/Multilink FX
- Open Source JTAG
- CodeWarrior USB TAP

1.6.4 Kinetis

- TraceLink (USB/Ethernet)
- Cyclone MAX (USB/Ethernet/Serial)
- Universal Multilink FX
- Segger J-Trace / SWO (SWD based)

1.6.5 Qorivva

- Cyclone MAX (USB/Ethernet/Serial)
- Universal Multilink FX

1.6.6 PX

- Cyclone MAX (USB/Ethernet/Serial)
- Universal Multilink FX

1.6.7 RS08/S08

- Universal Multilink FX

1.6.8 S12Z

- Universal Multilink/Universal Multilink FX
- Cyclone PRO (USB/Ethernet/Serial)
- Open Source BDM

1.6.9 General

- Unreferenced Remote System configurations are automatically removed from the Remote Systems view.
- Remote System configuration reuse is improved by separating connection configuration from device specification.
- Eclipse will not lock user interface when changing RSE systems in workspaces with hundreds of RSE systems loaded.
- Faster download to RAM

1.7 Software Analysis

- Real-time trace/profiling support for ColdFire v2-v4, based on external trace collection with P&E TraceLink probe
- DSC data visualization
- DSC trace/profiling
- Kinetis SWO trace/profiling with Segger J-Link and J-Trace probes
- Kinetis 50 MHz external trace collection (ITM) with J-Trace
- Real time (non-intrusive) trace/profiling support for Kinetis, based on external trace collection with P&E TraceLink and J-Trace
- Qorivva/PX trace/profiling support based on PCFIFO
- S12z trace/profiling support
- Software Analysis view automatically opens when trace is enabled
- Start/stop/reset trace anytime directly from debug session toolbar
- Wizard to automatically add software trace points support
- Automatically disable profiler for probes that do not collect trace
- Support to configure the time unit in the Trace Viewer
- Support for long run trace collections (big traces)
- Import trace wizard for raw trace files
- Remote Launch support to automate trace/profiling through scripting
- Export to csv functionality available from all views
- Log view available for all derivatives to log all target activities
- Timeline support to see percentage of application in low power mode for Kinetis
- Support for all newly supported derivatives
- Consolidated same user interface look and feel for all supported derivatives
- Improved trace collection and processing performance

1.8 Processor Expert

- Added support for following Kinetis family of derivatives
  - MK10DX32
  - MK10DX64
  - MK10DX128
  - MK10DN32
  - MK10DN64
  - MK10DN128
  - MK10FX512
  - MK10FN1M0
  - MK20DX32
  - MK20DX64
  - MK20DX128
  - MK20DN32
  - MK20DN64
  - MK20DN128
  - MK20FX512
  - MK20FN1M0
  - MK60FX512
  - MK60FN1M0
  - MK61FX512
  - MK61FN1M0
  - MK60FX512
  - MK60FN1M0
  - MK61FX512
  - MK61FN1M0

- MK70FX512
- MK70FN1M0
- MK70FX512
- MK70FN1M0

- Added support for the following DSC families of derivatives
  - DSP5685x
  - MC56F800x
  - MC56F801x
  - MC56F802x
  - MC56F81xx
  - MC56F82xx
  - MC56F83xx

- Added Logical Device Drivers
  - DRY_LDD – DryIce (tamper detect and secure storage)
  - LCDC_LDD – LCD Controller
  - NFC_LDD – Flash Memory

1.9 Documentation

CodeWarrior Development Studio for Microcontrollers V10.2 contains the following new or updated documentation.

1.9.1 Getting Started

- Microcontrollers V10.x Getting Started Guide **(updated)**
- Microcontrollers V10.x Quick Start **(updated)**
- CodeWarrior Project Importer Quick Start **(updated)**
- Eclipse Quick Reference Card **(updated)**
- HCS08 Profiling and Analysis for Microcontrollers V10.x Quick Start **(updated)**
- ColdFire V1 Profiling and Analysis for Microcontrollers V10.x Quick Start **(updated)**
- Kinetis Profiling and Analysis for Microcontrollers V10.x Quick Start **(updated)**
- Ethernet TAP Quick Start **(updated)**

1.9.2 Users Guides

- CodeWarrior Common Features Guide (formerly known as Freescale Eclipse Extensions Guide) **(updated)**
- Microcontrollers V10.x Targeting Manual **(updated)**
- HCS08 Build Tools Reference Manual for Microcontrollers V10.x **(updated)**
- RS08 Build Tools Reference Manual for Microcontrollers V10.x **(updated)**
- ColdFire Build Tools Reference Manual for Microcontrollers V10.x **(updated)**
- Kinetis Build Tools Reference Manual for Microcontrollers V10.x **(updated)**
- 56800/E (DSC) Build Tools Reference Manual for Microcontrollers V10.x **(new)**
- Power Architecture Processors Build Tools Reference Manual for Microcontrollers V10.x **(updated)**
- MISRA-C:2004 Compliance Exceptions for the HCS08, RS08, ColdFire, Kinetis and Power Architecture Processors Libraries for Microcontrollers V10.x **(updated)**
- EWL C Reference Manual **(updated)**
- EWL C++ Reference Manual **(updated)**
- HCS08/RS08 Assembler Reference Manual for Microcontrollers V10.x **(updated)**
- ColdFire Assembler Reference Manual for Microcontrollers V10.x **(updated)**
- Kinetis Assembler Reference Manual for Microcontrollers V10.x **(updated)**
- 56800/E (DSC) Assembler Reference Manual for Microcontrollers V10.x **(new)**
- HC(S)08/RS08 Build Tools Utilities Manual for Microcontrollers V10.x **(updated)**
- Profiling and Analysis User Guide for Microcontrollers V10.x **(updated)**

- USB TAP Probe Users Guide **(updated)**
- Ethernet TAP Probe Users Guide **(updated)**
- Open Source BDM-JM60 Users Guide **(updated)**
- Processor Expert User Manual **(updated)**
- Device initialization User Manual **(updated)**
- Component Development Environment (CDE) Guide (**new)**
- RTOS Adapter Developer's Guide **(updated)**
- Signal Processing Engine Auxiliary Processing Unit Programming Interface Manual for Power Architectures Processors **(updated)**

1.9.3 Application Notes

- AN3859 – Adding Device(s) to the CodeWarrior Flash Programmer for Microcontrollers V10.x
- AN3967 – How to Write Flash Programming Applets
- AN4095 – CodeWarrior Build Tools Options for Optimal Performance on the Power Architecture e200 Core
- AN4104 – Converting ColdFire Projects to CodeWarrior Development Studio for Microcontrollers 10.x
- AN4188 – RS08 Upper Memory Access
- AN4329 – Relocating Code and Data Using LCF for ColdFire Architecture **(new)**
- AN4331 – Enabling OSBDM DLLs **(new)**
- AN4316 – Configuring Compiler Options for Optimal Performance of ColdFire Devices **(new)**
- AN4414 – CodeWarrior Build Tools Options for Optimal Performance on HCS08 Cores **(new)**
- AN4415 – CodeWarrior Build Tools Options for Optimal Performance on RS08 Cores **(new)**
- AN4416 – CodeWarrior Build Tools Options for Optimal Performance on Kinetis Cores **(new)**

1.9.4 Supporting Information

- Microcontrollers V10.x FAQ Guide **(updated)**

1.9.5 Cheat Sheets **(updated)**

- CodeWarrior for Microcontrollers Features **(updated)**
  - Building Library (HCS08)
  - Creating, Building and Debugging a Project
  - Creating New Project from Example Project
  - Changing P&E Connections Setting
  - Configuring Perspectives
  - Debugging Projects in ROM
  - Debugging Project Using Command Line
  - Examples: Porting Classic IDE Projects to Eclipse
  - Importing and Debugging Externally Built Executable File
  - Using Memory View
  - Using Microcontrollers Change Wizard
  - Using Registers View
  - Working with Build Configurations

- CodeWarrior Core Features **(updated)**
  - Create a Linux AppTRK Remote System
  - Making C/C++ the IDE's Default Perspective
  - Target Management via RSE

- Using the Flash Programmer
- Using the Import Wizard

- CodeWarrior Processor Expert Features **(new)**
  - Processor Expert Basics for CodeWarrior for MCUs
  - Processor Expert Component Settings Tips
  - Processor Expert Examples for Kinetis and LDD
  - Processor Expert Device Initialization Basics

- CodeWarrior Profiling and Analysis Features **(updated)**
  - Creating, Debugging, Collecting, and Viewing Data (on ColdFire V1 Target)
  - Creating, Debugging, Collecting, and Viewing Data (on HCS08 Target)
  - Creating, Debugging, Collecting, and Viewing Data (on Kinetis Target)
  - Collecting Trace on Different Modes (on Kinetis Target)
  - Collecting Trace on Different Modes (on ColdFire V1 Target)
  - Collecting Trace on Different Modes (on HCS08 Target)
  - Collecting Trace on Kinetis Target

1.9.6 Videos (available at www.freescale.com/cwmcu10)

- CodeWarrior Overview
- How to Debug a Project
- Importing a Project
- Critical Code
- Creating a Project from an Example Project
- Changing Build Configurations
- Adding Registers View
- Adding Memory Monitors How to Debug a Target Without an Application

1.10 Example Projects (available in Help -> Welcome-> Example Projects)

1.10.1 Kinetis – TWR-K40X256-KIT

- AComp (uses Processor Expert)
- adc_demo
- can_loopback_node
- crc_demo
- DAC_ADC (uses Processor Expert)
- dac12bit_demo
- flexbus
- flexmem
- gpio
- hello_world
- hscmp
- i2c
- LCD (uses Processor Expert)
- lptmr
- mcg_demo
- pdb_adc_demo
- pmc
- rtc
- RTC (uses Processor Expert)
- sa_itm
- sa_software_tracepoints
- sci2can
- Serial (uses Processor Expert)

- Slcd
- SPI (uses Processor Expert)
- tsi
- USB_DCD
- USB_device
- USB_dual_role
- USB_HOST
- USB_MAX3353
- USB_MS_Device (uses Processor Expert)

1.10.2 Kinetis – TWR-K60N512-KIT

- adc_demo
- can_loopback_node
- CAN (uses Processor Expert)
- crc_demo
- dac12bit_demo
- DMA (uses Processor Expert)
- Ethernet (uses Processor Expert)
- flexbus
- flexmem
- freertos_lwip
- freertos_uip
- gpio
- hello_world
- hscmp
- i2c
- I2C (uses Processor Expert)
- Low_Power_Debug_Example
- lptmr
- mcg_demo
- pdb_adc_demo
- pmc
- PWM (uses Processor Expert)
- rtc
- sci2can
- SDHC (uses Processor Expert)
- slcd
- tsi
- TSI (uses Processor Expert)
- USB_DCD
- USB_device
- USB_dual_role
- USB_HOST
- USB_MAX3353
- USB_MS_Device

1.10.3 Kinetis – TWR-K70F120M-KIT

- cache_example
- CRC (uses Processor Expert)
- Ethernet (uses Processor Expert)
- low_power_example
- RNG (uses Processor Expert)
- RTC (uses Processor Expert)
- SDHC (uses Processor Expert)
- USB_Mass_Storage (uses Processor Expert)

1.10.4 Qorivva
- INTC-HW-VLE-MPC5604P
- INTC-SW-VLE-LSM-MPC5643L
- INTC-SW-VLE-LSM-MPC5675K
- INTC-SW-VLE- MPC5604P

1.11 New or updated license limits

1.11.1 DSC **(new)**
- Special Edition: up to 64K of ANSI C code
- Basic Edition: up to 128K of ANSI C code
- Standard Edition: unlimited C code
- Professional Edition: unlimited C code, kernel awareness support, Component Development Environment

1.11.2 PX **(new)**
- Special Edition: up to 512K of ANSI C code
- Basic Edition: up to 1MB of ANSI C code
- Standard Edition: unlimited C code
- Professional Edition: unlimited C/C++ code, kernel awareness support

1.11.3 RS08/S08 **(updated)**
- Special Edition: **up to 64K of ANSI C code,** Basic and **Advanced Processor Expert Components**
- Basic Edition: up to 128K of ANSI C code, Basic and **Advanced Processor Expert Components**
- Standard Edition: unlimited C code, Basic and **Advanced Processor Expert Components**
- Professional Edition: unlimited C (RS08, S08), unlimited C++ (S08) code, Basic and Advanced Processor Expert Components, kernel awareness support

1.11.4 V1 ColdFire/ColdFire+ **(updated)**
- Special Edition: up to 64K of ANSI C code, Basic and **Advanced Processor Expert Components**
- Basic Edition: up to 128K of ANSI C code, Basic and **Advanced Processor Expert Components**
- Standard Edition: unlimited C code, Basic and **Advanced Processor Expert Components**
- Professional Edition: unlimited C/C++ code, Basic and Advanced Processor Expert Components, kernel awareness support

1.11.5 V2-V4 ColdFire **(updated)**
- Special Edition: up to 128K of ANSI C code, Basic and **Advanced Processor Expert Components**
- Basic Edition: up to 512K of ANSI C code, Basic and **Advanced Processor Expert Components**
- Standard Edition: unlimited C code, Basic and **Advanced Processor Expert Components**
- Professional Edition: unlimited C/C++ code, Basic and Advanced Processor Expert Components, kernel awareness support

1.11.4 S12Z **(new)**

- Special Edition: up to 64K of ANSI C code
- Basic Edition: up to 128K of ANSI C code
- Standard Edition: unlimited C code
- Professional Edition: unlimited C/C++ code, kernel awareness support, Component Development Environment

2 System Requirements

2.1 Recommended Configuration

- 2.6 GHz Pentium® compatible processor or better
- Microsoft Windows XP/Vista
- 2 GB RAM
- 2 GB hard disk space, 400 MB on Windows system disk
- CD-ROM drive for installation
- USB port for communications with target hardware
- Ethernet port for communications with target hardware (optional)

2.2 Operational Minimum Configuration

- 1.8 GHz Pentium® compatible processor or better
- Microsoft Windows XP
- 2 GB RAM
- 3GB (When the installer is run directly from a DVD)
- 5GB (When the software installer is downloaded)
- 400MB on Windows system disk
- CD-ROM drive for installation
- USB port for communications with target hardware

2.3 Host Operating System Support

- Microsoft® Windows XP 32-bit and 64-bit (Professional Edition)
- Microsoft Windows Vista® 32-bit and 64-bit (Home Premium Edition and Business Edition)
- Microsoft Windows 7 32-bit and 64-bit (Home Premium Edition and Professional Edition)
- Red Hat Enterprise Edition 5.4 (32-bit and 64-bit)
- Ubuntu 10.04 (32-bit and 64-bit)

3 Product WEB page

CodeWarrior Development Studio for Microcontrollers v10.2 is available for download at http://www.freescale.com/cwmcu10.

4 Installation and Licensing

To install CodeWarrior Development Studio for Microcontrollers v10.2, double-click the installation package and a wizard will guide you through the installation process. An Evaluation license is automatically installed with your product and you do not need to register it. This license allows you to develop projects as Professional Edition during the evaluation period. After 30 days, the license works as a Special Edition license (free permanent, but feature limited) which supports unlimited assembly code, up to 64KB of C code for S08/RS08 and V1 ColdFire/ColdFire+ derivatives, up to 128KB of C code for V2-V4 ColdFire and Kinetis derivatives and up to 512KB of C code for Qorivva and PX derivatives.

5 Technical Support

All CodeWarrior issues are tracked through Freescale's normal Service Request Process. To report feature requests (enhancements) or defects for CodeWarrior Development Studio for Microcontrollers v10.2, please submit a Service Request.

1. Go to http://www.freescale.com/support
2. Log in.
3. On the resulting MyFreescale page, click Enter a Service Request
4. Choose category Software Product Support
5. Choose topic CodeWarrior
6. Click Next.
7. Provide the required information. An attachment up to 10 MB may be attached to the SR. You may also specify email addresses of people you would like to keep notified on the progress of the SR. Separate multiple email addresses with commas.

    - **Type:** pick from Question, Defect Report, Feature Request
    - **Subject:** be short and descriptive
    - **Description:** details your question, defect or feature request
    - **Severity:** choose from Normal, High, or Highest
    - **Reproducibility:** enter reproducibility information
    - **Target:** specify the hardware microcontroller/microprocessor family involved
    - **Product:** CW for Microcontrollers
    - **Root Cause/Nature:** enter root cause (e.g. software defect)
    - **RTOS:** enter the RTOS being used (e.g. NA)
    - **Major:** 10
    - **Minor:** 2
    - **Patch:** leave blank
    - **Component:** enter component (e.g. Debugger)
    - **Host:** enter host operating system

    Please note: The Product field must be set to CW for Microcontrollers. This will allow Freescale to find SRs related to this project very easily, report on them, and gather statistics on how the product is doing.

8. When finished, click Submit.

After Submit is selected, a confirmation page will be displayed with the SR number. You will also receive a confirming email sent to the address specified in your Freescale account.

Appendix A: Known issues and Workarounds

Issue ID	Description
IDE	
MTWX46503	**Description:** The Eclipse IDE's Syntax parser reports a syntax error for lines using the construct "@(address)" **Workaround:** 1) Ignore Syntax errors for constructs with this format. The compiler will confirm if there is an actual error in your code. 2) To disable error markers generated by the Indexer, go to Window ->Preferences. Select C/C++ ->Editor ->Hovers. Turn off 'Enable editor problem annotation'. Note that this will not prevent any build error or warnings in the editor, but will prevent any syntax error from generating a marker in the editor.
MTWX51041	**Description:** The Eclipse IDE's Syntax parser reports a syntax error for lines using the declarations: • extern asm __declspec(register_abi) void _ExitProcess(void); • asm __declspec(register_abi) _ExitProcess(void); • asm void asm_exception_handler(void); **Workaround:** 1) Ignore Syntax errors for declarations with this format. The compiler will confirm if there is an actual error in your code. 2) To disable error markers generated by the Indexer, go to Window ->Preferences. Select C/C++ ->Editor ->Hovers. Turn off 'Enable editor problem annotation'. Note that this will not prevent any build error or warnings in the editor, but will prevent any syntax error from generating a marker in the editor.
MTWX44966	**Description:** The Eclipse IDE's Syntax parser reports a syntax error for lines using the construct 'interrupt <number>' **Workaround:** 1) Ignore Syntax errors for constructs with this format. The compiler will confirm if there is an actual error in your code. 2) To disable error markers generated by the Indexer, go to Window ->Preferences. Select C/C++ ->Editor ->Hovers. Turn off 'Enable editor problem annotation'. Note that this will not prevent any build error or warnings in the editor, but will prevent any syntax error from generating a marker in the editor.
MTWX51305	**Description:** Build error reported when an imported Classic Project is renamed. This creates a path that is incorrect. **Workaround:** 1) Rename the Classic Project before importing it into your workspace. Or 2) If you rename an imported Classic Project • Open the project Properties • In the Resource-> Linked Resources panel select the Path Variables tab • Update the "CW_Importer_ProjectDir1" variable
MTWX51669	**Description:** When the Change Connection Wizard (CCW) is used, the compiler input path in the new project is set as an absolute path. This could cause problems if you move the project to a different location. **Workaround:** Manually edit the absolute paths if the project is moved to a new location. Open the settings of the new CCW project and click 'Apply' to avoid the

	problem of disappearing user-added include paths.
MTWX51955	**Description:** After creating a new path variable that depends on another relative path variable, operations that use that new path variable will fail. **Workaround:** When creating the relative link location, use a custom variable that isn't relative to another variable or use one of the CDT variables like WORKSPACE_LOC.
MTWX52004	**Description:** On some Linux hosts, the welcome screen does not display the Freescale specific banner and two customized icons are replaced with a default Eclipse icon. All functionality in the welcome screen works correctly. **Workaround:** None. Everything works and is labeled as expected.
MTWX52028	**Description:** Import of a legacy project (.mcp) from Classic product does not work in Linux when the classic project is inside the Eclipse workspace. **Workaround:** Place the legacy project (.mcp) outside of the Eclipse workspace before importing it.
Build Tools	
MTWX49686	**Description:** Prefetching to the FlexBus aliased region (0x1800_0000 - 0x1BFF_FFFF) causes Kinetis core to hang since FlexBus is ON by default. **Workaround:** Avoid the code pattern exposing the core issue. Example : Avoid BXEQ LR (Z bit 0) where LR has FlexBus address (0x1800_0000 - 0x1BFF_FFFF)
MTWX50592	**Description:** Suboptimal performance with 'softfp' Floating Point option. Kinetis FP math library functions need to be optimized. **Workaround:** None
MTWX51702	**Description:** Dead variables may be displayed in the 'Variables' windows with invalid value and location. **Workaround:** Right click and add the variable of interest in the current execution C statement to 'Expressions' window through 'Add watch Expression…"
MTWX51953	**Description:** Linker file mismatch for certain DSC devices **Workaround:** Edit two lines in lcf so that p-mem origin at 0x4000 is correctly set: .p_interrupts_ROM (RX) : ORIGIN = 0x004000, LENGTH = 0x0080 # reserved for interrupts .p_flash_ROM (RX) : ORIGIN = 0x004080, LENGTH = 0x0003F7F
Debugger	
MTWX46004	**Description:** Flashing Kinetis K40/60 boards with OSJTAG is slow. **Workaround:** Use P&E Universal Multilink or Segger J-Link for faster flash/debug performance.
MTWX50421	**Description:** Debugging DSC using OSJTAG is very slow – 20-25 seconds per step **Workaround:** Use P&E Universal Multilink or USB TAP for faster debug performance.
MTWX51667	**Description**: Memory browser view does not display values. This is actually a defect in the Eclipse environment. (https://bugs.eclipse.org/bugs/show_bug.cgi?id=161859) **Workaround**: Use the Memory view that is opened by default.
MTWX51856	**Description:** Browsing to the last register in the ADC group using the Register Details View, on the DSC platform, causes an internal error. **Workaround:** None
MTWX51877	**Description**: Verifying Memory Write operations results in an error on MC56F84xxx devices. **Workaround**: Don't verify the write operation. The write operation is in fact successful, it is the verify operation that causes the error.
Software Analysis	
MTWX51245	**Description:** CodeWarrior generates an error when moved after install or installed without installer **Workaround:** Close CodeWarrior. Manually remove all compreg.dat and xpti.dat files

	from $CWInstallDir\eclipse\plugins\com.freescale.morpho.core_7.0.0.106089-201110132018\xpcom\bin\components\
MTWX51800	**Description:** The conditional watchpoint for DSC 56F84xxx is hit without checking condition **Workaround:** None
Component Development Environment	
MTWX51994	**Description:** Cannot open "Component Information" in editor view when import embedded component project **Workaround:** None. Import/Export components using CodeWarrior editor view is not supported. This feature will be available in the next release.
MTWX51965	**Description:** Unable to the rename and copy component projects in CodeWarrior **Workaround:** None. This feature will be available in the next release.

Appendix B: CodeWarrior Eclipse usage on a Windows Vista or Windows 7 system

1. The CodeWarrior installer should be run using the 'Run as administrator' option. CodeWarrior service packs are installed with the Eclipse Updater. The updater should also be run with this option. To start the Eclipse Updater select 'Window > Install new software' in the menu.

2. Eclipse needs read/write access to the installation folder. Make sure the eclipse installation folder has the appropriate permissions for all users.

3. Make sure your project workspace has read and write permissions.

Appendix C: CodeWarrior Eclipse usage on a Linux system

1. The CodeWarrior installer should be run from a root account. CodeWarrior service packs are installed with the Eclipse Updater. The updater should also be run from a root account. To start the Eclipse Updater select 'Window > Install new software' in the menu.

2. Eclipse needs read/write access to the installation folder. Make sure the eclipse installation folder has the appropriate permissions for all users.

3. Make sure your project workspace has read and write permissions.