

CodeWarrior Development Studio for Advanced Packet Processing v10.4.0

NXP Semiconductors

1. Revision History

The following modifications to this release note have been made relative to the note provided with the CodeWarrior for Advanced Packet Processing v10.4.0 installation:

Date	Description
Jan 31, 2019	V10.4.0

An updated version of this document is available on [CodeWarrior Documentation Page](#).

Contents

- 1. Revision History..... 2
- 2. About this release..... 3
- 3. System Requirements 5
- 4. Release Contents..... 5
- 5. Fixed Issues 13
- 6. Known Issues..... 14



2. About this release

2.1 Version information

This release note provides important information for users of CodeWarrior Development Studio for APP. You are encouraged to read this document to become familiar with this release's supported targets, new features, errata with workarounds, and other useful information.

CodeWarrior for Advanced Packet Processing includes compile, debug, trace and profile tools for the AIOP packet processor in selected NXP devices.

CW-APP v10.4.0 contains:

- Installer: CW-APP is provided now as a standalone installer. It can be installed as is, in new location or over a previous CW-ARMv8 installation.
- Simulator continued to be no longer delivered as part of the main CW4NET installer. External users should contact support to get access to the component.

Functional content in 10.4.0 is the same as for 10.3.x series.

2.2 Licensing

This release is distributed under an End-User License Agreement. The EULA is displayed during product installation and a copy is in the layout. Please read the contents of the document carefully before using this product.

When used for evaluation purposes CodeWarrior will generate a temporary license valid for 15 days from installation date. Please note that the generated license certificate is node-locked to the machine running the installer; evaluation on multiple machines requires separate installs.

Past expiry, external users can solicit an evaluation extension by opening a licensing SR – please see section 2.4. The license certificate (license.dat) needs to be placed in the CW4NET_<version>/Common/ folder.

2.3 Getting Help

Product documentation is included with the layout, available for off-line from the Documentation folder in the link above, or from [NXP Infocenter](#).

If you have questions, issues, or want to provide feedback, please use the NXP online support web page. To use this page, follow these steps:

1. In a web browser, go to <https://www.nxp.com/support/support:SUPPORTHOME> .
NXP **Technical Support** web page appears.
2. For general purpose questions/support please follow the Technical Communities link. Note this is a public, moderated forum.
3. For specific questions please follow the Support Requests link and instructions listed in the Web wizard

3. System Requirements

CodeWarrior Host

- *Windows® OS*: Intel® Pentium® 4 processor, 2 GHz or faster, Intel® Xeon™, Intel® Core™, AMD Athlon™ 64, AMD Opteron™, or later
- *Linux® OS*: 1.8 GHz Intel® Pentium® class processor (or better).
- At least 2 GB of RAM
- At least 3 GB of free disk space.
- Internet connectivity for web downloads and update access.

Operating System

Windows			
Host OS	SP Level	32-bit	64-bit
Windows 10	1803		X
Windows 7	SP1	X	X
Linux			
Host OS	Version	64-bit	
Ubuntu	18.04, 16.04	X	
Fedora	21	X	
Mint	17.x, 18.x	X	
RedHat Enterprise Linux / CentOS	6.x, 7.0	X	

NOTE CodeWarrior installation on 64-bit Linux hosts requires presence of dependent 32-bit libraries in the system. For supported distributions, the installer detects missing dependencies and provides options to install them automatically.

It is recommended (although not required) that all product maintenance operations be done with administrative privileges.

CodeWarrior installer may update system drivers for probe connectivity. It is highly recommended that any CW4NET versions or products should be closed before installing or updating an existing CW.

4. Release Contents

4.1 General

This release is based on Eclipse 3.7.1 version.

CodeWarrior tools assume the target board is configured for ARM debugger compliant mode, as opposed to Boundary Scan mode.

The following switches need to be applied:

- LS2088/85A-RDB: SW7.7 set to ON (TBSCAN_EN_B=1)
- LS2088/85A-QDS: SW5.7 set to ON (TBSCAN_EN_B=1)
- LS1088A-QDS: SW5.8 set to ON (TBSCAN_EN_B=1)
- LS1088A-RDB: SW4.8 set to ON (TBSCAN_EN_B=1)

Failure to apply the correct switch configuration results in inability to connect to the target. Connectivity in Boundary Scan mode is no longer supported.

4.2 Device Support Matrix

The following hardware configuration targets are supported:

Device	Board	GPP cores	AIOP cores	Comments
LA1575	LA1575-RDB	4 (A53)	4	Silicon rev. 1
LS2088A	LS2088A-RDB	8 (A72)	16	Silicon revs. 1 and 1.1.
LS1088A	LS1088A-RDB	8 (A53)	4	Silicon rev.1 @1.2 GHz

4.3 Installer and Product Layout

CodeWarrior for APP is provided as standalone installer starting with this version. It can be installed on top of an existing CW-ARMv8 installation or as a new clean directory.

4.4 Build Tools

The product incorporates an embedded, optimizing build system targeted for AIOP packet processor. The build tools include a C99 compliant compiler, linker, and assembler.

The build tools are fully integrated with CodeWarrior Eclipse IDE and are also available as command-line tools.

AIOP specific enhancement in the build tools are summarized as follows:

- Complete support for AIOP-specific instructions via dedicated intrinsics (built-in functions).
- Supports entry point qualifiers (an AIOP “task” starts at an entry point)
- Supports “task” specific variables (similar to thread local variables)
- Static stack usage analysis report (to optimize available workspace)
- Supports gcov format code coverage data generation
- Standard C99 and runtime library support

This release includes build 306 of Freescale AIOP build tools. There are important changes in debug information relative to AIOP build tools included in previous release (build 302) and also other bug fixes.

4.5 Documentation

Product documentation is available in several formats:

- Integrated with the help system of the Eclipse IDE
- In the product layout in PDF format (CW_APP/LS/Help/PDF/)
- Off-line download from [Compass Extranet](#)
- [Freescale Infocenter](#)

Getting Started Guide

Use this document to quickly get started with setting up boards, creating & debugging projects, collecting the trace data, or installing a service pack.

CodeWarrior for APP Getting Started Guide

Explains how to install the CodeWarrior software and then create, build, and debug a simple project for an LS2 hardware target.

Quick Start for CW for APP

Explains how to install CodeWarrior Development Studio for APP and how to create and debug a simple project.

Application Notes

AN5165 - AIOP SDK Applications Debug

Describes how to import and build AIOP SL and AIOP reflector projects with CW, and how to debug them when loaded within the SDK.

AN5044 - AIOP Task Aware Debug

Describes AIOP specific features in CodeWarrior debugger and task level debug scenarios: task specific breakpoints, task stepping, and viewing task specific information.

AN4950: Simultaneous Debug using CW for ARMv8/APP

This application note depicts the steps required to debug both the GPP and LDPAA cores simultaneously

using CW for ARMv8 and CW for Advanced Packet Processing.

Application notes are not included in the product build but they are available for download from CodeWarrior for Advanced Packet Processing [product support page](#).

User Guides

Use these documents to learn how to perform what to perform, solve the most common problems using the FAQs guide, or simply learn more about the built-in features.

CodeWarrior APP Targeting Manual

Explains how to use the CodeWarrior development tools to develop software that targets the Power Architecture cores; includes examples, definition of target-settings panel options, and a debugging tutorial.

CodeWarrior Common Features Guide

Describes the CodeWarrior IDE and debugger features that are common across all the CodeWarrior products.

CodeWarrior APP Build Tools Reference

Explains how to use CodeWarrior tools to build programs.

EWL C Reference

Documents the CodeWarrior implementation of the C standard library.

EWL C++ Reference

Documents the CodeWarrior implementation of the C++ standard library.

CodeWarrior TAP Users Guide

Explains how to set up the CodeWarrior TAP debug probe so it can communicate with the debugger and the target hardware.

CodeWarrior for APP - Frequently Asked Questions

Provides answers to the most frequently asked questions about the CodeWarrior IDE and the APP tools.

Cheat Sheets

Use these cheat sheets to complete moderate-to-complex tasks by following instructions that you execute real-time within the CodeWarrior IDE.

Select **Help > Cheat Sheets** in CodeWarrior IDE to see and run one of the available cheat sheets.

4.6 Debugger

The product includes an AIOP bare metal debugger, integrated in CodeWarrior Eclipse IDE. The debugger includes support for all hardware and simulated devices listed in sections 4.2 Device Support.

The AIOP debugger uses a CodeWarrior TAP hardware probe for interfacing with the target. Simultaneous connection with CodeWarrior for ARMv8 debugger is possible from the AIOP debugger using a single run control server (CCS) instance and a single JTAG hardware probe (CodeWarrior TAP).

AIOP debugger is designed to be used in two the following scenarios:

- Attach to AIOP applications included in Freescale Linux SDK
- Download and debug of standalone AIOP applications (available in CodeWarrior examples)

The following AIOP specific features are included in CodeWarrior debugger:

- Dynamic list of tasks running on the AIOP core complex
- Visibility of task specific information: ID, core, entry point name, scheduling state, OSM details, etc.
- Visibility of debug context of background tasks in Debug views: Stack, local variables, core registers, workspace memory, etc.
- Per-task breakpoint functionality
- Per-task run control commands: step and resume

As of v11.2.1 Update 1, the AIOP debugger supports connectivity and debug of the AIOP cores with task awareness capability.

4.7 Software Analysis

4.7.1 New Features

- The v11.2 release included the Beta version of the statistical profiler for AIOP, with some improvements. Instead of program trace, the statistical profiler is based on PC (program counter) and task state samples collected during the execution of the application to profile. The profiler generates a statistical view of this data, offering multiple perspectives of the profiled data:
 - Distribution of tasks running time per function
 - Distribution of accelerator running time per accelerator
 - Distribution of tasks running time and accelerator running time
 - Distribution of running, waiting, and idle time

All perspectives are able to show the source location of the accelerator call instructions executed during the execution.

The profiler is able to collect samples using the JTAG debug connection and using a Linux debug agent.

The Linux debug agent has superior performance (more than 100 times higher sample rate compared to JTAG version). It is able to run continuously, does not have any constraints related to execution time or required space.

Trace

- The Beta 1 release added a new feature that permits removing individual lines or entire code regions from the GCov reports. It uses by default the LCOV comment tags to mark the code. The supported tags are LCOV_EXCL_LINE, LCOV_EXCL_START and LCOV_EXCL_STOP. The support can be enabled or disabled from CodeWarrior Preferences on the Software Analysis page.
- Include code profilers based on program trace for AIOP. The profilers can be used by selecting Performance or Code Coverage views for collected trace. There is an architecturally accepted shortcoming for AIOP that doesn't allow all hardware trace events to be collected by the debug tool (due to limited bandwidth). This limitation is further propagated in code profilers based on program trace.
The mitigation is to use only one or two AIOP cores.
- Added "Upload trace" functionality to overcome limitation of collecting trace only on "Terminate" (user can upload only once on hardware)
- Display accelerator name instead of its code in Trace Viewer

4.7.2 Trace

This release incorporates the tools for extracting the hardware trace for the devices listed in sections 4.2 Device Support. The feature is integrated in CodeWarrior IDE and is also available from command line. There is also the possibility to generate and collect trace for the simulated devices.

Program trace collected from hardware and simulated targets can be visualized in IDE with the following viewers:

- Trace viewer - shows all program events that are found and recognized in the compressed data that was previously collected.
- Timeline – shows the evolution of the execution in time.
- CodeCoverage - shows code coverage data in two views: top view displays functions summary, and bottom view displays which source line and assembly instruction was executed or not for the selected function.
- Performance - shows a hierarchical profiler with invocation information and execution times for each function (with and without children). It does not support task awareness.
- Call Tree - shows all execution paths of the application, highlighting critical execution path (longest call chain). It does not support task awareness.

4.7.3 AIOP Analysis Tool

AIOP Analysis Tool represents a set of advanced AIOP specific functionalities installable on top of CodeWarrior release.

The feature set is summarized as follows:

- Present data in a number of useful views
 - APP element & resource utilization views over time

- Task timeline
- Contention time by resource
- Task & resource utilization correlation view
- Export to CSV for spreadsheet analysis
- Easy to configure and use
 - Operates like a real-time experience with on-chip hardware buffers
 - Option to use offline trace streams
 - Retrieve data using JTAG & CodeWarrior TAP
- Additional features:
 - Zoom in on time spans
 - Markers & measurements
 - Filter, sort, & find
 - Export to CSV for spreadsheet analysis

AIOP Analysis Tool is not available in CodeWarrior by default, it has to be manually installed into the product using the Install New Software functionality (Help->Install New Software -> FSL APP Eclipse Update Site-> AIOP Analysis Tool).

4.8 New Project Wizard

The New Project Wizard allows generation of sample C projects for all hardware and simulated devices listed in sections 4.2 Device Support.

In wizard pages the user is able to configure and enable build, debug, and software analysis settings.

4.9 Product Examples

CodeWarrior for Advanced Packet Processing integrates example projects. For every example, the table below indicates the features that are suitable to be exercised.

The examples are all available for import from the following layout location:

CW_APP\LS\CodeWarrior_Examples\

Project Name	Features	Comments
GCov_Examples	Basic performance statistics on a per source file level such as: <ul style="list-style-type: none"> * how often each line of code executes * what lines of code are actually executed, hardware trace, point to point profiler, and code coverage. 	



The examples are all available for import from the following layout location:

CW_APP\LS\CodeWarrior_Examples\

The examples are all available for import from the following layout location:

CW_APP\LS\CodeWarrior_Examples\

5. Fixed Issues

The examples are all available for import from the following layout location:

CW_APP\LS\CodeWarrior_Examples\

6. Known Issues

The following defects are known to be present in this release. Wherever applicable, the workaround is listed below defect description.

Issue Type	Key	Summary

How to Reach Us:

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP 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 NXP 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. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: <http://www.nxp.com/SalesTermsandConditions>.

NXP, the NXP logo, Freescale, the Freescale logo, Processor Expert, QorIQ are trademarks of NXP B.V. All other product or service names are the property of their respective owners. Arm and Cortex are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

© 2019 NXP B.V.

Error! No text of specified style in document.

31 January 2019

