

Kinetis Design Studio V3.0.0 Release Notes

1 Overview

The Kinetis Design Studio IDE is a complimentary integrated development environment for Kinetis MCUs that enables robust editing, compiling and debugging of your designs. Based on free, open-source software including Eclipse, GNU Compiler Collection (GCC), GNU Debugger (GDB), and others, the Kinetis Design Studio IDE offers designers a simple development tool with no code-size limitations. Furthermore, Processor Expert software enables your design with its knowledge base and helps create powerful applications with a few mouse clicks.

Contents

1	Overview	1
2	Release Contents	2
3	What's New in this Release	2
4	Using Kinetis Design Studio with the Kinetis SDK	3
5	Updates.....	5
6	Devices Supported.....	5
7	Known Issues and Workarounds	12
8	Revision history.....	13

2 Release Contents

- Eclipse Luna 4.4 Framework
- Host operating systems:
 - Windows® 7/8 (32-bit binaries running on 32 and 64-bit OS)
 - Linux® (Ubuntu 14.04 (deb), RedHat/Centos 7 (rpm)) (64 bit binaries), requires 32-bit libraries for GNU ARM Embedded (launchpad) tools
 - Mac OS X 10.10 ("Yosemite") with Segger support
- GDB debugger with support for the following debug interface hardware:
 - SEGGER J-Link (w/SEGGER GDB Server)
 - P&E Multilink (w/P&E GDB Server)
 - CMSIS-DAP (w/OpenOCD GDB and OpenSDA embedded circuit) for selected boards/devices
 - Command Line (CL) debugging with GDB and OpenOCD TCL
- Additional device support is added through Kinetis SDK Eclipse updates.
- Support for additional downloadable Eclipse plug-ins including RTOS-awareness (including MQX™ and FreeRTOS)
- Project wizard to create bare metal, Kinetis SDK and Processor Expert software projects
- Processor Expert software with support for Kinetis SDK
- Languages supported: Assembly, C and C++ (all with no code size restrictions)
- GNU tools for ARM® Embedded Processors (launchpad) 4.8 tool chain and libraries (Q3_2014 release)
- Conversion assistant to convert pre-V3.0.0 KDS projects to GNU ARM Embedded projects
- Industry standard Eclipse Framework with CDT for C/C++
- Kinetis SDK peripheral drivers and CMSIS compliant startup code
- GNU ARM® Eclipse plugins for managed make projects and make-file projects
- Support for MQX project creation and MQX kernel awareness (available as MQX download)
- Compatible with Eclipse plugins either from the Eclipse ecosystem or from partners

Important: because the GNU ARM Embedded (launchpad) tool chain for Linux is 32-bit only, it requires the necessary 32-bit libraries installed. See section 4 of this document. Follow carefully the installation instructions in the User's Guide, both for Mac OS X and Linux.

3 What's New in this Release

The V3.0.0 release comes with the following major changes:

- **Mac OS X Support:** This is the first release with support for Mac OS X "Yosemite" which includes run control/debug support with Segger J-Link. P&E Multilink and OpenOCD support might be added at a later date.
- The **Eclipse IDE** and framework has been moved to Luna (4.4).

- **Launchpad GNU ARM Embedded Toolchain:** instead of a custom toolchain, this release includes the GNU ARM toolchain and libraries as released by ARM (4_8-2014q3-20140805 release) with GNU libraries for devices with less than 1 KByte of RAM. The Project menu contains a **KDS Upgrade Assistant** menu item.
- **EmbSysReg Register Detail Viewer:** A register view to inspect Cortex core and peripheral registers is included in the product.
- **Processor Expert V3.0.0:** This release comes with an updated Processor Expert RT7 release. This release includes support for component repositories and supports the V1.0.0, V1.1.0 and V1.2.0 of the Kinetis SDK. Adding new drivers to an SDK project will automatically add the necessary include folders to the project. The Processor Expert components for the SDK are part of the SDK and need to be installed from the SDK installation folder, see next section.
- **New Project Wizard:** The wizard offers a choice of compilers at the end of the dialogs to choose from GNU (default), IAR and Keil compiler. IAR and Keil projects cannot be used directly with KDS, but with IAR and Keil tool chains (separate from KDS). The wizard offers a better way to select the Kinetis SDK to be used in the project: a drop-box, environment variable or an absolute path.
- Updated **GNU ARM Eclipse** (<http://gnuarmeclipse.livius.net/>) plugins with many bug fixes, improvements and support for switching tool chains. The IDE does not change the PATH to the GNU toolchain any more, instead it uses the project, workspace or global path to the GNU ARM toolchain to make it easy to use different toolchains.
- Updated **SEGGER** and **P&E** debug plugins to the latest and greatest versions. The wizard creates launch configurations for all the supported debug connections automatically and with the device name set.
- **New Welcome View** which points to Kinetis Design Studio and Kinetis SDK web sites.
- The **installer** creates a shortcut in the program group under windows according to the Microsoft guidelines.
- Updated and extended **KDS porting guide** and **KDS User guide** documentation.
- **Bug Fixes:** This release contains numerous bug fixes, enhancements and performance. A detailed list of Processor Expert fixes can be found after installation in the <installation_path>\eclipse\ProcessorExpert folder.

4 Migrating to GNU ARM Embedded (Launchpad) Tools

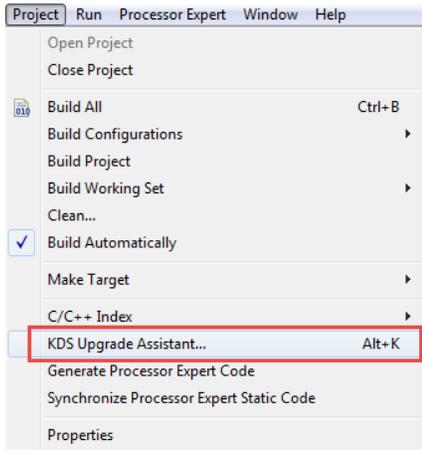
This release uses the GNU ARM Embedded (launchpad) toolchain. To use existing KDS v2.0.0 projects with KDS V3.0.0 requires a small modification in the linker settings. Additional details are documented in the Freescale Kinetis Design Studio community: <https://community.freescale.com/docs/DOC-103698>.

To migrate an existing 'legacy' project to a 'launchpad' project, usually only the linker settings need to be changed. To do this, go into the project properties, and check the 'Other linker flags' settings of the Linker. The table below shows the difference between the two:

Legacy 'Other linker flags'	Launchpad 'Other linker flags'
<none>	-specs=nosys.specs

Legacy 'Other linker flags'	Launchpad 'Other linker flags'
-nanolibc	-specs=nano.specs -specs=nosys.specs

To assist migration of existing (pre-V3.0.0) projects, the Project menu includes a KDS Upgrade Assistant:



Known Issues:

- 1) The converter does not distinguish between Executable projects and Static Library projects. In the latter case the options for converting Newlib-nano and adding semihosting should be suppressed.
- 2) When selecting options to add `_exit()` implementation and `semihosting=rdimon` on projects from KSDK_1.1.0rc3\demos\hello_world\kds a Cross ARM C Linker errors reports "multiple definition of `_isatty`. It is necessary to remove the user definition of `_isatty`.
- 3) If you get a linker error about `_lseed()` (or `_sbrk()`, `_kill()`, etc) undefined, then probably semihosting support has been specified in the conversion assistant.
One solution is to add `-specs=rdimon.specs` to the linker options. This will provide semihosting to the application.
The other option is to add `-specs=nosys.specs` linker option. This will provide 'do-nothing' options to the library, and no text will be printed if your application is using `printf()` or other ways of printing.

GNU ARM Embedded (launchpad) on Linux

Because the GNU ARM Embedded tools are 32-bit only for Linux, on Ubuntu 14.04 64-bit (and others) you may see error messages suggesting that `arm-none-eabi-gcc` could not be found. The tools do exist, however the system doesn't know how to run them. This is because the 32-bit compatibility packages need to be installed: See <http://gnuarmeclipse.livius.net/blog/toolchain-install> for details and suggested solution.

5 Using Kinetis Design Studio with the Kinetis SDK

Before using Kinetis Design Studio with the Kinetis SDK, the KSDK Eclipse Update must be installed. Unless this update is installed, Eclipse cannot generate Kinetis SDK-compatible projects.

To install the necessary files:

1. Have the Kinetis SDK installed (e.g. in C:\Freescale\KSDK_1.2.0)
2. In Kinetis Design Studio, use the menu *Help > Install New Software*
3. Press the Add button
4. In the 'Add Repository' dialog, use the 'Archive' button to browse for the .zip file inside **tools\ eclipse_ update** folder of the SDK
5. Press OK
6. Select all packages (screenshot shown for Kinetis SDK V1.2.0):
 -  KSDK 1.2.0 Eclipse Update
 -  KSDK 1.2.0 Eclipse Update
7. Press 'Next >' to complete the installation.
8. A restart of Eclipse is needed at the end of the installation.

6 Kinetis Design Studio Updates

NOTE: Before running the Eclipse updater, make sure that you have the needed privileges (read/write permissions) for your Eclipse installation folder.

After the release of the product, additional updates can be installed through the Kinetis Design Studio Update Site (<http://freescale.com/lgfiles/updates/Eclipse/KDS>):

1. In Kinetis Design Studio, use the menu *Help > Install New Software*.
2. Select the Freescale KDS Update Site (<http://freescale.com/lgfiles/updates/Eclipse/KDS>) into the "Work with" field-
3. Choose and install the updates.

To update the P&E debugging support, use the following Eclipse Update site:

<http://www.pemicro.com/eclipse/updates>

To update the SEGGER debugging support, download the latest SEGGER software from

<http://www.segger.com/jlink-software.html> and install it. It will detect the installation folder of KDS and updates the drivers.

To update the GNU ARM Eclipse (<http://gnuarmeclipse.livius.net/>) build tools, use the following Eclipse Update site:

<http://gnuarmeclipse.sourceforge.net/updates>

7 OpenOCD

KDS comes with OpenOCD (<http://openocd.sourceforge.net/>) with OpenSDA (<http://www.freescale.com/opensda>) debugging support. The following boards are supported with OpenOCD and CMSIS-DAP/mbed firmware on the board:

- FRDM-KL02Z, FRDM-KL05Z
- FRDM-KL25Z, FRDM-KL26Z, FRDM-KL46Z
- FRDM-K22F, FRDM-K64F
- TWR-K24F120M

All other boards not listed are considered as not supported (e.g. the FRDM-K20 is NOT supported with OpenOCD).

Installation of OpenOCD on Linux

A `udev` rules file is used when installing Kinetis Design Studio on Linux host. You might need to run the rules files after installation in order to set the correct permissions, otherwise OpenOCD will not be able to connect to the board. The `udev` rules file is located in:

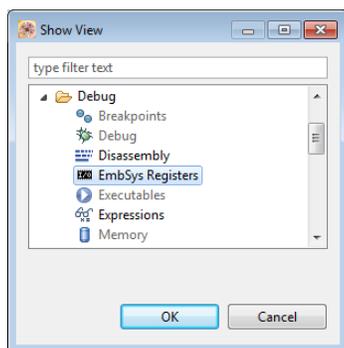
`<installationPath>/openocd/openocd.udev`

To run the rules file:

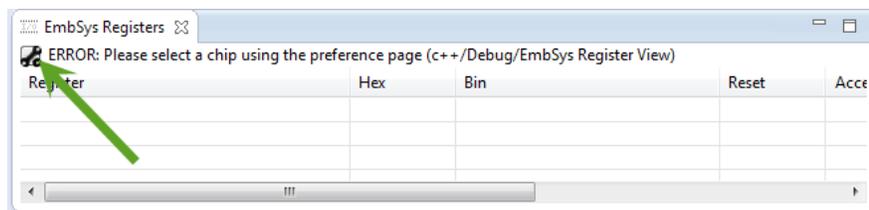
1. Locate the `udev` rules file and copy it into the configuration directory (for example under `/etc/udev/rules.d/`)
2. Rename the `udev` rules file to `99-openocd.rules` (for example)
3. Optionally the permissions allocated by the rules file can be adjusted. By default this requires users to be in the `plugdev` group.
4. Run the command `udevadm control --reload-rules` to instruct `udev` to reload its rules

8 Register Detail Viewer

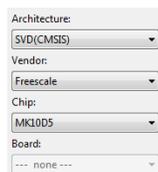
The [EmbSys Register Detail viewer](#) is included with the product. To open the view the first time, use the menu `Window > Show Views > Other > Debug` to open the viewer.



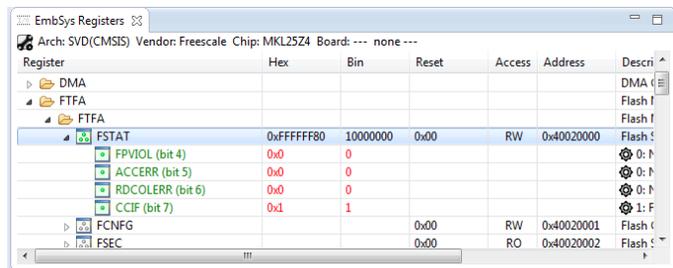
Use the 'wrench' icon to select the chip/device:



Select SVD(CMSIS) for the Architecture with Freescale for the vendor and the matching device/chip:



Double click on a register to read/modify the values:



The values in green color are read from the device and can be modified.

9 Devices Supported

Kinetis Design Studio supports a broad range of Freescale Kinetis devices, and more can be added with the Eclipse updater mechanism and serviced packs.

The following devices are supported with the Kinetis SDK (requires Kinetis SDK installed and Kinetis SDK Eclipse Update Files installed in KDS).

Devices/Boards listed in the New Project Wizard (File > New Kinetis Project) (screenshot shows boards listed after adding Kinetis SDK v1.2):

FRDM-K22F	TWR-K21D50M
FRDM-K64F	TWR-K21F120M
FRDM-KL02Z	TWR-K22F120M
FRDM-KL03Z	TWR-K24F120M
FRDM-KL25Z	TWR-K60D100M
FRDM-KL26Z	TWR-K64F120M
FRDM-KL27Z	TWR-K65F180M
FRDM-KL43Z	TWR-KL43Z48M
FRDM-KL46Z	TWR-KV10Z32
FRDM-KW24	TWR-KV31F120M
MRB-KW019030JA	TWR-KV46F150M
MRB-KW019032EU	TWR-KW24D512
MRB-KW019032NA	USB-KW24D512

Devices supported (note that additional device support and board support gets installed with the Kinetis SDK Eclipse tool updates, e.g MKL03Z32xxx4, MKL33Z64xxx4, MKL43Z256xxx4, MKM34Z256xxx7, MKV46F256xxx15):

<ul style="list-style-type: none"> ▲ KE02Z / KEAZN (20MHz,40MHz) <ul style="list-style-type: none"> MKE02Z16xxx2 MKE02Z16xxx4 MKE02Z32xxx2 MKE02Z32xxx4 MKE02Z64xxx2 MKE02Z64xxx4 SKEAZN16xxx2 SKEAZN32xxx2 SKEAZN64xxx2 ▲ KE04Z / KEAZN (48MHz) <ul style="list-style-type: none"> MKE04Z128xxx4 MKE04Z64xxx4 MKE04Z8xxx4 SKEAZN8xxx4 ▲ KE06Z / KEAZ (48MHz) <ul style="list-style-type: none"> MKE06Z128xxx4 MKE06Z64xxx4 SKEAZ128xxx4 SKEAZ64xxx4 	<ul style="list-style-type: none"> ▲ MK10D (50MHz) <ul style="list-style-type: none"> MK10DN128xxx5 MK10DN32xxx5 MK10DN64xxx5 MK10DX128xxx5 MK10DX32xxx5 MK10DX64xxx5 ▲ MK10D (72MHz) <ul style="list-style-type: none"> MK10DX128xxx7 MK10DX256xxx7 MK10DX64xxx7 ▲ MK10D (100MHz) <ul style="list-style-type: none"> MK10DN512Zxxx10 MK10DN512xxx10 MK10DX128Zxxx10 MK10DX128xxx10 MK10DX256Zxxx10 MK10DX256xxx10 ▲ MK10F (120MHz, 150MHz) <ul style="list-style-type: none"> MK10FN1M0xxx12 MK10FX512xxx12 ▲ MK11D (50MHz) <ul style="list-style-type: none"> MK11DN512Axxx5 MK11DN512xxx5 MK11DX128Axxx5 MK11DX128xxx5 MK11DX256Axxx5 MK11DX256xxx5 ▲ MK12D (50MHz) <ul style="list-style-type: none"> MK12DN512xxx5 MK12DX128xxx5 MK12DX256xxx5 	<ul style="list-style-type: none"> ▲ MK20D (50MHz) <ul style="list-style-type: none"> MK20DN128xxx5 MK20DN32xxx5 MK20DN64xxx5 MK20DX128xxx5 MK20DX32xxx5 MK20DX64xxx5 ▲ MK20D (72MHz) <ul style="list-style-type: none"> MK20DX128xxx7 MK20DX256xxx7 MK20DX64xxx7 ▲ MK20D (100MHz) <ul style="list-style-type: none"> MK20DN512Zxxx10 MK20DN512xxx10 MK20DX128Zxxx10 MK20DX128xxx10 MK20DX256Zxxx10 MK20DX256xxx10 ▲ MK20F (120MHz, 150MHz) <ul style="list-style-type: none"> MK20FN1M0xxx12 MK20FX512xxx12 ▲ MK21D (50MHz) <ul style="list-style-type: none"> MK21DN512Axxx5 MK21DN512xxx5 MK21DX128Axxx5 MK21DX128xxx5 MK21DX256Axxx5 MK21DX256xxx5 ▲ MK21F (120MHz) <ul style="list-style-type: none"> MK21FN1M0Axxx12 MK21FN1M0xxx12 MK21FX512Axxx12 MK21FX512xxx12 ▲ MK22D (50MHz) <ul style="list-style-type: none"> MK22DN512xxx5 MK22DX128xxx5 MK22DX256xxx5 ▲ MK22F (100MHz) <ul style="list-style-type: none"> MK22FN128xxx10 ▲ MK22F (120MHz) <ul style="list-style-type: none"> MK22FN1M0xxx12 MK22FN256xxx12 MK22FN512xxx12 MK22FX512xxx12 ▲ MK24F (120MHz) <ul style="list-style-type: none"> MK24FN1M0xxx12
---	---	---

<ul style="list-style-type: none"> ▲ MK30 <ul style="list-style-type: none"> ▲ MK30D (72MHz) <ul style="list-style-type: none"> MK30DX128xxx7 MK30DX256xxx7 MK30DX64xxx7 ▲ MK30D (100MHz) <ul style="list-style-type: none"> MK30DN512Zxxx10 MK30DN512xxx10 MK30DX128Zxxx10 MK30DX128xxx10 MK30DX256Zxxx10 MK30DX256xxx10 	<ul style="list-style-type: none"> ▲ MK40 <ul style="list-style-type: none"> ▲ MK40D (72MHz) <ul style="list-style-type: none"> MK40DX128xxx7 MK40DX256xxx7 MK40DX64xxx7 ▲ MK40D (100MHz) <ul style="list-style-type: none"> MK40DN512Zxxx10 MK40DN512xxx10 MK40DX128Zxxx10 MK40DX128xxx10 MK40DX256Zxxx10 MK40DX256xxx10 	<ul style="list-style-type: none"> ▲ MK50 <ul style="list-style-type: none"> ▲ MK50D (72MHz) <ul style="list-style-type: none"> MK50DX128xxx7 MK50DX256xxx7 MK51DX128xxx7 MK51DX256xxx7 ▲ MK50D (100MHz) <ul style="list-style-type: none"> MK50DN512Zxxx10 MK50DN512xxx10 MK50DX256Zxxx10 MK50DX256xxx10 MK51DN256Zxxx10 MK51DN256xxx10 MK51DN512Zxxx10 MK51DN512xxx10 MK51DX256Zxxx10 MK51DX256xxx10 MK52DN512Zxxx10 MK52DN512xxx10 MK53DN512Zxxx10 MK53DN512xxx10 MK53DX256Zxxx10 MK53DX256xxx10
--	--	---

<ul style="list-style-type: none"> ▲ MK60 <ul style="list-style-type: none"> ▲ MK60D (100MHz) <ul style="list-style-type: none"> MK60DN256Zxxx10 MK60DN256xxx10 MK60DN512Zxxx10 MK60DN512xxx10 MK60DX256Zxxx10 MK60DX256xxx10 ▲ MK60F (120MHz, 150MHz) <ul style="list-style-type: none"> MK60FN1M0xxx12 MK60FN1M0xxx15 MK60FX512xxx12 MK60FX512xxx15 ▲ MK61F (120MHz, 150MHz) <ul style="list-style-type: none"> MK61FN1M0xxx12 MK61FN1M0xxx15 MK61FX512xxx12 MK61FX512xxx15 ▲ MK63F (120MHz) <ul style="list-style-type: none"> MK63FN1M0xxx12 ▲ MK64F (120MHz) <ul style="list-style-type: none"> MK64FN1M0xxx12 MK64FX512xxx12 	<ul style="list-style-type: none"> ▲ MK70 <ul style="list-style-type: none"> ▲ MK70F (120MHz, 150MHz) <ul style="list-style-type: none"> MK70FN1M0xxx12 MK70FN1M0xxx15 MK70FX512xxx12 MK70FX512xxx15 	<ul style="list-style-type: none"> ▲ MKL0x <ul style="list-style-type: none"> ▲ KL02Z (48 MHz) <ul style="list-style-type: none"> MKL02Z16xxx4 MKL02Z32xxx4 MKL02Z8xxx4 ▲ KL04Z (48 MHz) <ul style="list-style-type: none"> MKL04Z16xxx4 MKL04Z32xxx4 MKL04Z8xxx4 ▲ KL05Z (48 MHz) <ul style="list-style-type: none"> MKL05Z16xxx4 MKL05Z32xxx4 MKL05Z8xxx4
<ul style="list-style-type: none"> ▲ MKL1x <ul style="list-style-type: none"> ▲ KL14Z (48 MHz) <ul style="list-style-type: none"> MKL14Z32xxx4 MKL14Z64xxx4 ▲ KL15Z (48 MHz) <ul style="list-style-type: none"> MKL15Z128xxx4 MKL15Z32xxx4 MKL15Z64xxx4 ▲ KL16Z (48 MHz) <ul style="list-style-type: none"> MKL16Z128xxx4 MKL16Z256xxx4 MKL16Z32xxx4 MKL16Z64xxx4 	<ul style="list-style-type: none"> ▲ MKL2x <ul style="list-style-type: none"> ▲ KL24Z (48 MHz) <ul style="list-style-type: none"> MKL24Z32xxx4 MKL24Z64xxx4 ▲ KL25Z (48 MHz) <ul style="list-style-type: none"> MKL25Z128xxx4 MKL25Z32xxx4 MKL25Z64xxx4 ▲ KL26Z (48 MHz) <ul style="list-style-type: none"> MKL26Z128xxx4 MKL26Z256xxx4 MKL26Z32xxx4 MKL26Z64xxx4 	<ul style="list-style-type: none"> ▲ MKL3x <ul style="list-style-type: none"> ▲ KL34Z (48 MHz) <ul style="list-style-type: none"> MKL34Z64xxx4 ▲ KL36Z (48 MHz) <ul style="list-style-type: none"> MKL36Z128xxx4 MKL36Z256xxx4 MKL36Z64xxx4

<ul style="list-style-type: none"> ▲ MKV1x <ul style="list-style-type: none"> ▲ KV10Z (75 MHz) <ul style="list-style-type: none"> MKV10Z128xxx7 MKV10Z16xxx7 MKV10Z32xxx7 MKV10Z64xxx7 	<ul style="list-style-type: none"> ▲ MKV3x <ul style="list-style-type: none"> ▲ MKV31F (100MHz) <ul style="list-style-type: none"> MKV31F128xxx10 ▲ MKV31F (120MHz) <ul style="list-style-type: none"> MKV31F256xxx12 MKV31F512xxx12 	<ul style="list-style-type: none"> ▲ MKW0x <ul style="list-style-type: none"> ▲ KW01Z (48 MHz) <ul style="list-style-type: none"> MKW01Z128xxx4 ▲ MKW2x <ul style="list-style-type: none"> ▲ KW21D (50 MHz) <ul style="list-style-type: none"> MKW21D256xxx5 MKW21D512xxx5 ▲ KW22D (50 MHz) <ul style="list-style-type: none"> MKW22D512xxx5 ▲ KW24D (50 MHz) <ul style="list-style-type: none"> MKW24D512xxx5
--	---	---

10 Known Issues and Workarounds

For the latest information, training material and Frequently Asked Questions, visit the Kinetis Design Studio Community at <https://community.freescale.com/community/kinetis-design-studio>.

Note: Check the Freescale downloads from the web (<http://www.freescale.com/kds>) and the updates (menu Help > Install New Software using the KDS update site (<http://freescale.com/lgfiles/updates/Eclipse/KDS>)).

- **Windows Command Line Limit, KDS-178:** Under Windows there is a command line length limitation of 8192 characters. If the command line length to the compiler or linker exceed that limit, the command line gets truncated and the build will fail.
Workaround: Shorten the path to files and folders as much as possible. Instead of absolute paths, use project relative paths or use the Windows 'subst' drive letters to create shorter paths to the sources. Alternatively, build libraries and link large projects with libraries instead of individual object files. Another solution is to use the GNU ARM Eclipse Build Tools from <https://sourceforge.net/projects/gnuarmeclipse/files/Build%20Tools/>, see this article: <http://mcuoneclipse.com/2015/03/29/solving-the-8192-character-command-line-limit-on-windows/>.
- **Administrative rights to Eclipse installation folder for updates:** Users must have write access to the KDS installation directory to install new Eclipse plugins. This means that on Linux users need to launch KDS with root privileges when installing new plugins.
- **Conditional watchpoints and breakpoints:** Conditional breakpoints and watchpoints, including those using ignore counts, do not work always.
Workaround: do not use conditions for breakpoints and watchpoints, instead check for condition in the code and set a normal breakpoint.
- **Symbolic Link to libudev for Linux:** Like many other Linux packages, users of Ubuntu 14.04 must create a symbolic link to libudev.
Workaround:
`ln -s /lib/x86_64-linux-gnu/libudev.so.1.3.5 /usr/lib/libudev.so.0`
- **Installation time on Ubuntu:** Users attempting to install KDS using the Ubuntu Software Center may find that the Software Center claims to be installing for a long period of time, then returns to the start screen without emitting an error or installing the product. This is because the Software Center runs a quality checking tool, lintian, on the package before installing it. This tool is not implemented in a scalable manner, and doesn't handle the large KDS packages well. Users of high-end machines may find they are able to install successfully.
Workaround: install using the command-line tool: `dpkg`
- **Build binary not found:** Occasionally after a successful build Eclipse does not find the built binary. This can manifest in a number of ways: The project does not show the Binaries meta-folder in the Project Explorer view, when the debug button on the debug toolbar is pressed the Debug configuration fails to launch a debug session because it reports binary file not found.
Workaround: refresh the project folder (F5 under Windows).
- PEXMCU-531: Compiler error if using the **TSS Processor Expert component**. The compiler is because the component sources are using `asm()` instead of `_asm()`. Workaround described in <https://community.freescale.com/message/435546#435546>
- **KDS-223: Doing a reset command on the FRDM-K22F** and OpenOCD debug connection gets stuck in the watchdog reset handler. Workaround is to use an alternative debug connection (P&E Multilink or SEGGER J-Link).
- **KDS-190: Stepping over an endless loop** (branch instruction pointing to itself) might fail with OpenOCD.
Workaround: set a breakpoint on the loop statement.

- KDS-189: **Debugging, downloading and stepping with OpenOCD** is slow compared to other debug solutions. Workaround is to use alternative debug connections like P&E Multilink or SEGGER J-Link.
- KDS-240: The **OpenOCD** in KDS v3.0.0 is the same as in KDS v2.0.0, therefore any newer Kinetis devices are not supported with OpenOCD. Workaround is to use an alternative debug connection (P&E Multilink or SEGGER J-Link).
- Using **TSS** component: adding the TSS component for non-Kinetis SDK Processor Expert project will cause a compilation error. Workaround is documented here: <https://community.freescale.com/thread/330174>
- **Mac OS X and Eclox**: The installation of the Doxygen Eclox Eclipse plugin (<http://home.gna.org/eclox/>) fails under Mac OS X with an error message. That plugin is not maintained anymore and a fix is not likely.
- **Moving/using projects between different host operating systems**: If using a project created/used on a different operating system (e.g. migrating a project from Windows to Linux), it is recommended to delete the output (usually named 'Debug') folder of the project to enforce proper regeneration of the make files, as a 'clean' operation alone might not be enough.
- The GNU ARM Eclipse plugins have now a linker panel check box to specify the --specs=nano.specs. Checking that box and in addition to have this option present in the 'Other linker flags' will cause the link phase to fail. The solution is to specify the option only once.
- KDS-277: Creating a new project with the wizard, for some devices (KW01Z128, KV32, K70, K60, K53, K51, K50, K30, K20, K10) it is possible that the debug configuration settings for P&E have not the device name populated, and debugging will fail. Workaround: In the debug/launch configuration for P&E, select the matching device.
- KDS300KI004: Creating a new Processor Expert Board (MRB-KW019032E, FRDM-KL25Z, FRDM-KL26Z and TWR-KW24D512) project without Kinetis SDK enabled result in build errors because of Target Processor not selected. Workaround: Create the project not for the board, but use the microcontroller of the board during project creation instead.

11 Revision history

Table 12. summarizes revisions to this document.

Table 12. Revision History	
Revision	Change description
Rev. 0	Initial Version
Rev. 1	Added items after formal release.

How to Reach Us:

Home Page:

www.freescale.com

Web Support:

www.freescale.com/support

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

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, Kinetis, Processor Expert, and CodeWarrior are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. 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 EU and/or elsewhere. mbed is a trademark of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. Kinetis Design Studio is produced for Freescale by SOMNIUM™ Technologies <http://www.somniumtech.com>. All rights reserved.

© 2014-2015 Freescale Semiconductor, Inc.

© 2013-2015 SOMNIUM™ Technologies Limited.

