

S32 Design Studio for Power Architecture version 2.1

Contents

1. Release Description.....	2
1.1. Release content.....	2
1.2. Supported Devices.....	2
2. What's New.....	3
2.1. New device support.....	3
2.2. New Features.....	3
2.3. Bug fixes.....	4
3. System Requirements for Windows Host.....	4
3.1. Recommended configuration.....	4
3.2. Operational minimum configuration.....	4
3.3. Host operating system support.....	5
4. System Requirements for Linux Host.....	5
4.1. Recommended configuration.....	5
4.2. Operational minimum configuration.....	5
4.3. Host operating system support.....	5
5. Product Web Page.....	6
6. Installation and Licensing.....	6
8. Technical Support.....	6
Appendix A. Known issues and Workarounds.....	7
Appendix B. Performance Considerations.....	8



1. Release Description

NXP Semiconductors is pleased to announce the release of the S32 Design Studio for Power Architecture 2.1 for NXP Automotive Power Architecture[®]-based MCUs. S32 Design Studio is based on the Eclipse open development platform and integrates the Eclipse IDE, GNU Compiler Collection (GCC), GNU Debugger (GDB), and other software to offer designers a straightforward development tool with no code-size limitations.

1.1. Release content

- Eclipse Neon 4.6 Framework
- GNU Build Tools for e200 processors (VLE and BookE ISA support, based on gcc 4.9.4, binutils 2.28 and gdb 7.8.2, build 1607 revision gceb1328)
- Libraries: NewLib, NewLib Nano, Freescale EWL2
- MSYS2 32bit version 20161025
- The wizards for creating application, library projects and projects from project examples for the supported processor families
- SPT, SPT2 and SPT2.5 tools
- Visual Graph Tool to support SPT targeted software development
- Integrated S32 SDK for Power Architecture RTM 3.0.0 (Windows only)
- Integrated AMMCLIB SDKs for MPC56xxx/MPC57xxx version 1.1.15
- FreeMASTER Serial Communication Driver v2.0
- Diab compiler support
- Green Hills compiler support¹
- PEMicro debugger support (P&E Multilink/Cyclone/OpenSDA)
- Lauterbach Trace32 debugger support
- iSYSTEM debugger support
- PLS debugger support
- EmbSys and SPR Registers views
- Kernel Aware debugging for FreeRTOS, eCOS, OSEK
- SDK management
- The Getting Started page
- The S32DS Extensions and Updates tool

1.2 Supported Devices

- S32R274
- S32R372
- MPC5775B, MPC5775E

¹The Green Hills Software (GHS) compiler support depends on the availability of the Eclipse plug-in integrating GHS compiler compatible with the Eclipse Neon. Full support for the Green Hills Software compiler Eclipse Neon-compatible plug-in within S32DS for PA 2.1 was not validated in time for this release.

- MPC5775K, MPC5774K
- MPC5746R, MPC5745R, MPC5743R
- MPC5777M
- MPC5777C
- MPC5748G, MPC5747G, MPC5746G
- MPC5745B, MPC5744B, MPC5746B, MPC5744C, MPC5745C, MPC5746C
- MPC5744P, MPC5743P, MPC5742P, MPC5741P
- MPC5601P, MPC5602P, MPC5603P, MPC5604P
- MPC5644B, MPC5644C, MPC5645B, MPC5645C, MPC5646B, MPC5646C
- MPC5601D, MPC5602B, MPC5602C, MPC5602D, MPC5603B, MPC5603C, MPC5604B, MPC5604C, MPC5605B, MPC5606B, MPC5607B
- MPC5606S
- MPC5604E
- MPC5644A
- MPC5643L
- MPC5676R
- MPC5632M, MPC5633M, MPC5634M
- MPC5642A
- MPC5674F
- MPC5673K, MPC5674K, MPC5675K

2. What's New

This release comes with several new features, improvements and bug fixes.

2.1. New device support

- MPC5775B, MPC5775E

2.2. New Features

- The S32DS Extensions and Updates tool for automatic lookup and on-demand installation of software packages with SDKs and support for the NXP Automotive Power Architecture[®] based processor families
- The Getting Started page enables navigation through the S32 Design Studio structure and provides collateral materials such as videos and How-To articles
- The default product directory structure is modified
- The new GNU Tools for e200 processors version. For details, refer to the release notes located in `S32DS/build_tools/powerpc-eabivle-4_9/`
- The new AMMCLIB SDK version. For details, refer to the release notes located in `S32DS/software/AMMCLIB_1.1.15/MPC5xxxx_AMMCLIB_v1.1.15/`
- The new S32 SDK for Power Architecture version (for Windows only). For details, refer to the release notes located in `S32DS/software/S32_SDK_S32PA_RTM_3.0.0/`

- Support for importing MCAL configuration to a custom SDK
- An SDK can be attached to a library project using the project wizard
- The updated documentation set includes the Migration guide to help migrate projects from an earlier version to S32 Design Studio for Power Architecture 2.1

2.3. Bug fixes

- For detailed list of the GNU Tools bug fixes, refer to the release notes located in S32DS/build_tools/powerpc-eabivle-4_9/
- Fixed the semihosting issues with the EWL and NewLib libraries
- Fixed the FLASH programming algorithm for MPC5744P
- Added missing linker script sections for MPC5748G
- Fixed reading values from the peripheral bridge A registers for MPC5634M
- Fixed access to the RAM memory for MPC5634M
- Removed unavailable addresses from the MPC574xB linker files
- Added the `-fstrict-volatile-bitfield` compiler option to the project settings
- Fixed secure connection to MPC5744P
- Disabled RTTI for EWL library due to incompatibility with the GNU tools
- Fixed importing/exporting projectinfo.xml with library settings

3. System Requirements for Windows Host

3.1. Recommended configuration

- PC with 2.6 GHz Intel® Pentium® compatible processor or better
- 4 GB of RAM
- 8.5 GB of disk space
- 3.5 GB of temporary storage (required only during the product installation)
- USB port for communications with target hardware
- Ethernet port for communications with target hardware (optional)

3.2. Operational minimum configuration

- PC with 1.8 GHz Intel® Pentium® compatible processor
- 2 GB of RAM
- 8.5 GB of disk space
- 3.5 GB of temporary storage (required only during the product installation)
- USB port for communications with target hardware

3.3. Host operating system support

- Microsoft® Windows® 7 32-bit and 64-bit
- Microsoft® Windows® 8/8.1 32-bit and 64-bit
- Microsoft® Windows® 10 32-bit and 64-bit

S32 Design Studio for Power Architecture 2.1 supports all editions of the operating systems listed above and is limited only by the requirements of the Java Runtime Environment.

4. System Requirements for Linux Host

4.1. Recommended configuration

- PC with 2.6 GHz Intel® Pentium® compatible processor or better
- 4 GB of RAM
- 7.5 GB of disk space for installation files (full product or updates)
- 3 GB of temporary storage (required only during the product installation)
- Java 1.8
- USB port for communications with target hardware
- Ethernet port for communications with target hardware (optional)

S32 Design Studio for Power Architecture 2.1 Installation Guide specifies the installation prerequisites for Linux platforms.

4.2. Operational minimum configuration

- PC with 1.8 GHz Intel® Pentium® compatible processor
- 2 GB of RAM
- 7.5 GB of disk space for installation files (full product or updates)
- 3 GB of temporary storage (required only during the product installation)
- Java 1.8
- USB port for communications with target hardware

4.3. Host operating system support

- Ubuntu LTS 16.04 64-bit
- Debian 8 64-bit
- CentOS 7 64-bit

S32 Design Studio for Power Architecture 2.1 supports all editions of the operating systems listed above and is limited only by the requirements of the Java Runtime Environment.

5. Product Web Page

S32 Design Studio for Power Architecture 2.1 product page is www.nxp.com/S32DS. It contains general information about the product, updates, and download links.

6. Installation and Licensing

The installation package contains the complete S32 Design Studio for Power Architecture 2.1 suite and the installer. All data needed by the installer is included in the package and no other download will be performed.

Note: The plug-ins to support third-party compilers or debuggers such as Lauterbach Trace32® are not included in the installation package and have to be installed from the corresponding sites or installation packages. If your computer is connected to the Internet, select **Install New Software** on the **Help** menu to find and install all available plug-ins. If your computer does not have access to the Internet, you can download and install them from the archive files.

Run the installer. The wizard will guide you through the installation process.

Note: Installation of S32 Design Studio for Power Architecture 2.1 from the command line interface in the console or silent mode is not supported.

During installation, license activation is requested. The following activation types are supported:

- Online activation requires access to the Internet. A license activation request is sent automatically.
- Offline activation requires no Internet access. A license activation request is generated and needs to be passed to the licensing site manually. The activation response is loaded from the site to the installer manually.

New functionality can be added to S32 Design Studio for Power Architecture 2.1 with updates and patches. Updates and patches extend general functionality of the product and correct software defects. If your computer is connected to the Internet, select **S32DS Extensions and Updates** on the **Help** menu to find and install all updates and packages available. If your computer does not have access to the Internet, you can download software packages, updates, and patches from the product page and install them from the archive files using the **S32DS Extensions and Updates** tool.

8. Technical Support

- The S32 Design Studio general issues are tracked through the S32DS Public NXP Community space:
<https://community.nxp.com/community/s32/s32ds>
- The S32 SDK issues are tracked through the S32 SDK Public NXP Community space:
<https://community.nxp.com/community/s32/s32sdk>
- For confidential cases and cases which cannot be publicly shared on NXP Community please follow the steps described here:

<https://community.nxp.com/docs/DOC-329745>

- Submit a support request for P&E Microcomputer Systems:

<http://www.pemicro.com/support/>

Appendix A. Known issues and Workarounds

Note: There are some issues which are introduced by Eclipse CDT and are therefore reproduced in the S32 Design Studio. These issues might be fixed when the fix is available in the future version of Eclipse CDT and the S32 Design Studio migrates to the updated CDT.

- **Conditional watchpoints and breakpoints:** Conditional breakpoints and watchpoints, including those using ignore counts, may not work in some cases.

Workaround: Avoid using conditions for breakpoints and watchpoints, instead check for condition in the code and set a normal breakpoint.

- **Disassembly view** content might be destroyed occasionally.

Workaround: Close the Disassembly view and open it again by selecting **Window > Show View > Disassembly** on the menu.

- **Hyperlinks** may not work correctly on Windows 10 when Microsoft Edge is configured as the default browser.

Workaround: Set a different browser as the default one.

- **Views** are not updated instantly after executing a command in the GDB console.

Workaround: Use the `step` command to see refreshed views.

- The **Getting Started** page may be displayed blank in Linux.

Workaround: Install Webkit1 for GTK2 using the following command: `sudo apt-get install libwebkitgtk-1.0-0`

- The **Getting Started** page may fail to load content at startup of S32 Design Studio for Power Architecture 2.1. “The page can't be displayed” error message may be shown instead.

Workaround: Refresh the **Getting Started** page by pressing the F5 key.

- **Debugging an elf file generated by the GHS compiler** may not work correctly: missing call stack information, incorrect info on step into function, etc.

Workaround: In case of GHS generates the GDB incompatible debug information, it prevents the debugger displaying the call stack. Sometimes gdb can still debug your program without this information. The post-build step should be executed: open the project properties, click **C/C++ Build > Settings** and select the **Build Steps** tab, then add the following command: `gstrip -r=.debug_frame`. Switch to Instruction Stepping Mode to be able to step through function. If this workaround does not help – please report issue to the GreenHills support.

- **Debugging from flash fails for the multi-core project with GHS compiler and attached SDK:** The GHS compiler defines the `TURN_ON_CPU1` (`TURN_ON_CPU2`) macros for the boot core ASM files.

Workaround: Open the boot core project properties, click **C/C++ Build > Settings**, select the flash configuration and open the **Build Steps** tab. Then add the following post-build command: `${cross_prefix}objcopy -R .cpu1_reset_vector -R .cpu2_reset_vector ${ProjName}.elf`.

- **Duplicated Content type and Language** in the Language Mappings settings: When adding new language mapping in the project properties (**C/C++ General > Language Mappings**) or global preferences (**C/C++ > Language Mappings**), the Content type and Language drop-down lists can display duplicated values.

Workaround: Select any option from the lists, all options are valid.

- **Creating new project in the root of drive C:\ fails:** When creating a new project in the root directory, the project wizard shows the "Can't create directory: C:\project_name" error message. Windows does not allow user to create files in this location.

Workaround: Create a subdirectory in the root directory (for example, C:\projects) and specify this location in the project creation wizard.

- **Building project after generating the PEx code may fail:** The build process can be run in parallel with the PEx code generating. The missing source files cause build fail.

Workaround: Make sure that generating the PEx code is finished: open the **Progress** view (**Window > Show View > Other... > General > Progress**), wait until the generating is complete and build the project again.

- **Debug and runtime issues with the multi-core device project with attached S32 SDK for Power Architecture:** When creating a new project for multi-core device and disabling some secondary cores, the SDK still defines the `TURN_ON_CPUx` macros, where x stands for the unused core. The device behavior is not defined and might lead to unexpected results.

Workaround: Open the boot core project properties, click **C/C++ Build > Settings** and remove `TURN_ON_CPUx` for unused cores from the **Defined symbols** lists on the following pages: **Standard S32DS C/C++ Compiler > Preprocessor**, **Standard S32DS Assembler > Symbols**.

- **Unable to start a new GDB server session** after the S32DS for Power Architecture is not responding: When IDE is not responding and the running process must be stopped using Task Manager, the GDB server process is not killed and prevents starting a new session.

Workaround: Open Task Manager and stop the `pegdbserver_power_console` process.

Appendix B. Performance Considerations

The following suggestions will help you keep the S32 Design Studio tools running at a respectable performance level.

1. To maximize the performance, the S32 Design Studio tools should be installed on a computer with the recommended configuration. While the tools will operate on a computer with the operational

minimum configuration, limited hardware will restrict its ability to function at desired performance levels.

2. Close unused projects. Eclipse caches files for all open projects in the workspace. If you need multiple projects open, try to limit the number of projects to no more than 10.
3. The Eclipse IDE provides several options that provide user assistance tools. These options, however, use memory and CPU resources. If the performance is slow, and you do not need these options, turn them off.
4. Scalability mode options configure how Eclipse deals with large source files and may affect the overall performance of the IDE. Turn off these options to maximize the performance. The following scalability mode options are available: editor live parsing, semantic highlighting, syntax coloring, parsing-based content assist proposals, and content assist auto activation.

To disable:

- a. Choose **Window > Preferences**
 - b. Expand **C/C++ > Editor > Scalability**
 - c. Uncheck **Enable all scalability mode options**
5. Content Assist Auto Activation can reduce the number of keystrokes one must type to create the code. The Content Assist plug-in consists of components that predict what keystroke you may type next, based on the current context, scope and prefix. Turn off this predictor to maximize the performance.

To disable:

- a. Choose **Window > Preferences**
- b. Expand **C/C++ > Editor > Content Assist**
- c. Uncheck all the options for **Auto-Activation**



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 based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

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: nxp.com/SalesTermsandConditions.

Registered trademarks: NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C-5, CodeTest, CodeWarrior, ColdFire, ColdFire +, C-Ware, Energy Efficient Solutions logo, Kinetis, mobileGT, PowerQUICC, Processor Expert, QorIQ, Qorivva, StarCore, Symphony, and VortiQa are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. and Tm. Off. Airfast, BeeKit, BeeStack, CoreNet, Flexis, Layerscape, MagniV, MXC, Platform in a Package, QorIQonverge, QUICC Engine, Ready Play, SafeAssure, SafeAssure logo, SMARTMOS, Tower, TurboLink, Vybrid, and Xtrinsic are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

Arm, the Arm logo, 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. All rights reserved.

IEEE nnn, nnn, and nnn are registered trademarks of the Institute of Electrical and Electronics Engineers, Inc. (IEEE). This product is not endorsed or approved by the IEEE. Java are registered trademarks of Oracle and/or its affiliates. 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.

Document Number: S32DSPA21

Rev. 1.0

06/2019

