UG10060

MCUXpresso IDE 11.10.0 Installation Guide

Rev. 3 — 1 July 2024 User guide

Document information

Information	Content
Keywords	MCUXpresso, MCUXpresso IDE
Abstract	MCUXpresso is a low-cost development platform available from NXP. This document explains how to install MCUXpresso IDE.

1 Installation

MCUXpresso IDE is installed with a base set of drivers and built-in support for a large range of LPC MCUs and native debug connections via LinkServer (CMSIS-DAP). You can add additional part (MCU) support at any time by downloading and installing the required SDK packages.

Support for SEGGER J-Link Debug probes and PEmicro Debug probes is also installed by default.

Note: From MCUXpresso IDE version 10.2.0, only one product variant is available, replacing the previous Free and Pro Editions. MCUXpresso IDE now includes *out-of-the-box* all features previously restricted to the Pro Edition. It does not require any activation procedure and contains no limitations on build or debug code sizes.

Note: From MCUXpresso IDE version 11.9.0, LinkServer support is installed as a separate package, in a separate location. LinkServer has become an independent tool, with its own installer, that you can use as a standalone command line utility. The main MCUXpresso IDE installer automatically installs LinkServer, therefore no manual steps are required. You can find more information about NXP's LinkServer solution on the official LinkServer website and inside the documentation page, available after installation - see *mcuxpresso_install_dir/ide/LinkServer/Readme.md*.

1.1 Host computer requirements

Before installing MCUXpresso IDE, you should make sure that your development host computer meets the following requirements:

- A standard x64 host with 8 GB RAM minimum, 4 GB of available disk space (although more may be required depending on the number of SDKs installed), a recommended screen resolution of 1080p or better and running one of the operating systems specified below.

Downloading of SDKs, product updates, and the use of Config Tools require an Internet connection.

Important Note: To ensure that debug probe drivers can be installed correctly, ensure that you remove all USB debug probes before you perform the MCUXpresso IDE installation.

1.2 Windows

- Microsoft(r) Windows 10 and Windows 11

Note: **From MCUXpresso IDE version 11.0.0, only 64-bit Windows is supported.**

MCUXpresso IDE is installed into a single directory of your choice. Unlike many software packages, MCUXpresso IDE does not install or use any keys in the Windows Registry, or use or modify any environment variables (including PATH), resulting in a clean installation that does not interfere with anything else on your PC. However, third-party debug probe support code may make such modifications.

At a certain point, the installer prompts you to install various drivers. These are required for correct operation, and include:

- Philips (NXP) Universal Serial Bus
  – This can take some time to complete
- Jungo Connectivity and Jungo Ltd
  – These drivers are installed by the PEmicro debug plugin
- PEmicro
- Ashling/NXP

The installer also silently installs drivers for:

- SEGGER
- LPC-Link

- LPC-Link2
- RedProbe+
- RDB-Link
- MCU-Link

After the installation has been completed, to use some Kinetis boards with OpenSDA Mbed CMSIS-DAP debug connection and LPCXpresso Max boards, an Mbed Serial Port driver is required. You can download the driver via the IDE link at:

```
Help -> Additional Resources -> MBED Serial Port Driver Website
```

Without this driver, the IDE cannot find the Mbed-based debug probe.

1.2.1 Command line use

Should you wish to use the command-line tools, a command file *MCUXpressoPath.cmd* is provided to set up the path for the local command window. This file is located within the installation directory.

Note: the low-level components within MCUXpresso IDE are located inside plugins installed within the IDE's directory structure. Updates to MCUXpresso IDE may install new plugins but these are correctly located by this script. Since this file makes relative path assumptions it can only be used from its location within the IDE's installation directory.

To use this file from within a terminal session or from within another script, the file must be sourced, for example (the text details vary depending on IDE version number):

```
C:\Windows\System32>C:\nxp\MCUXpressoIDE_11.9.0_2144\MCUXpressoPath.cmd
Configuring command line environment for MCUXpresso IDE installed at C:\nxp
\MCUXpressoIDE_11.9.0_2144
C:\nxp\MCUXpressoIDE_11.9.0_2144\ide\plugins
\com.nxp.mcuxpresso.tools.win32_11.9.0.202307121226\tools
\bin;C:\nxp\MCUXpressoIDE_11.9.0_2144\ide\plugins
\com.nxp.mcuxpresso.tools.bin.win32_11.9.0.202312111712\binaries;C:\nxp
\MCUXpressoIDE_11.9.0_2144\ide\plugins
\com.nxp.mcuxpresso.tools.win32_11.9.0.202307121226\buildtools\bin;...

C:\Windows\System32>arm-none-eabi-gcc --version

arm-none-eabi-gcc (Arm GNU Toolchain 12.3.Rel1 (Build arm-12.35)) 12.3.1
 20230626
Copyright (C) 2022 Free Software Foundation, Inc.
...
```

Additionally supplied is a Bash version of this script. Should this be used it must be sourced within a user's command file in order for the paths to be updated correctly.

1.3 macOS

- macOS supported versions
  - Version 13: "Ventura"
  - Version 14: "Sonoma"

The MCUXpresso IDE installer is supplied as a macOS *.pkg* installer file. Double-click on the installer to install MCUXpresso IDE into a subfolder of your Applications folder.

To start MCUXpresso IDE, use the macOS Launchpad. Alternatively, click the **Open MCUXpresso IDE** icon in the */Applications/MCUXpressoIDE_version* folder or run **MCUXpresso IDE.app**, which you can find in the *MCUXpresso IDE* subfolder of the main MCUXpresso IDE installation directory within */Applications*.

1.3.1 Command line use

Should you wish to use the command-line tools, a bash script file *MCUXpressoPath.sh* is provided to set up the path for the local shell. This file is located within the installation directory.

Note: the low-level components within MCUXpresso IDE are located inside plugins installed within the IDE's directory structure. Updates to MCUXpresso IDE may install new plugins but these are correctly located by this script. Since this file makes relative path assumptions it can only be used from its location within the IDE's installation directory.

To use this file from within a terminal session or from within another script, the file must be sourced, for example (the text details vary depending on IDE version number):

```
~$ source /Applications/MCUXpressoIDE/MCUXpressoPath.sh

Set PATH to /Applications/MCUXpressoIDE_11.9.0_2144/ide/
plugins/com.nxp.mcuxpresso.tools.bin.macosx_11.9.0.202312111712/
binaries:/Applications/MCUXpressoIDE_11.9.0_2144/ide/plugins/
com.nxp.mcuxpresso.tools.macosx_11.9.0.202307121226/tools/bin:/...

~$ arm-none-eabi-gcc --version

arm-none-eabi-gcc (Arm GNU Toolchain 12.3.Rel1 (Build arm-12.35)) 12.3.1
 20230626
Copyright (C) 2022 Free Software Foundation, Inc.
```

1.4 Linux

- Linux - Ubuntu 20.04 LTS and 22.04 LTS
  - Only 64-bit versions of Linux are supported.

MCUXpresso IDE for Linux is a 64-bit application, so it does not run on 32-bit systems. It is supported and tested only on the Linux distributions mentioned above.

The installer is supplied as an executable that installs the MCUXpresso IDE components. The installer requires root privileges, although, once it is installed, no special privileges are required to run the MCUXpresso IDE. The installer requests a superuser password when it is started. Once installation has been completed, we strongly recommend that you restart your system - if you do not do this, then some areas of the tools may not function correctly.

Block: Untagged paragraph For further details, see Appendix A - Linux installation.

Note: Some packages (libpython3.8, libncurses5, and libncursesw5) are not declared as hard dependencies. The MCUXpresso IDE installation is successful even if the libraries are not available on the system. However, these components are mandatory, and the installer has a mechanism to install them. In the latest releases of Ubuntu, the libraries are no longer available in the main package sources. As a result, the installer may complete its execution with a note indicating the missing packages. If these libraries are not installed, the Arm GNU Toolchain does not work as expected. To install them, you can either update your apt sources with a source from an older Ubuntu version or download and install them manually.

1.4.1 Command line use

Should you wish to use the command-line tools, a bash script file *MCUXpressoPath.sh* is provided to set up the path for the local shell. This file is located within the installation directory.

Note: the low-level components within MCUXpresso IDE are located inside plugins installed within the IDE’s directory structure. Updates to MCUXpresso IDE may install new plugins but these are correctly located by this script. Since this file makes relative path assumptions it can only be used from its location within the IDE’s installation directory.

To use this file from within a terminal session or from within another script, the file must be sourced, for example (the text details vary depending on IDE version number):

```
~$ . '/usr/local/mcuxpressoide-11.9.0_2144/MCUXpressoPath.sh'
Set PATH to /usr/local/mcuxpressoide-11.9.0_2144/ide/plugins/
com.nxp.mcuxpresso.tools.bin.linux_11.9.0.202312111712/
binaries:/usr/local/mcuxpressoide-11.9.0_2144/ide/plugins/
com.nxp.mcuxpresso.tools.linux_11.9.0.202307121226/tools/bin:/
```

1.4.2 Other Linux distro

Due to the huge variation in capabilities of different Linux distributions and versions, MCUXpresso IDE may work on other distributions/versions but we cannot provide support if it does not.

In such circumstances, the MCUXpresso IDE forum is a good place to search for information or to post questions, as other users may be able to assist you.

1.5 Installation notes

1.5.1 High-resolution displays

When using high-resolution displays, high-dpi icons can be selected by adding an extra argument -*Dswt.autoScale=200* to the end of the mcuxpressoide.ini file.

This can be found, for example, at:

```
Windows: C:\nxp\<install_dir>\ide\mcuxpressoide.ini
macOS: /Applications/<install_dir>/ide/MCUXpressoIDE.app/Contents/MacOS/
mcuxpressoide.ini
Linux: /usr/local/<install_dir>/ide/mcuxpressoide.ini
```

Note: This may cause the startup splash screen to be cropped, but it does not affect the usability of the product.

1.5.2 Running under virtual machines

It is possible to install the MCUXpresso IDE within a virtual machine (VM) environment. Generally, such installations cause few issues. Due to the nature of VMs, the most likely problems relate to the sharing of resources (USB, memory).

In the unlikely event that you experience issues, we welcome reports, but due to the nature of VM operation we can offer no guarantee of resolution.

1.5.3 Help us improve MCUXpresso IDE

MCUXpresso IDE can send anonymous information to NXP on how you use the IDE, including the built-in Config Tools, and with which MCUs. This information can help us to improve the functionality of the tools and to resolve problems. You can turn off this information collection at any time by unticking the workspace option:

Win/Linux **Window -> Preferences -> MCUXpresso IDE -> General -> Help us improve the tool**

macOS **MCUXpresso IDE -> Preferences -> MCUXpresso IDE -> General -> Help us improve the tool**

2 Migrating from an earlier version of MCUXpresso IDE

MCUXpresso IDE functions and features are under continuous development, it is recommended for users to read both the ReadMe and KnownIssues files inside the product installation directory.

We would generally recommend the following flow when a new IDE release becomes available...

Install the new release in parallel with the original version. This allows you to evaluate the new release before committing to using it for your main product development work.

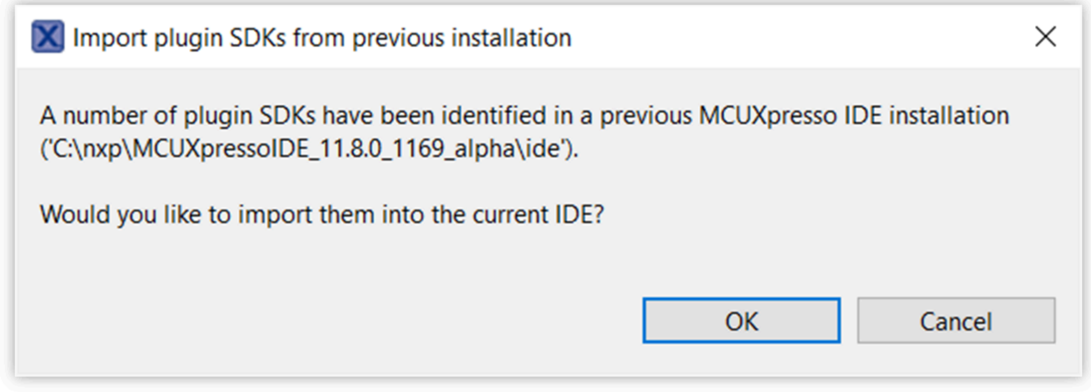
When using the new version, use a new workspace - to keep your new "evaluation" world separate from your old "development" world. You can easily copy your projects from your existing workspace to your new workspace - for example by checking them out of version control again, or simply using the IDE's Quickstart Panel option to "Import project(s) from file system..." and pointing at the root directory of your existing workspace.

Note: MCUXpresso IDE v11.2.x (or later) projects are not backward-compatible with earlier versions of MCUXpresso IDE. Also, if you edit an existing project with MCUXpresso IDE v11.5.x, it may no longer be usable with an earlier version of MCUXpresso IDE.

You can also import preferences from your existing workspace to your new workspace. To do this run your old IDE installation, and use **File -> Export -> General -> Preferences** to export. Then in your new IDE installation use **File -> Import -> General -> Preferences** to pull your preferences in.

If you have installed additional plugins into your original IDE installation, there are two ways to import them into the newly installed IDE:

1. Manually, using the dedicated Eclipse import wizard. From your new installation, go to **File -> Import -> Install -> From Existing Installation** and point at the *ide* directory within the installation directory of your original IDE.
2. Automatically, upon first launching the IDE. This only applies to plugin SDKs and in order to import them, you have to click **OK** when the following dialog appears:



Import plugin SDKs from previous installation

A number of plugin SDKs have been identified in a previous MCUXpresso IDE installation ('C:\nxp\MCUXpressoIDE_11.8.0_1169_alpha\ide').

Would you like to import them into the current IDE?

Figure 1. Import plugin SDKs from previous installation

Note: To enable MCUXpresso IDE to automatically detect a previous installation, you must install the newer version IDE in the same directory as the older one.

Tip: *MCUXpresso IDE version 10.2.0 (or later) allows projects to be imported by simply dragging a project folder (or zip archive of projects) directly into the Project Explorer view of the IDE. In addition, it is possible to drag from the Project Explorer view of an older IDE directly into the Project Explorer view of MCUXpresso IDE version 10.2.0. This provides a simple way of transferring projects into the new IDE. However, it is recommended that you delete build configuration and launch configuration folders before (or after) copying.*

Note: Due to enhancements from MCUXpresso IDE version 10.2, older launch configurations are no longer compatible with this version, failure to delete them leads to a warning and the launch configuration then gets deleted automatically on the next debug attempt.

3 Appendix A - Linux installation

3.1 Ubuntu

The product is distributed as a file called `mcuxpressoide-<build>.x86_64.deb.bin`, which is a binary file that when run, it creates a Debian package and installs it.

To install this file, you must make it executable and then run it as root. For example, if the file is in the current working directory:

```
chmod +x mcuxpressoide-<build>.x86_64.deb.bin
sudo ./mcuxpressoide-<build>.x86_64.deb.bin
```

Once you have agreed to the license terms (use the keyboard arrow keys) the Debian package gets installed along with any packages that it requires.

3.1.1 Creating a backup

To create a backup of an older version during installation, use `-b` or `--backup`. You have to pass it to the underlying script of the `.run` package by calling:

```
<install_package>.deb.bin -- -b or
<install_package>.deb.bin -- --backup
```

3.2 Other Linux distributions

Other distributions are not supported or tested. The Debian package lists other package names as dependencies, and these may not be among the packages provided by all distributions. Nonetheless, it may run on other Linux distributions.

- For distributions based on Debian (with a Debian-based package manager), try the `.x86_64.deb.bin` installation image.

3.3 Running the MCUXpresso IDE

3.3.1 From the desktop

To run from the desktop, search for a program containing "MCUXpresso" in its name and run it as normal for your desktop. You can usually find it in the "Development" application category. (This should work in most Linux desktop environments.)

Note: The installation script now creates a soft link at `/usr/local/mcuxpressoide` pointing to the real installation directory.

3.3.2 From bash

The product is installed in the directory `/usr/local/mcuxpressoide-<build>` and you can run it using the command `mcuxpressoide` if `/usr/local/mcuxpressoide-<build>/ide` is placed on your path, for example using:

```
export PATH="/usr/local/mcuxpressoide-<build>/ide:$PATH"
```

Depending on the desktop manager you use you may have to set some environment variables. It is safe to use these settings for any desktop, however, and you can always run using the following command line.

```
SWT_GTK3=0 UBUNTU_MENUPROXY=0 mcuxpressoide &
```

3.3.3 Further information

- `SWT_GTK3` controls the use of your distribution's GTK libraries that are used in Gnome-based desktops (including Ubuntu Unity and Gnome Desktop). The setting above stops GTK3 from being used in the IDE. The version of Eclipse underlying MCUXpresso IDE shows small errors that make normal working impossible in these desktops unless you use this setting.
- `UBUNTU_MENUPROXY` controls the way in which the menu bar of an application can appear at the top of the screen even when an application is not used in full-screen mode. Some users have reported issues in Eclipse in some window managers when the setting above is not used (although we have not observed them ourselves).

3.3.4 Known issues

On (at least) Ubuntu 16.10 the names of the boards underneath their photos do not appear.

Note: The user intending to run MCUXpresso IDE must have write permission on the install directory.

4 Appendix B - Migrating from LPCXpresso IDE version 8.2.x - Hints and tips

4.1 Introduction

MCUXpresso IDE incorporates core technology from LPCXpresso IDE 8.2.2.

Migrating code from LPCXpresso IDE to MCUXpresso IDE should be straightforward, though you should always browse the release notes, the supplied documentation and the online FAQ material.

Below are some hints and suggestions of things that you should do or consider when migrating.

4.1.1 Parallel installations

You can install a new version of the MCUXpresso IDE in parallel with existing installations and also in parallel with LPCXpresso IDE. This allows you to try a newly released version alongside a version currently installed.

Furthermore, there is no need to take any special care with licenses (activation codes), since any installed code is automatically picked up by the new MCUXpresso IDE installation.

4.1.2 Installing Eclipse plugins

If you install a new version of MCUXpresso IDE on macOS or Linux, the first time you run the new product, it offers you the opportunity to reinstall previously used plugins (for example, those for version control). However, this does not happen on Windows, and manually installing your favorite plugins may take considerable time to complete.

An alternative approach is to import the plugins from an earlier LPCXpresso IDE installation. To do this, follow:

```
File->Import->Install->From Existing Installation
```

Then browse to the `lpcxpresso` directory within an existing LPCXpresso IDE application's installation.

4.1.3 Managing workspaces

While a new MCUXpresso IDE version can open workspaces created by an earlier release, a workspace (and the projects it contains) that have been used by a new MCUXpresso IDE version may not correctly load into an earlier version. Therefore, we would strongly recommend that you back up your projects before commencing any migration.

The simplest way to do this is to create a new workspace in the new MCUXpresso IDE version, and then import any projects into this new workspace. The FAQ details how to import projects into a new workspace.

https://community.nxp.com/message/630625

Alternatively, if you have your projects checked into a version control system (for example, using Subversion and the Subclipe Eclipse plugin), then you can simply check out your projects into the new workspace.

Also ensure that you do a full, clean build after switching to the new version.

4.1.4 Launch configurations

Sometimes the contents of, or options specified in, the debug launch configurations used by MCUXpresso IDE can change between versions. Therefore, when moving to a new version of the MCUXpresso IDE, we would recommend deleting any debug launch configurations within your project that were created by an earlier version. These files are typically named.

```
<projectname> Debug.launch and <projectname> Release.launch.
```

The easiest way to do this is to right-click on the project in Project Explorer and select **Launch Configurations -> Delete Launch Configurations**. The IDE then automatically creates a fresh set of launch configurations the next time you start a debug session. You may have to reapply any modifications you made to your launch configurations in your previous version of MCUXpresso IDE.

For more information on launch configurations, see the FAQ Launch Configuration Menu at

https://community.nxp.com/message/630714

4.1.5 Startup code

The startup code generated by MCUXpresso IDE can sometimes be updated between releases, often to support new tool features. We would therefore strongly recommend that you consider updating your startup code to match the latest generated by the project wizard for the part that you are using.

4.1.6 Linker scripting

In LPCXpresso IDE V7.9.0 and later, the linker script template mechanism was overhauled to provide a much more flexible and powerful means for the user to change the content of the linker script generated by the managed linker script mechanism.

If you are moving a project that uses a modified linker script from a version of LPCXpresso IDE prior to version 7.9.0, then read the detailed FAQ on Freemarker Linker Script Templates at

https://community.nxp.com/message/630611

4.1.7 Compiler symbols

LPCXpresso IDE projects generally would define the compiler symbol **__CODE_RED**. This could then be used in source code to determine if the LPCXpresso IDE was being used to build the code, and conditionally compile sections of code in (or out) of the image being built.

When building under MCUXpresso IDE, the **__CODE_RED** symbol does not get removed from existing LPCXpresso IDE generated projects (for instance LPCOpen examples), either already in your workspace or in projects that you import into a new workspace. Also, if you create new projects for the preinstalled (LPC) MCUs then again, the preinstalled MCU's new project wizards sets up the symbol.

However, if you create projects for SDK installed MCUs, then the symbol **__CODE_RED** is not set up for the compiler; the symbol **__MCUXPRESSO** is defined instead.

Therefore, if you are porting existing code from LPCXpresso IDE into a new project created for an SDK installed MCU, then you have to check whether it is appropriate to change any instances of **__CODE_RED** to **__MCUXPRESSO**.

4.1.8 SPIFI Flash drivers for LPC18xx and LPC43xx

Legacy SPIFI flash drivers, for example the `LPC18_43_SPIFI_1MB_64KB.cfx` or `LPC18_43_S25FL032P.cfx`, and so on, have been removed from MCUXpresso IDE. In the last few releases of LPCXpresso IDE these drivers were in fact just copies of the `LPC18_43_SPIFI_GENERIC.cfx` driver, and were included to maintain compatibility with certain older prebuilt examples.

If you import a project for the LPC18xx or LPC43xx and experience an error because the SPIFI flash driver is not present, simply edit the project memory configuration and replace the missing driver with the `LPC18_43_SPIFI_GENERIC.cfx` driver.

4.1.9 License compatibility with LPCXpresso IDE

MCUXpresso IDE requires no activation procedure and uses no licenses. Free or Pro Edition license from an LPCXpresso IDE install have no impact on an MCUXpresso IDE installation.

5 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

6 Revision history

Table 1. Revision history

Document ID	Release date	Description
UG10060 v.3	1 July 2024	11.10.0 - major release version update. See chapter 2 from *MCUXpresso IDE User Guide* for details.
UG10060 v.2	17 January 2024	11.9.0 - major release version update. See chapter 2 from *MCUXpresso IDE User Guide* for details.
UG10060 v.1	31 July 2023	11.8.0 - major release version update. See chapter 2 from *MCUXpresso IDE User Guide* for details.

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.