MCUXSDKMC56F83000GSUG

Getting Started with MCUXpresso SDK for MC56F83000-EVK

Rev. 2 — 1 April 2022 User Guide

Contents

1 Overview

The MCUXpresso Software Development Kit (SDK) provides comprehensive software support for Kinetis, LPC, and DSC Microcontrollers. The MCUXpresso SDK includes a flexible set of peripheral drivers designed to speed up and simplify development of embedded applications. Along with the peripheral drivers, the MCUXpresso SDK provides an extensive and rich set of example applications covering everything from basic peripheral use case examples to full demo applications. The MCUXpresso SDK contains various middleware to support rapid development.

For supported toolchain versions, see *MCUXpresso SDK Release Notes for MC56F83000-EVK* (document MCUXSDKMC56F83000RN).

For more details about MCUXpresso SDK, see MCUXpresso-SDK.

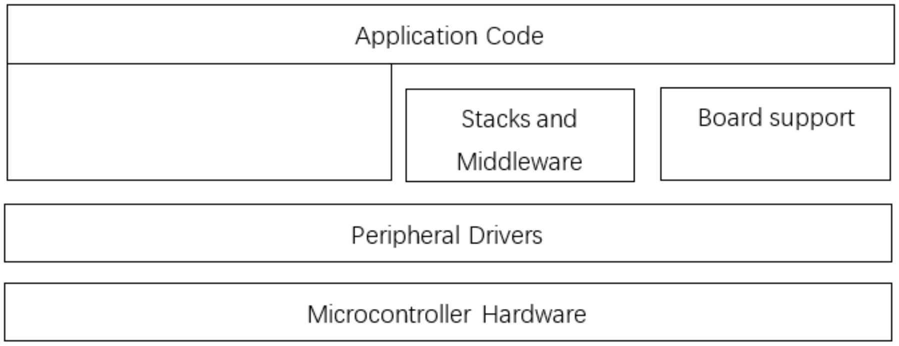


Figure 1. MCUXpresso SDK layers

2 Run a demo application using CodeWarrior

This section describes the steps required to build, run, and debug example applications provided in the MCUXpresso SDK.

NOTE

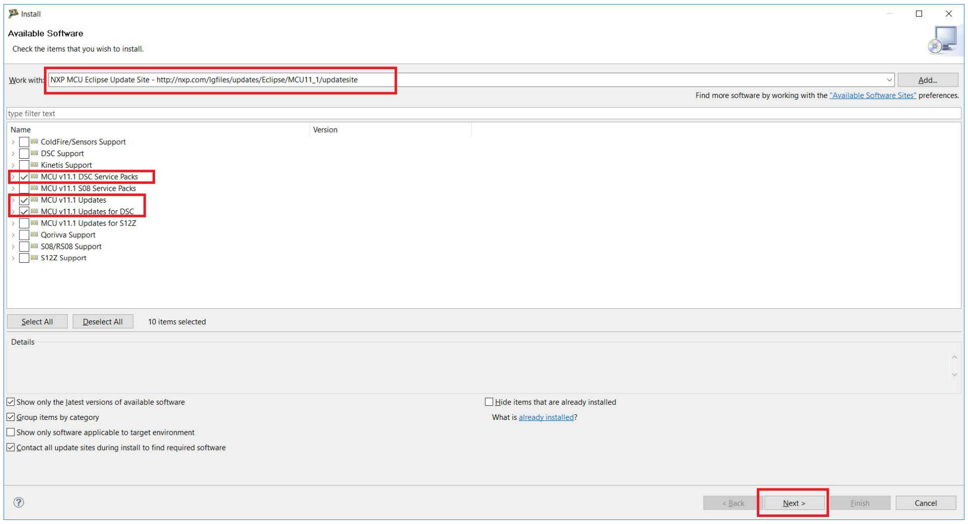
CodeWarrior IDE version 11.1 with update4 is used as an example to show below steps, and the DSC toolchain should correspond to the latest supported version, as described in *MCUXpresso SDK Release Notes for MC56F83000-EVK* (document MCUXSDKMC56F83000RN).

Download CodeWarrior 11.1 for DSC from CodeWarrior® for MCUs.

Two option to install update4:

1. **Online**

   Open Codewarrior, click **help -> Install New Software**. The settings are as shown in Figure 2. Click **Next** to install.



Install

Available Software

Check the items that you wish to install.

Work with: NXP MCU Eclipse Update Site - http://nxp.com/lgfiles/updates/Eclipse/MCU11_1/updatesite

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
□ ColdFire/Sensors Support	
□ DSC Support	
□ Kinetis Support	
☑ MCU v11.1 DSC Service Packs	
□ MCU v11.1 S08 Service Packs	
☑ MCU v11.1 Updates	
☑ MCU v11.1 Updates for DSC	
□ MCU v11.1 Updates for S12Z	
□ Qorivva Support	
□ S08/RS08 Support	
□ S12Z Support	

10 items selected

Details

☑ Show only the latest versions of available software

☑ Group items by category

□ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

□ Hide items that are already installed

What is already installed?

Figure 2. Update settings

2. **Offline**

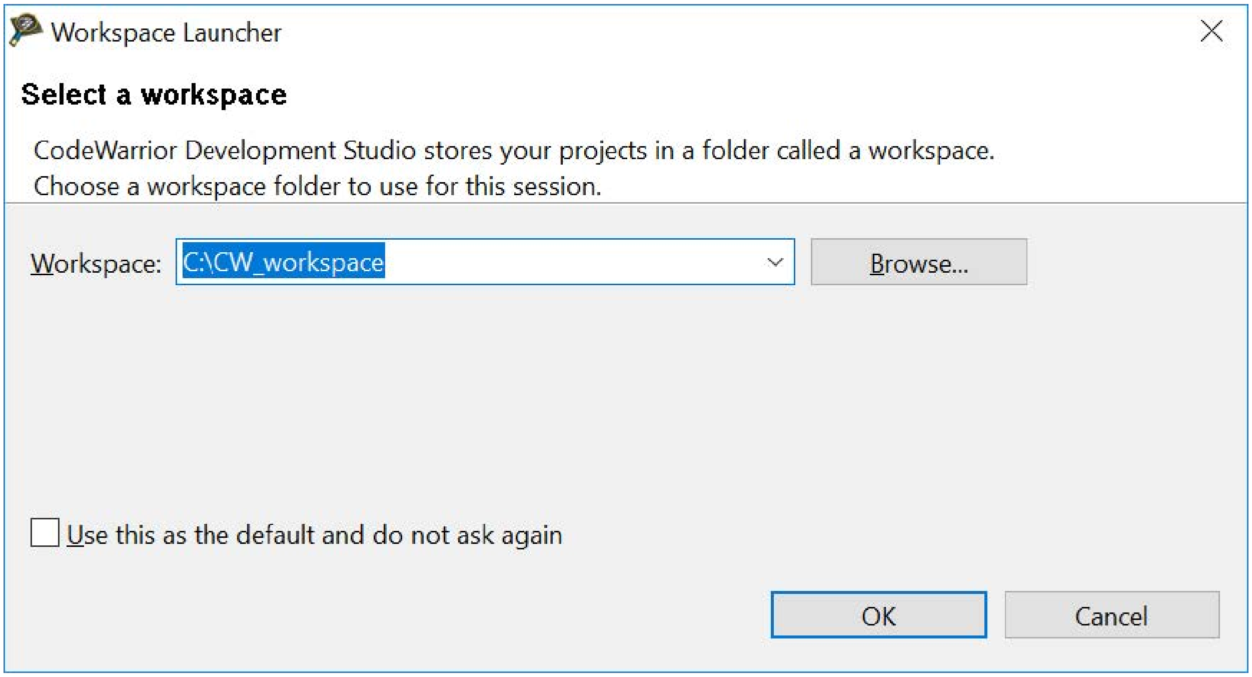
   Download (via above link) CodeWarrior for MCU 11.1 Update4.

   For details about how to download and install a package, see an example from https://www.nxp.com/webapp/Download?colCode=INSTALL-CODEWARRIOR-DSC.

2.1 Build an example application

To build the `hello_world` example application, perform the following example steps:

1. Launch CodeWarrior and in the workspace launcher, choose a workspace which holds the projects to use. If the dialogue box does not pop up, enter a workspace folder and create one workspace.



Workspace Launcher

Select a workspace

CodeWarrior Development Studio stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: C:\CW_workspace

□ Use this as the default and do not ask again

Figure 3. Workspace launcher view

Then the CodeWarrior Development Studio workspace with empty projects appears.

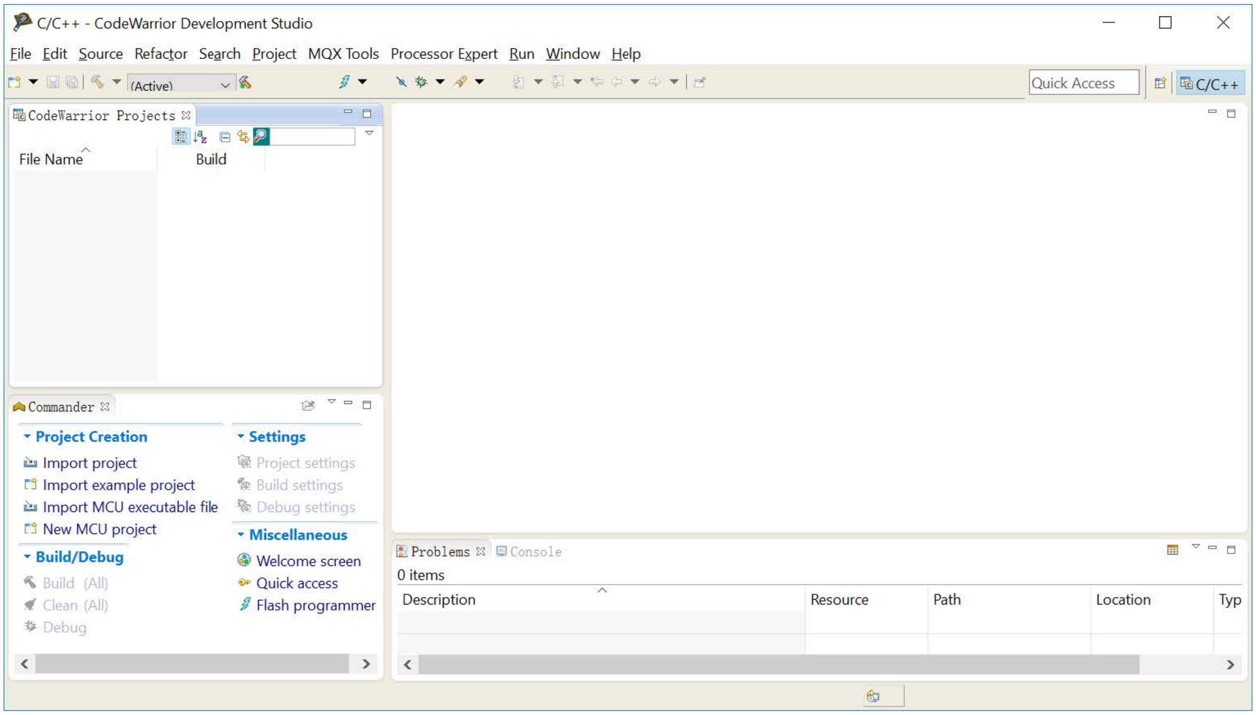
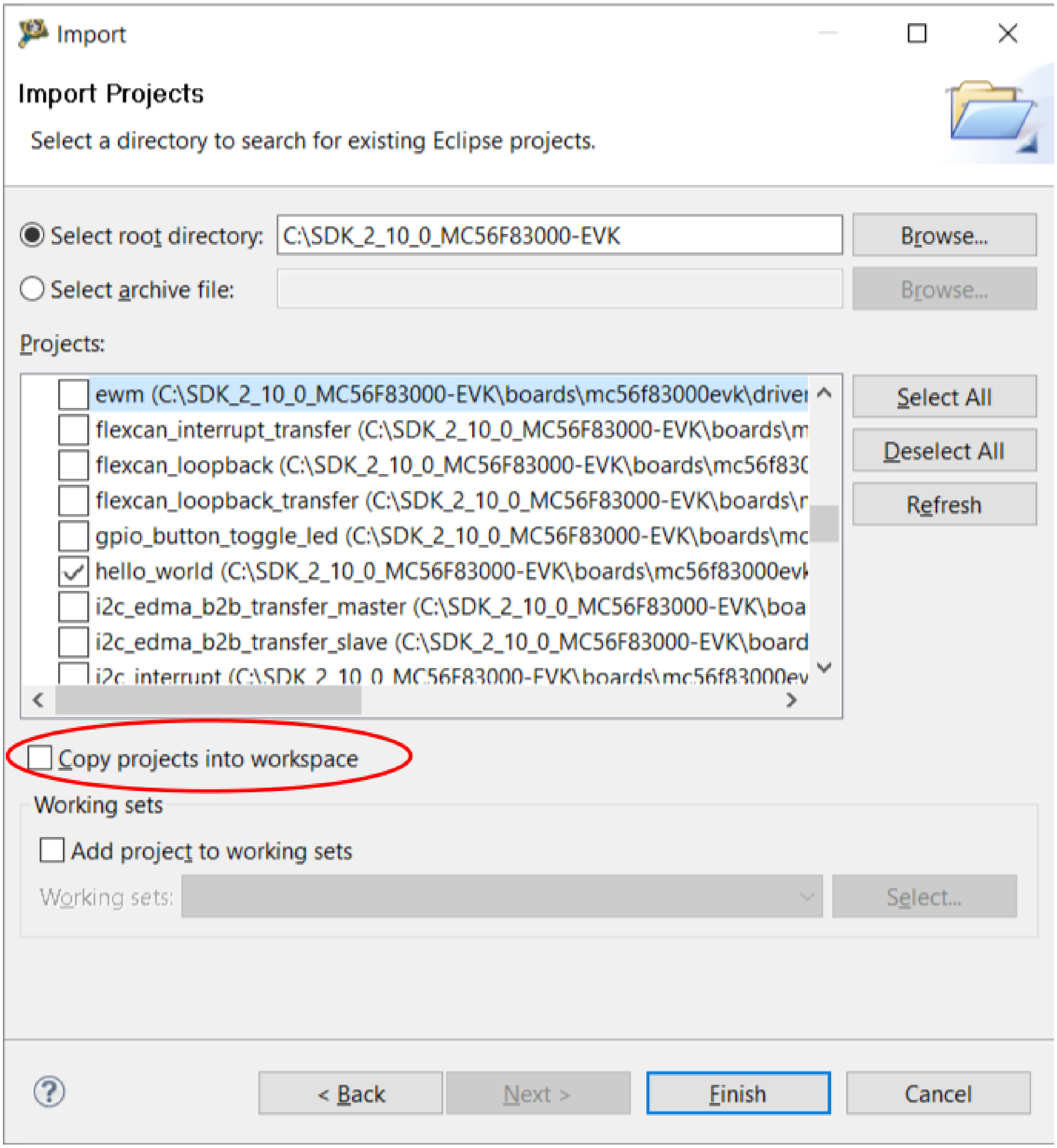


Figure 4. CodeWarrior development studio view

2. Import the project into the workspace.

   Click **Import project** in the **Commander** pane. A form pops up. Click **Browse** to the SDK install directory.



Import

Import Projects

Select a directory to search for existing Eclipse projects.

● Select root directory: C:\SDK_2_10_0_MC56F83000-EVK

○ Select archive file:

Projects:

- □ ewm (C:\SDK_2_10_0_MC56F83000-EVK\boards\mc56f83000evk\driver
- □ flexcan_interrupt_transfer (C:\SDK_2_10_0_MC56F83000-EVK\boards\m
- □ flexcan_loopback (C:\SDK_2_10_0_MC56F83000-EVK\boards\mc56f830
- □ flexcan_loopback_transfer (C:\SDK_2_10_0_MC56F83000-EVK\boards\r
- □ gpio_button_toggle_led (C:\SDK_2_10_0_MC56F83000-EVK\boards\mc
- ☑ hello_world (C:\SDK_2_10_0_MC56F83000-EVK\boards\mc56f83000evk
- □ i2c_edma_b2b_transfer_master (C:\SDK_2_10_0_MC56F83000-EVK\boa
- □ i2c_edma_b2b_transfer_slave (C:\SDK_2_10_0_MC56F83000-EVK\board
- □ i2c_interrupt (C:\SDK_2_10_0_MC56F83000-EVK\boards\mc56f83000ev

□ Copy projects into workspace

Working sets

□ Add project to working sets

Working sets:

Figure 5. Import projects view

Then all available demo projects are shown as Figure 5. Select the `hello_world` project in the list and click **Finish**.

If you already know the project location, navigate to the folder when clicking **Browse**, and only one project can be seen.

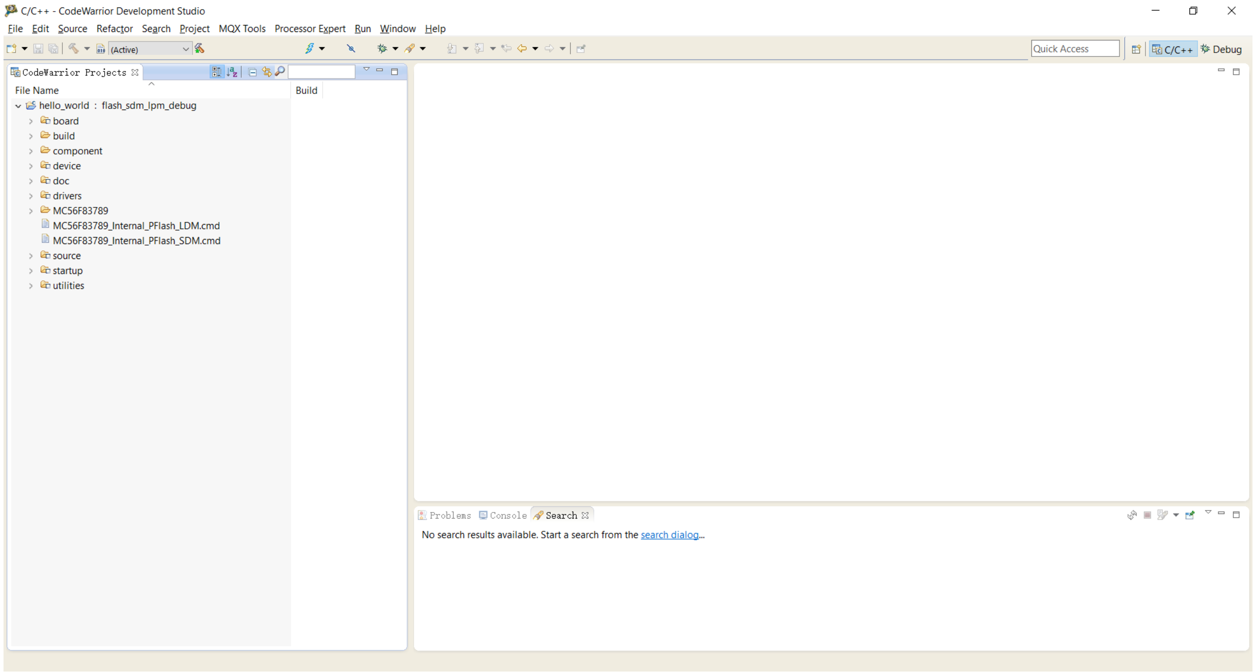
To locate most example application workspace files, use the following path:

```
<install_dir>/boards/<board_name>/<example_type>/<application_name>/codewarrior
```

Using the MC56F83000-EVK hardware platform as an example, the `hello_world` workspace is located in:

```
<install_dir>/boards/evkmc56f83000/demo_apps/hello_world/codewarrior
```

3. Select the desired build target from the drop-down menu. For this example, select **hello_world – flash_sdm_lpm_debug**, as shown in Figure 6.


C/C++ - CodeWarrior Development Studio
File Edit Source Refactor Search Project MQX Tools Processor Expert Run Window Help
(Active)
Quick Access
C/C++
Debug
CodeWarrior Projects
File Name
Build
hello_world : flash_sdm_lpm_debug
board
build
component
device
doc
drivers
MC56F83789
MC56F83789_Internal_PFlash_LDM.cmd
MC56F83789_Internal_PFlash_SDM.cmd
source
startup
utilities
Problems
Console
Search

No search results available. Start a search from the search dialog...

Figure 6. Demo build target selection

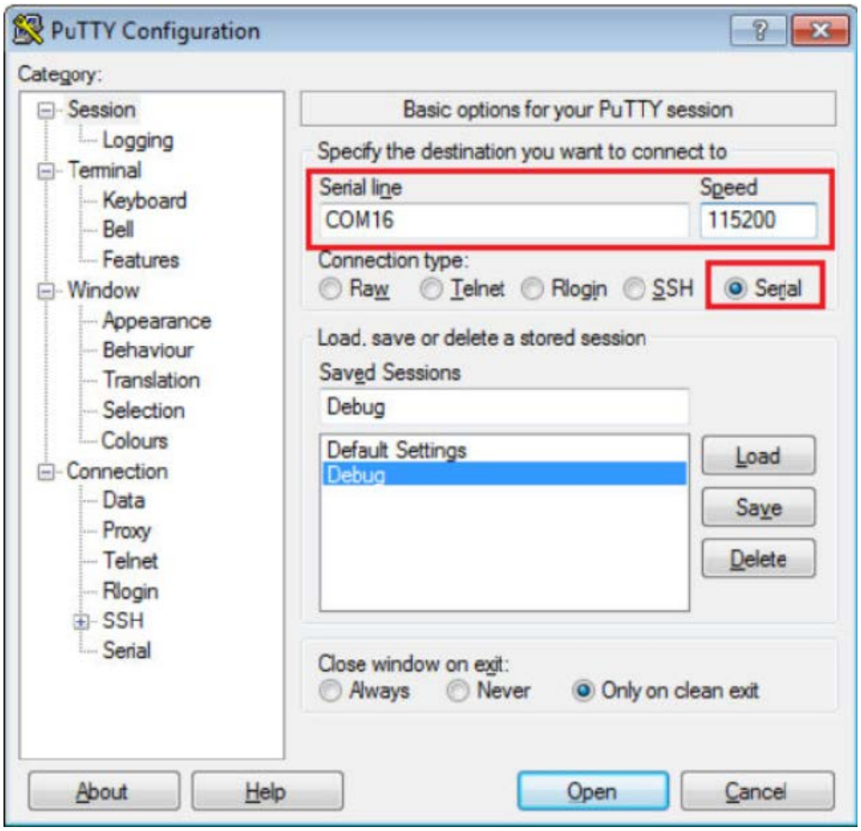
4. To build the demo application, click **Build (All)** in the **Commander** pane.
5. The build completes without errors.

2.2 Run an example application

To download and run the application, perform the following steps:

1. Connect the MC56F83000-EVK JM60 USB port, **J8**, to your PC via USB cable within the board package. This USB port is used for on board OSJTAG debugger and USB to UART bridge.
2. Install the OSJTAG driver and USB CDC driver as PC hint if it is the first time you run it on your PC. The OSJTAG and USB CDC driver are provided by Codewarrior by default.

   The Codewarrior may prompt to update the JM60 firmware, which requires to connect the **J6** with a jumper on board and then follow the instruction by Codewarrior to finish the firmware update. The default debug interface is OSJTAG for MC56F83000-EVK board.
3. Open the terminal application on the PC, such as PuTTY or TeraTerm, and connect to the debug COM port (to determine the COM port number, see How to determine COM port). Configure the terminal with these settings:
   a. 115200, defined by `BOARD_DEBUG_UART_BAUDRATE` in the *board.h* file
   b. No parity
   c. 8 data bits
   d. 1 stop bit



Basic options for your PuTTY session

Specify the destination you want to connect to

Serial line: COM16

Speed: 115200

Connection type: ○ Raw ○ Telnet ○ Rlogin ○ SSH ● Serial

Load, save or delete a stored session

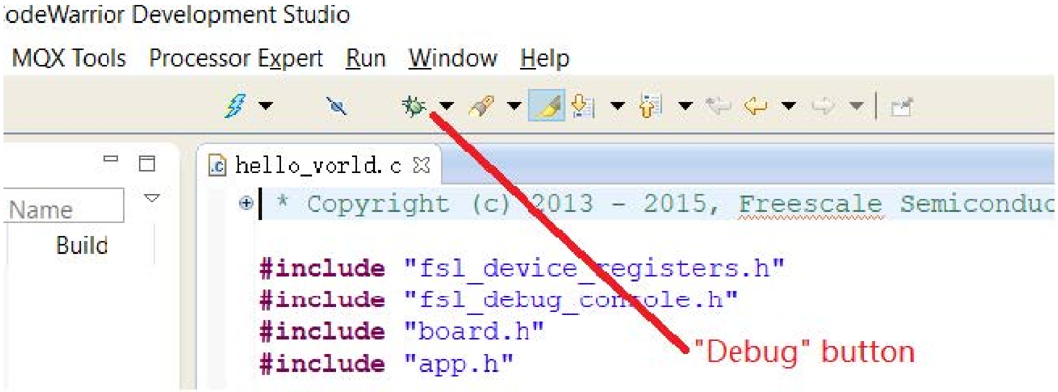
Saved Sessions: Debug

Default Settings
Debug

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 7. Terminal (PuTTY) configuration

4. For this example, click **Debug** in the **Commander** pane, and select the `hello_world_flash_sdm_lpm_debug_OSJTAG` launch configuration.

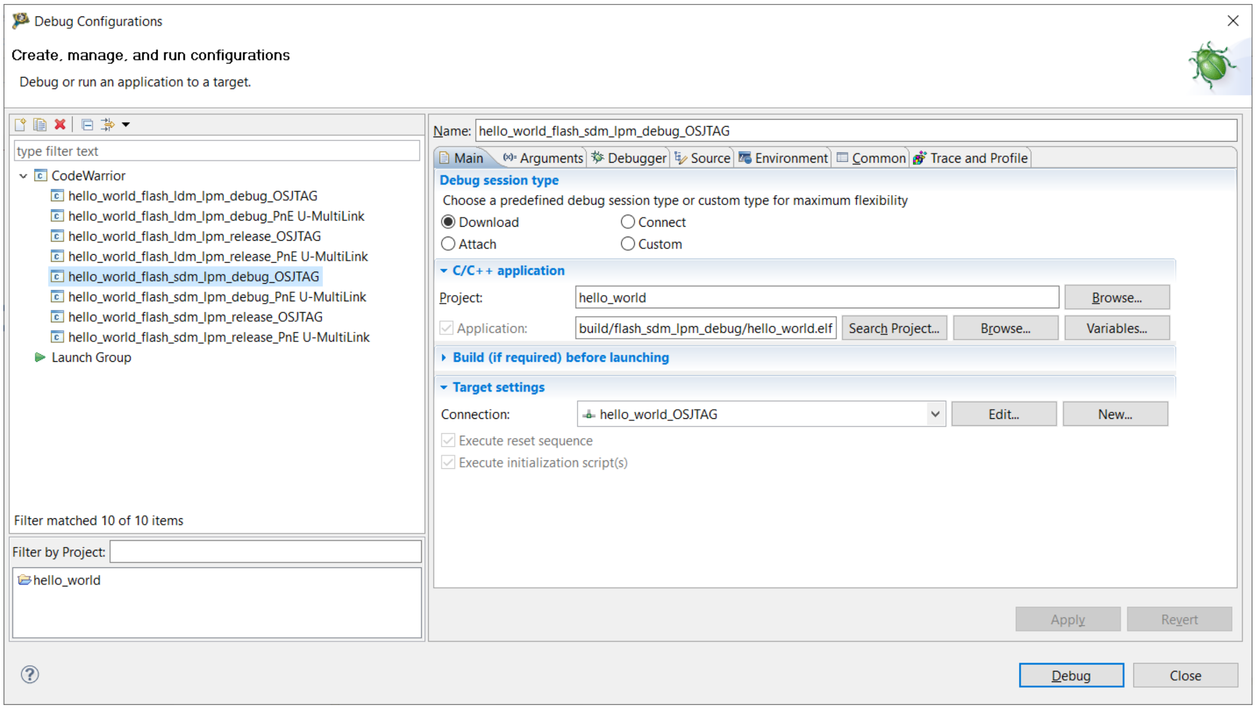


```
 * Copyright (c) 2013 - 2015, Freescale Semiconduc

#include "fsl_device_registers.h"
#include "fsl_debug_console.h"
#include "board.h"
#include "app.h"
```

"Debug" button

Figure 8. Debug button



Debug Configurations

Create, manage, and run configurations

Debug or run an application to a target.

type filter text

- CodeWarrior
  - hello_world_flash_ldm_lpm_debug_OSJTAG
  - hello_world_flash_ldm_lpm_debug_PnE U-MultiLink
  - hello_world_flash_ldm_lpm_release_OSJTAG
  - hello_world_flash_ldm_lpm_release_PnE U-MultiLink
  - hello_world_flash_sdm_lpm_debug_OSJTAG
  - hello_world_flash_sdm_lpm_debug_PnE U-MultiLink
  - hello_world_flash_sdm_lpm_release_OSJTAG
  - hello_world_flash_sdm_lpm_release_PnE U-MultiLink
- Launch Group

Filter matched 10 of 10 items

Filter by Project:

hello_world

Name: hello_world_flash_sdm_lpm_debug_OSJTAG

Debug session type

Choose a predefined debug session type or custom type for maximum flexibility

● Download ○ Connect

○ Attach ○ Custom

C/C++ application

Project: hello_world

☑ Application: build/flash_sdm_lpm_debug/hello_world.elf

Build (if required) before launching

Target settings

Connection: hello_world_OSJTAG

☑ Execute reset sequence

☑ Execute initialization script(s)

Figure 9. Debug configuration selection

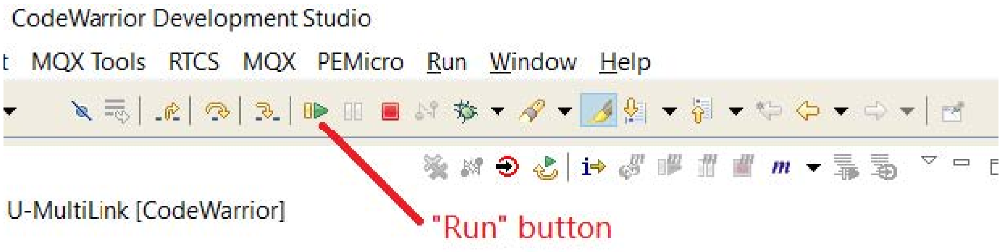
Then the application is downloaded onto target board and automatically runs to the `main()` function.

Click **Run** on the toolbar to run the code.

NOTE

- Generally there are four build configuration for DSC SDK 2.10.x: `flash_sdm_lpm_debug`, `flash_sdm_lpm_release`, `flash_ldm_lpm_debug`, and `flash_ldm_lpm_release`.

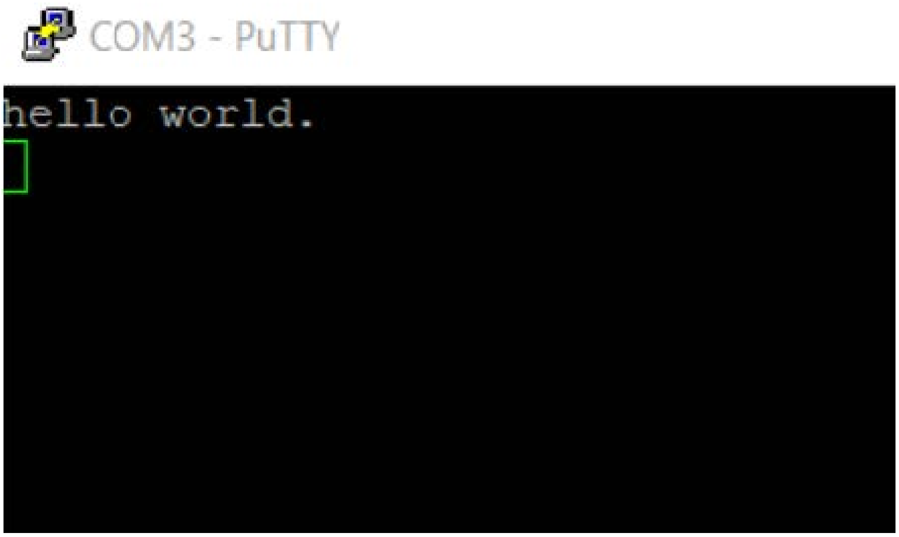
  `debug` uses optimization level 1 and `release` uses optimization level 4. `sdm` means small data memory model. `ldm` means large data memory model. `lpm` means large program memory model.
- Check each demo `readme` document, which includes detailed instructions for HW and SW settings.



CodeWarrior Development Studio
t MQX Tools RTCS MQX PEMicro Run Window Help
U-MultiLink [CodeWarrior]
"Run" button

Figure 10. Run button

5. The `hello_world` application is now running and a banner is displayed on the terminal, as shown in Figure 11. If it does not appear, check your terminal settings and connections.



```
hello world.
```

Figure 11. Text display of the hello_world demo

3 Project template project for a specific DSC part

For device with specific part number, the easiest way to set up customer own project based on MCUXpresso DSC SDK peripheral driver, is the project template. The project template project is supposed to be generated by MCUXpresso Config Tool.

The project template project provides basic MCUXpresso DSC SDK software framework, including startup, linker file, device header file, debug setting, peripheral driver, FreeMASTER, etc.

3.1 MCUXpresso Config Tool support

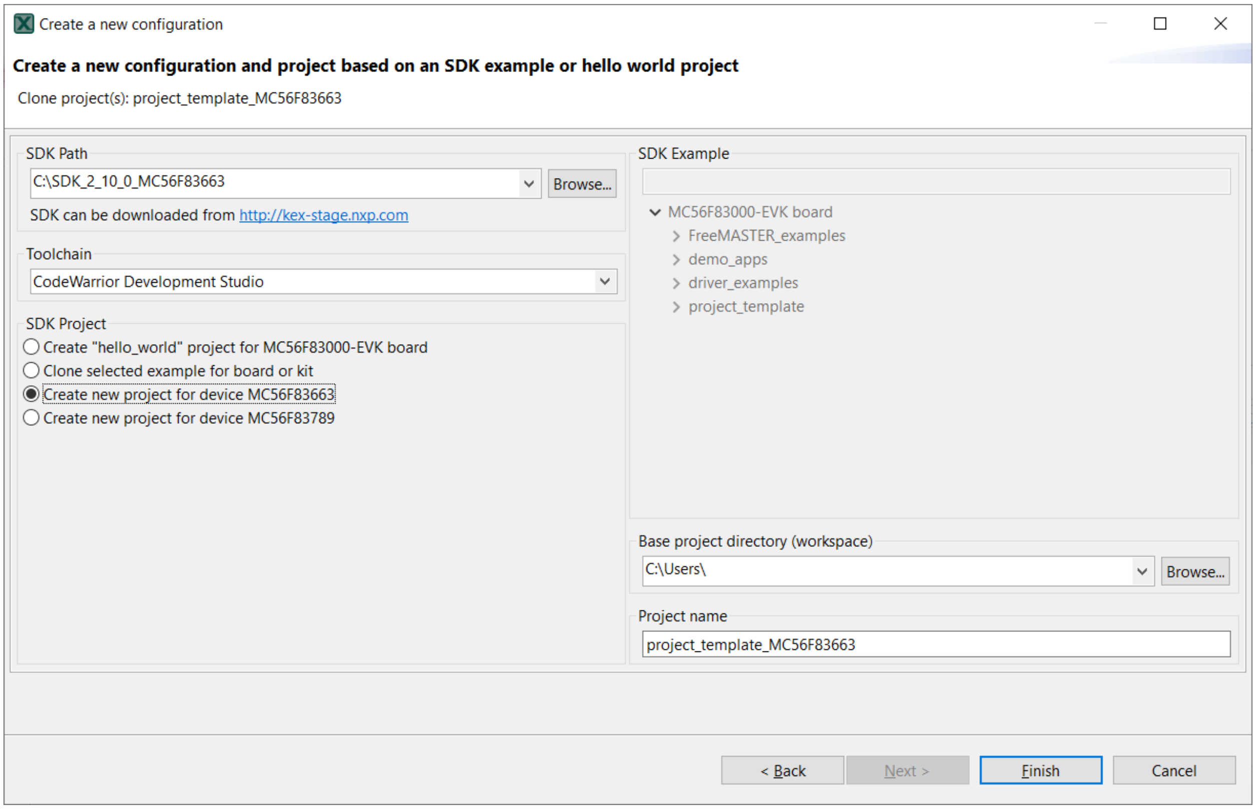
Steps to generate the project template project for specific derivative part number by MCUXpresso Config Tool:

1. Download the specific device SDK package and unzip it.

   NOTE

   Since the project template requires freemaster, select the middleware freemaster in the SDK builder when downloading.

2. Use MCUXpresso Config Tool (v10 or later) to create a project template project, as shown in Figure 12.



Create a new configuration

Create a new configuration and project based on an SDK example or hello world project

Clone project(s): project_template_MC56F83663

SDK Path: C:\SDK_2_10_0_MC56F83663 [Browse...]

SDK can be downloaded from http://kex-stage.nxp.com

Toolchain: CodeWarrior Development Studio

SDK Project

- ○ Create "hello_world" project for MC56F83000-EVK board
- ○ Clone selected example for board or kit
- ● Create new project for device MC56F83663
- ○ Create new project for device MC56F83789

SDK Example

- MC56F83000-EVK board
  - FreeMASTER_examples
  - demo_apps
  - driver_examples
  - project_template

Base project directory (workspace): C:\Users\ [Browse...]

Project name: project_template_MC56F83663

Figure 12. MCUXpress Config Tool support

3. Import the generated template project into CodeWarrior IDE and start the development. By now, a config tool file (extension .mex) with the same project name as shown in Figure 12 is also generated in the generated project folder. This config tool file achieve easy configurations for pins, clocks, and peripherals.

3.2 Project name

The default created project template by Config Tool is `project_template_{part_number}`. For modification, modify the default words in **Project name** textbox, as shown in Figure 12.

3.3 Peripheral driver

All peripheral drivers files are included in the generated project template project. They are same as the peripheral drivers within SDK package.

If some drivers are not used or required, users may delete them in Codewarrior, or delete them in the *project_template_MC56F83xxx/drivers* folder.

4 Revision history

Rev.	Date	Description
2	1 April 2022	Updated for KSDK 2.10.1 NEVIS2 and Anguilla Silver RFP: • Updated Run an example application

Table continues on the next page...

Table continued from the previous page...

Rev.	Date	Description
1	5 January 2022	Updated for KSDK 2.10.0 NEVIS2 and Anguilla Silver RFP: • Updated MCUXpresso Config Tool support • Updated Project name
0	November 2020	Initial release

A How to determine COM port

This section describes the steps necessary to determine the debug COM port number of your NXP hardware development platform.

1. **Linux**: The serial port can be determined by running the following command after the USB Serial is connected to the host:

```
$ dmesg | grep "ttyUSB"
                    [503175.307873] usb 3-12: cp210x converter now attached to ttyUSB0
                    [503175.309372] usb 3-12: cp210x converter now attached to ttyUSB1
```

   There are two ports, one is Cortex-A core debug console and the other is for Cortex M4.

2. **Windows**: To determine the COM port open Device Manager in the Windows operating system. Click on the **Start** menu and type **Device Manager** in the search bar.

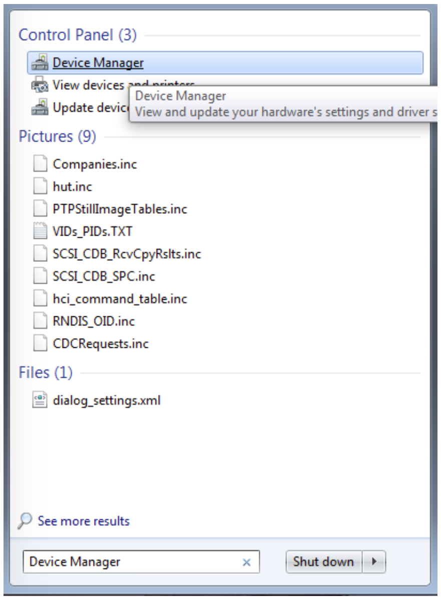


Figure 13. Device Manager

3. In the Device Manager, expand the **Ports (COM & LPT)** section to view the available ports. The COM port names are different for all the NXP boards.

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 http://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 data sheet 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.

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.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μVision, Versatile — are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Airfast — is a trademark of NXP B.V.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

CodeWarrior — is a trademark of NXP B.V.

ColdFire — is a trademark of NXP B.V.

ColdFire+ — is a trademark of NXP B.V.

EdgeLock — is a trademark of NXP B.V.

EdgeScale — is a trademark of NXP B.V.

EdgeVerse — is a trademark of NXP B.V.

eIQ — is a trademark of NXP B.V.

FeliCa — is a trademark of Sony Corporation.

Freescale — is a trademark of NXP B.V.

HITAG — is a trademark of NXP B.V.

ICODE and I-CODE — are trademarks of NXP B.V.

Immersiv3D — is a trademark of NXP B.V.

I2C-bus — logo is a trademark of NXP B.V.

Kinetis — is a trademark of NXP B.V.

Layerscape — is a trademark of NXP B.V.

Mantis — is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

MOBILEGT — is a trademark of NXP B.V.

NTAG — is a trademark of NXP B.V.

Processor Expert — is a trademark of NXP B.V.

QorIQ — is a trademark of NXP B.V.

SafeAssure — is a trademark of NXP B.V.

SafeAssure — logo is a trademark of NXP B.V.

StarCore — is a trademark of NXP B.V.

Synopsys — Portions Copyright © 2021 Synopsys, Inc. Used with permission. All rights reserved.

Tower — is a trademark of NXP B.V.

UCODE — is a trademark of NXP B.V.

VortiQa — is a trademark of NXP B.V.

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

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 1 April 2022
Document identifier: MCUXSDKMC56F83000GSUG