

1 Overview

The MCUXpresso Software Development Kit (SDK) is a collection of software enablement for microcontrollers that includes peripheral drivers, high-level stacks including USB and other middleware packages, such as multicore support and FatFs, and integrated RTOS support for FreeRTOS™ OS. In addition to the base enablement, the MCUXpresso SDK is augmented with demo applications and driver example projects, and API documentation to help the customers quickly leverage the support of the MCUXpresso SDK.

For more details about MCUXpresso SDK, see [MCUXpresso Software Development Kit \(SDK\)](#).

2 MCUXpresso SDK

As part of the MCUXpresso software and tools, MCUXpresso SDK is the evolution of Kinetis SDK, includes support for both LPC and i.MX System-on-Chips (SoC). The same drivers, APIs, and middleware are still available with support for Kinetis, LPC, and i.MX silicon. The MCUXpresso SDK adds support for the MCUXpresso IDE, an Eclipse-based toolchain that works with all MCUXpresso SDKs. Easily import your SDK into the new toolchain to access to all of the available components, examples, and demos for your target silicon. In addition to the MCUXpresso IDE, support for the MCUXpresso Config Tools allows easy cloning of existing SDK examples and demos, allowing users to leverage the existing software examples provided by the SDK for their own projects.

NOTE

In order to maintain compatibility with legacy Freescale code, the filenames and the source code in MCUXpresso SDK containing the legacy Freescale prefix **FSL** has been left as is. The **FSL** prefix has been redefined as the NXP Foundation Software Library. It is suggested to keep the downloaded SDK archive in the root directory of your drive to avoid any unexpected build issues caused by deep path of files.

3 Development tools

The MCUXpresso SDK is compiled and tested with these development tools:

- IAR Embedded Workbench for Arm version 8.50.9
- Makefiles support with GCC ARM Embedded, version 9-2020-q2-update
- MCUXpresso IDE v11.3.0
- MDK-Arm Microcontroller Development Kit (Keil)® 5.33
- Xtensa C Compiler, version 14.05
- Xtensa Xplorer, version 8.0.15

Contents

1	Overview.....	1
2	MCUXpresso SDK.....	1
3	Development tools.....	1
4	Supported development systems....	2
5	Release contents.....	2
6	MCUXpresso SDK release package	3
6.1	Device support.....	3
6.2	Middleware.....	4
7	MISRA compliance.....	6
8	Known issues.....	7
8.1	Maximum file path length in Windows 7® operating system.....	7
8.2	USB issue.....	7
8.3	IAR cannot debug RAM application with J-Link.....	8
8.4	New Project Wizard compile failure	8
8.5	Semihosting not supported on shell-related demo.....	8
8.6	CMSIS PACK new project compile failure.....	9
8.7	Dsp_xaf_demo cannot handle certain audio file types.....	9



4 Supported development systems

This release supports boards and devices listed in [Table 1](#). The boards and devices in bold were tested in this release.

Table 1. Supported MCU devices and development boards

Development boards	MCU devices
MIMXRT595-EVK RevD	MIMXRT595SFFOC , MIMXRT595SFAWCR, MIMXRT555SFFOC, MIMXRT555SFAWCR, MIMXRT533SFFOC, MIMXRT533SFAWCR, MIMXRT595SFFOCR, MIMXRT555SFFOCR, MIMXRT533SFFOCR

5 Release contents

[Table 2](#) provides an overview of the MCUXpresso SDK release package contents and locations.

Table 2. Release contents

Deliverable	Location
Amazon web services examples	<i><install_dir>/boards/<board>aws_examples</i>
Boards	<i><install_dir>/boards</i>
CM33/DSP communication stack	<i><install_dir>/middleware/multicore</i>
CMSIS Arm Cortex®-M header files, DSP library source	<i><install_dir>/CMSIS</i>
CMSIS drivers	<i><install_dir>/devices/<device_name>/cmsis_drivers</i>
Cortex Microcontroller Software Interface Standard (CMSIS) driver examples	<i><install_dir>/boards/<board_name>/cmsis_driver_examples</i>
Cypress Wiced SDK (WiFi, BLE)	<i><install_dir>/middleware/wiced</i>
Cypress WiFi stack examples	<i><install_dir>/boards/<board_name>/wifi_cypress_examples</i>
Demo applications	<i><install_dir>/boards/<board_name>/demo_apps</i>
Documentation	<i><install_dir>/docs</i>
Driver examples	<i><install_dir>/boards/<board_name>/driver_examples</i>
Driver, SoC header files, extension header files and feature header files, utilities	<i><install_dir>/devices/<device_name></i>
DSP examples	<i><install_dir>/boards/<board_name>/dsp_examples</i>
SP library	<i><install_dir>/middleware/dsp</i>
emWin examples	<i><install_dir>/boards/<board_name>/emwin_examples</i>
emWin library	<i><install_dir>/middleware/emWin</i>
FatFS stack	<i><install_dir>/middleware/fatfs</i>
fbdev, flash and mpi_loader examples	<i><install_dir>/boards/<board_name>/component_examples</i>
Jpeg examples	<i><install_dir>/boards/<board_name>/jpeg_examples</i>
LibJpeg	<i><install_dir>/middleware/libjpeg</i>
LittlevGL examples	<i><install_dir>/boards/<board_name>/littlevgl_examples</i>

Table continues on the next page...

Table 2. Release contents (continued)

Deliverable	Location
LittlevGL library	<i><install_dir>/middleware/littlevgl</i>
lwIP stack	<i><install_dir>/middleware/lwip</i>
mbed TLS	<i><install_dir>/middleware/mbedtls</i>
mbed TLS examples	<i><install_dir>/boards/<board_name>/mbedtls_examples</i>
nghttp2	<i><install_dir>/middleware/nghttp2</i>
NXP WiFi	<i><install_dir>/middleware/wifi</i>
NXP WiFi examples	<i><install_dir>/boards/<board_name>/wifi_examples</i>
Peripheral Drivers	<i><install_dir>/devices/<device_name>/drivers</i>
RTOS examples	<i><install_dir>/boards/<board_name>/rtos_examples</i>
RTOS Kernel Code	<i><install_dir>/rtos</i>
SDMMC card driver	<i><install_dir>/middleware/sdmmc</i>
SDMMC examples	<i><install_dir>/boards/<board_name>/sdmmc_examples</i>
Tools	<i><install_dir>/tools</i>
Trustzone examples	<i><install_dir>/boards/<board_name>/trustzone_examples</i>
USB demo applications	<i><install_dir>/boards/<board_name>/usb_examples</i>
USB stack	<i><install_dir>/middleware/usb</i>
Utilities such as debug console	<i><install_dir>/devices/<device_name>/utilities</i>
VGLite examples	<i><install_dir>/boards/<board_name>/vglite_examples</i>
VGLite Graphic library	<i><install_dir>/middleware/vglite</i>

6 MCUXpresso SDK release package

The MCUXpresso SDK release package content is aligned with the silicon subfamily it supports. This includes the boards, CMSIS, devices, documentation, middleware, and RTOS support.

6.1 Device support

The device folder contains the whole software enablement available for the specific System-on-Chip (SoC) subfamily. This folder includes clock-specific implementation, device register header files, device register feature header files, CMSIS derived device SVD, and the system configuration source files. Included with the standard SoC support are folders containing peripheral drivers, toolchain support, and a standard debug console.

The device-specific header files provide a direct access to the microcontroller peripheral registers. The device header file provides an overall SoC memory mapped register definition. The folder also includes the feature header file for each peripheral on the microcontroller.

The toolchain folder contains the startup code and linker files for each supported toolchain. The startup code is a CMSIS compliant startup code that efficiently transfers the code execution to the main() function.

6.1.1 Board support

The boards folder provides the board-specific demo applications, driver examples, RTOS, and middleware examples.

6.1.2 Demo applications and other examples

The demo applications demonstrate the usage of the peripheral drivers to achieve a system level solution. Each demo application contains a readme file that describes the operation of the demo and required setup steps.

The driver examples demonstrate the capabilities of the peripheral drivers. Each example implements a common use case to help demonstrate the driver functionality.

6.2 Middleware

6.2.1 USB stack

For more information, see *MCUXpresso SDK USB Stack User's Guide* (document MCUXSDKUSBSUG).

6.2.1.1 Peripheral devices tested with USB Host stack

[Table 3](#) provides a list of USB devices tested with the USB Host stack.

Table 3. Peripheral devices

Device type	Device
USB HUB	BELKIN F5U233
	BELKIN F5U304
	BELKIN F5U307
	BELKIN F4U040
	UNITEK Y-2151
	Z-TEK ZK032A
	HYUNDAI HY-HB608
USB flash drive	ADATA C008 32 GB
	ADATA S102 8 G
	ADATA S102 16 G
	Verbatim STORE N GO USB Device 8 G
	Kingston DataTraveler DT101 G2
	SanDisk Cruzer Blade 8 GB
	Unisplendour 1 G
	Imation 2 GB
	V-mux 2 GB
	Sanmina-SCI 128 M
	Corporate Express 1 G
	TOSHIBA THUHYBS-008G 8 G
	Transcend JF700 8 G
	Netac U903 16 G
	SSK SFD205 8 GB
	Rex 4 GB

Table continues on the next page...

Table 3. Peripheral devices (continued)

Device type	Device
	SAMSUNG USB3.0 16GB
USB card reader/adaptor	SSK TF adapter Kawau Multi Card Reader Kawau TF adapter Kawau SDHC card
USB Mouse	DELL MS111-P DELL M066U0A DELL MUAVDEL8 TARGUS AMU76AP DELL MD56U0 DELL MS111-T RAPOO M110
USB Keyboard	DELL SK8135 DELL SK8115

6.2.2 File system

The FatFs file system is integrated with the MCUXpresso SDK and can be used to access either the SD card or the USB memory stick when the SD card driver or the USB Mass Storage Device class implementation is used.

6.2.3 RTOS

The MCUXpresso SDK is integrated with FreeRTOS OS.

6.2.4 CMSIS

The MCUXpresso SDK is shipped with the standard CMSIS development pack, including the prebuilt libraries.

6.2.5 SDMMC

The SDMMC software is integrated with MCUXpresso SDK to support SD/MMC/SDIO standard specification. This also includes a host adapter layer for bare-metal/RTOS applications.

6.2.6 VGLite: Graphic library

This software package contains VeriSilicon's platform independent VGLite Graphics library in source code.

6.2.7 emWin

The MCUXpresso SDK is pre-integrated with the SEGGER emWin GUIBuilder.

6.2.8 Other middleware

Optional middleware packages can be included in the release based on the user selection. See *<install_dir>/SW-Content-Register.txt* for a list of components and associated licenses.

7 MISRA compliance

All MCUXpresso SDK drivers comply to MISRA 2012 rules with exceptions in .

Table 4. MISRA exceptions

Exception rules	Description
Directive 4.4	Sections of code should not be commented out.
Directive 4.5	Identifiers in the same name space with overlapping visibility should be typographically unambiguous.
Directive 4.6	Typedefs that indicate size and signedness should be used in place of the basic numerical types.
Directive 4.8	If a pointer to a structure or union is never dereferenced within a translation unit, then the implementation of the object should be hidden.
Directive 4.9	A function should be used in preference to a function-like macro where they are interchangeable.
Directive 4.13	Functions which are designed to provide operations on a resource should be called in an appropriate sequence.
Rule 1.2	Language extensions should not be used.
Rule 2.3	A project should not contain unused type declarations.
Rule 2.4	A project should not contain unused tag declarations.
Rule 2.5	A project should not contain unused macro declarations.
Rule 2.6	A function should not contain unused label declarations.
Rule 2.7	There should be no unused parameters in functions.
Rule 4.2	Trigraphs should not be used.
Rule 5.1	External identifiers shall be distinct.
Rule 5.4	Macro identifiers shall be distinct.
Rule 5.9	Identifiers that define objects or functions with internal linkage should be unique.
Rule 8.7	Functions and objects should not be defined with external linkage if they are referenced in only one translation unit.
Rule 8.9	An object should be defined at block scope if its identifier only appears in a single function.
Rule 8.11	When an array with external linkage is declared, its size should be explicitly specified.
Rule 8.13	A pointer should point to a const-qualified type whenever possible.

Table continues on the next page...

Table 4. MISRA exceptions (continued)

Exception rules	Description
Rule 10.5	The value of an expression should not be cast to an inappropriate essential type.
Rule 11.4	A conversion should not be performed between a pointer to object and an integer type.
Rule 11.5	A conversion should not be performed from pointer to void into pointer to object.
Rule 12.1	The precedence of operators within expressions should be made explicit.
Rule 12.3	The comma operator should not be used.
Rule 12.4	Evaluation of constant expressions should not lead to unsigned integer wrap-around.
Rule 13.3	A full expression containing an increment (++) or decrement (–) operator should have no other potential side effects other than that caused by the increment or decrement operator.
Rule 15.4	There should be no more than one break or go to statement used to terminate any iteration statement.
Rule 17.5	The function argument corresponding to a parameter declared to have an array type shall have an appropriate number of elements.
Rule 17.8	A function parameter should not be modified.
Rule 19.2	The union keyword should not be used.
Rule 20.1	#include directives should only be preceded by preprocessor directives or comments.
Rule 20.10	The # and ## preprocessor operators should not be used.
Rule 21.1	#define and #undef shall not be used on a reserved identifier or reserved macro name.
Rule 21.2	A reserved identifier or macro name shall not be declared.
Rule 21.12	The exception handling features of <fenv.h> should not be used.

8 Known issues

8.1 Maximum file path length in Windows 7[®] operating system

The Windows 7 operating system imposes a 260-character maximum length for file paths. When installing the MCUXpresso SDK, place it in a directory close to the root to prevent file paths from exceeding the maximum character length specified by the Windows operating system. The recommended location is the *C:\nxp* folder.

8.2 USB issue

8.2.1 Low speed devices not supported

The host examples cannot support low-speed devices.

8.3 IAR cannot debug RAM application with J-Link

Currently, IAR will call J-Link reset after the application is downloaded to SRAM, but such operation will cause SRAM data lost. Here's a workaround to avoid real reset, with the cost of no any reset during the debugging, and hardware status uncleared.

1. Build and debug IAR project once and see the **settings** folder created.
2. Create the `<ProjectName>_<DebugConfig>.JLinkScript` file in the **settings** folder with the following contents.

```
void ResetTarget(void) {  
    JLINK_TARGET_Halt();  
}
```

3. Debug the project again and now it can work.



8.4 New Project Wizard compile failure

The following components request the user to manually select other components that they depend upon in order to compile. These components depend on several other components and the New Project Wizard (NPW) is not able to decide which one is needed by the user.

NOTE

xxx means core variants, such as, cm0plus, cm33, cm4, cm33_nodsp.

Components: issdk_mag3110, issdk_host, systick, gpio_kinetis, gpio_lpc, issdk_mpl3115, sensor_fusion_agm01, sensor_fusion_agm01_lpc, issdk_mma845x, issdk_mma8491q, issdk_mma865x, issdk_mma9553, and CMSIS_RTOS2.

Also for low-level adapter components, currently the different types of the same adapter cannot be selected at the same time. For example, if there are two types of timer adapters, gpt_adapter and pit_adapter, only one can be selected as timer adapter in one project at a time. Duplicate implementation of the function results in an error.

NOTE

Most of middleware components have complex dependencies and are not fully supported in new project wizard. Adding a middleware component may result in compile failure.

8.5 Semihosting not supported on shell-related demo

The semihosting is not supported on shell-related demos (shell, dsp_xaf_demo_cm33, and dsp_audio_demo_bm_cm33). The shell component is based on debug console and serial manager. When semihosting is used, debug console and serial manager are bypassed. So the shell related demos cannot work with semihosting.

8.6 CMSIS PACK new project compile failure

The generated configuration cannot be applied globally. The components, `serial_manager_usb_cdc_virtual` and `serial_manager_usb_cdc_virtual_xxx` (xxx means core variants like `cm0plus`, `cm33`, `cm4`, and `cm33_nodsp`) are unsupported for new project wizard of CMSIS pack and will lead to compile failure if selected while creating a new project(s).

8.7 Dsp_xaf_demo cannot handle certain audio file types

Audio files encoded using ffmpeg tool and Lavf/Lavc codec cannot be played. M4A format of AAC files is not supported either.

If the demo stops responding while trying to play any of unsupported files, please reset the board.

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.

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.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. 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, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, 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, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. 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. Oracle and 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.

© NXP B.V. 2021.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: January 21, 2021

Document identifier: MCUSDKRT5XXRN

