

# MCUXpresso SDK Release Notes

## Supporting LPCXpresso54608, LPCXpresso54618, and LPCXpresso54628

### Contents

## 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 lwIP, integration with WolfSSL and mbed TLS cryptography libraries, 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 the latest version of this and other MCUXpresso SDK documents, see the MCUXpresso SDK homepage [MCUXpresso-SDK: Software Development Kit](#).

### NOTE

See the attached Change Logs section at the end of this document to reference the device-specific driver logs, middleware logs, and RTOS log.

1	Overview.....	1
2	MCUXpresso SDK.....	1
3	Development tools.....	2
4	Supported development systems.....	2
5	Release contents.....	2
6	MCUXpresso SDK release package.....	3
7	MISRA compliance.....	4
8	Known issues.....	6

## 2 MCUXpresso SDK



## Development tools

As part of the MCUXpresso software and tools, MCUXpressoSDK is the evolution of Kinetis SDK v2.3.0, 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, a new Eclipse-based toolchain that works with all MCUXpresso SDKs. Easily import your SDK into the new toolchain to have 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 for easy cloning of existing SDK examples and demos, allowing users to easily leverage the existing software examples provided by the SDK for their own projects.

### NOTE

In order to maintain compatibility with legacy FSL code, the filenames and 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.

## 3 Development tools

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

- IAR Embedded Workbench for Arm version 8.22.2
- MDK-Arm Microcontroller Development Kit (Keil)® 5.24a
- Makefiles support with GCC revision 7-2017-q4-major from Arm Embedded
- MCUXpresso IDE v10.2.0

## 4 Supported development systems

This release supports boards and devices listed in this table. Boards and devices in boldface were tested in this release:

**Table 1. Supported MCU devices and development boards**

Development boards	MCU devices
<b>LPCXpresso54608, LPCXpresso54628, LPCXpresso54618</b>	LPC54605J256ET180, LPC54605J512ET180, LPC54606J256ET180, LPC54606J512BD208, LPC54607J256ET180, LPC54607J512ET180, LPC54607J256BD208, <b>LPC54608J512ET180</b> , LPC54608J512BD208, LPC54628J512ET180, LPC54616J256ET180, LPC54616J512BD208, <b>LPC54618J512ET180</b> , LPC54618J512BD208

## 5 Release contents

This table provides an overview of the MCUXpresso SDK release package contents and locations.

**Table 2. Release contents**

Deliverable	Location
Boards	<install_dir>/boards

*Table continues on the next page...*

**Table 2. Release contents (continued)**

Demo applications	<install_dir>/boards/<board_name>/demo_apps
Driver examples	<install_dir>/boards/<board_name>/driver_examples
Documentation	<install_dir>/docs
Middleware	<install_dir>/middleware
DMA manager	<install_dir>/middleware/dma_manager
SDMMC card driver	<install_dir>/middleware/sdmmc
Driver, SoC header files, extension header files and feature header files, utilities	<install_dir>/devices/<device_name>
Cortex Microcontroller Software Interface Standard (CMSIS) ARM Cortex®-M header files, DSP library source	<install_dir>/CMSIS
Peripheral Drivers	<install_dir>/devices/<device_name>/drivers
Utilities such as debug console	<install_dir>/devices/<device_name>/utilities
Tools	<install_dir>/tools

## 6 MCUXpresso SDK release package

The MCUXpresso SDK release package contents are 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 all available software enablement for the specific System-on-Chip (SoC) subfamily. This folder includes clock-specific implementation, device register header file, device register feature header file, 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 simple debug console.

The device-specific header files provide a direct access to the MCU peripheral registers. The device header file provides an overall SoC memory mapped register definition. In addition to the overall device memory mapped header file, the MCUXpresso SDK also includes the feature header file for each peripheral instantiated on the SoC.

The toolchain folder contains the startup code and linker files for each supported toolchain. The startup code is a CMSIS-compliant startup 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

## MISRA compliance

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.

The RTOS and middleware folders each contain examples demonstrating the use of the included source.

## 6.2 Middleware

### 6.2.1 USB stack

See the *MCUXpresso SDK USB Stack User's Guide* (document MCUXSDKUSBSUG) for more information.

### 6.2.2 CMSIS

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

### 6.2.3 emWin

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

## 7 MISRA compliance

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

**Table 3. 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	Typedef that indicate size and signedness should be used in place of the basic numerical type.
Directive 4.8	If a pointer to a structure or union is never dereferenced within a transaction unit then the implementation of the object should hidden.
Directive 4.9	A function should be used in preference to a function like macro where they are interchangeable.
Directive 4.10	Precautions shall be taken in order to prevent the contents of a header file being included more than once.

*Table continues on the next page...*

**Table 3. MISRA exceptions (continued)**

Directive 4.11	The validity of values passed to library functions shall be checked.
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 2.7	There should be no unused parameters in functions.
Rule 3.1	The character sequences <code>/*</code> and <code>//</code> shall not be used within a comment.
Rule 5.1	External identifiers shall distinct.
Rule 5.3	A identifier declared in an inner scope shall not hide an identifier declared in an outer scope.
Rule 5.7	A tag name shall be a unique identifier.
Rule 5.9	Identifiers that define objects or functions with external linkage shall be unique.
Rule 8.13	A pointer should point to a const-qualified type whenever possible.
Rule 8.3	All declarations of an object or function shall use the same names and type qualifiers.
Rule 8.6	An identifier with external linkage shall have exactly one external definition.
Rule 8.7	Octal constants shall not be used.
Rule 8.9	A object should be defined at block scope if its identified only appears in a single function.
Rule 10.1	Operands shall not be of an inappropriate essential type.
Rule 10.3	The value of an expression shall not be assigned to an object with a narrower essential type of a different essential type category.
Rule 10.4	Both operands of an operator in which the usual arithmetic conversions are performed shall have the same essential type category.
Rule 10.5	The value of an expression should not be cast to an inappropriate essential type.
Rule 10.6	The value of a composite expression shall not be assigned to an object with wider essential type.
Rule 10.7	If a composite expression is used as one operand of an operator in which the usual arithmetic conversions are performed then the other operand shall not have wider essential type.
Rule 10.8	The value of a composite expression shall not be cast to a different essential type category or a wider essential type.
Rule 11.1	Conversions shall not be performed between a pointer to a function and any other type.
Rule 11.3	A case shall not be performed between a pointer to object type and a pointer to a different object type.
Rule 11.4	A conversion should not be performed between a pointer to object and an integer type.

*Table continues on the next page...*

**Table 3. MISRA exceptions (continued)**

Rule 11.5	A conversion should not be performed from pointer to void into pointer to object.
Rule 11.6	A cast shall not be performed between pointer to void and an arithmetic type.
Rule 12.1	The precedence of operators within expressions should be made explicit.
Rule 12.2	The right hand operator of a shift operator shall lie in the range zero to one less than the width in bits of the essential type of the left hand operand.
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 13.5	The right hand operand of a logical && or    operator shall not contain persistent side effects.
Rule 14.2	A for loop shall be well formed.
Rule 14.4	The controlling expressions of an statement and the controlling expression of an iteration-statement shall have essentially Boolean type.
Rule 15.5	A function should have a single point of exit at the end.
Rule 16.1	All switch statements shall be well-formed.
Rule 17.7	The feature of <stdarg.h> shall not be used.
Rule 18.4	The +, -, += and -= operators should not be applied to an expression of pointer type.
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.

## 8 Known issues

### 8.1 Maximum file path length in Windows® 7 Operating System

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 HUB Power supply

The external power supply of the USB HUB must be provided before it can be used. The development board power is not enough to supply multi-level USB HUBs and connected devices. Therefore, the external USB HUB that is connected to the development board should have its own power supply.

## 8.3 USBFS controller issue

Because of the USBFS controller design issues, the USB host suspend/resume demos (usb\_suspend\_resume\_host\_hid\_mouse) of the full speed controller do not support the low speed device directly.

## 8.4 SDIF issue

If the user uses this module to access the SD card/MMC card, sd\_data[4-7] must be enabled and configured as input and pull up. It is better to use P4\_17 to P5\_10, otherwise 4-bit mode does not work.

## 8.5 USB High-Speed Interrupt Endpoint issue

If the user wants to use a high-speed interrupt endpoint, the maximum packet size should be 512 bytes.

---

# **MCUXpresso SDK Release Notes Supporting LPCXpresso54608, LPCXpresso54618, and LPCXpresso54628**

**Change Logs**



# Contents

<b>Driver Change Log</b>	<b>1</b>
ADC . . . . .	1
CRC . . . . .	1
CTIMER . . . . .	1
DMA . . . . .	1
DMIC . . . . .	2
EEPROM . . . . .	2
EMC . . . . .	2
ENET . . . . .	2
FLASHIAP . . . . .	2
FLEXCOMM . . . . .	2
I2C . . . . .	3
I2S . . . . .	3
SPI . . . . .	3
USART . . . . .	4
FMC . . . . .	4
FMEAS . . . . .	4
GINT . . . . .	4
GPIO . . . . .	4
INPUTMUX . . . . .	5
IOCON . . . . .	5

# Contents

Title	Page Number
<b>LCDC</b> . . . . .	<b>5</b>
<b>MCAN</b> . . . . .	<b>5</b>
<b>MRT</b> . . . . .	<b>5</b>
<b>PDB</b> . . . . .	<b>5</b>
<b>PINT</b> . . . . .	<b>6</b>
<b>RIT</b> . . . . .	<b>6</b>
<b>RNG</b> . . . . .	<b>6</b>
<b>RTC</b> . . . . .	<b>6</b>
<b>SCTIMER</b> . . . . .	<b>6</b>
<b>SDIF</b> . . . . .	<b>7</b>
<b>SPI Flash Interface</b> . . . . .	<b>7</b>
<b>SYSCON</b> . . . . .	<b>8</b>
<b>UTICK</b> . . . . .	<b>8</b>
<b>WWDT</b> . . . . .	<b>8</b>
<b>CLOCK</b> . . . . .	<b>8</b>
<b>POWER</b> . . . . .	<b>8</b>
<b>RESET</b> . . . . .	<b>9</b>
<b>Middleware Change Log</b>	<b>10</b>
<b>emWin library</b> . . . . .	<b>10</b>
<b>FatFs for MCUXpresso SDK</b> . . . . .	<b>10</b>
<b>lwIP for MCUXpresso SDK</b> . . . . .	<b>10</b>
<b>SDMMC</b> . . . . .	<b>11</b>
<b>USB stack for MCUXpresso SDK</b> . . . . .	<b>13</b>



# Contents

	Title	Page Number
	<b>RTOS Change Log</b>	<b>17</b>
	<b>FreeRTOS for MCUXpresso SDK . . . . .</b>	<b>17</b>

# 1 Driver Change Log

## ADC

The current ADC driver version is 2.2.0.

- 2.2.0
  - Updated "ADC\_DoSelfCalibration" API and "adc\_config\_t" structure to match LPC845.
- 2.1.0
  - Renamed "ADC\_EnableShresholdCompareInterrupt" to "ADC\_EnableThresholdCompareInterrupt".
- 2.0.0
  - Initial version.

## CRC

The current CRC driver version is 2.0.1.

- 2.0.1
  - Fixed KPSDK-13362. MDK compiler issue when writing to WR\_DATA with -O3 optimize for time.
- 2.0.0
  - Initial version.

## CTIMER

The current CTimer driver version is 2.0.1.

- 2.0.1
  - API Interface Change Added CTIMER\_SetupPwmPeriod and CTIMER\_UpdatePwmPulsePeriod API. These two APIs can set up the right PWM with high resolution.
- 2.0.0
  - Initial version.

## DMA

The current DMA driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **DMIC**

The current DMIC driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **EEPROM**

The current eeprom driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **EMC**

The current EMC driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **ENET**

The current ENET driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **FLASHIAP**

The current FLASHIAP driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **FLEXCOMM**

The current FLEXCOMM driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **I2C**

The current I2C driver version is 2.0.3.

- 2.0.3
  - Unify component full name to FLEXCOMM I2C(DMA/FREERTOS) Driver
- 2.0.2
  - Improvements: In slave IRQ:
    1. Changed slave receive process to first set the I2C\_SLVCTL\_SLVCONTINUE\_MASK to ack the received data, then do data receive.
    2. Improved slave transmit process to set the I2C\_SLVCTL\_SLVCONTINUE\_MASK immediately after write the data.
- 2.0.1
  - Improvements:
    - \* Added I2C\_WATI\_TIMEOUT macro to allow user to specify the timeout times for waiting flags in functional API and blocking transfer API.
- 2.0.0
  - Initial version.

## **I2S**

The current I2S driver version is 2.0.1.

- 2.0.1
  - Unify component full name to FLEXCOMM I2S(DMA) Driver
- 2.0.0
  - Initial version.

## **SPI**

The current SPI driver version is 2.0.2.

- 2.0.2
  - Unify component full name to FLEXCOMM SPI(DMA/FREERTOS) Driver
- 2.0.1
  - Changed the data buffer from uint32\_t to uint8\_t which matches the real applications for SPI DMA driver.
- Added dummy data setup API to allow users to configure the dummy data to be transferred.
  - Added new APIs for half-duplex transfer function, users can send and receive data by one API in polling/interrupt/DMA way, and users can choose either transmit first or receivefirst. Besides, the PCS pin can be configured as assert status in transimission (between transmit and receive) by setting the isPcsAssertInTransfer to true.
- 2.0.0
  - Initial version.

## USART

- The current USART driver version is 2.0.1.
- 2.0.1
  - Unify component full name to FLEXCOMM USART(DMA/FREERTOS) Driver
- 2.0.0
  - Initial version.

## FMC

The current FMC driver version is 2.0.0.

- 2.0.0
  - Initial version.

## FMEAS

The current FMEAS driver version is 2.0.0.

- 2.0.0
  - Initial version ported from LPCOpen.

## GINT

The current GINT driver version is 2.0.0.

- 2.0.0
  - Initial version.

## GPIO

The current GPIO driver version is 2.1.1.

- 2.1.1:
  - API interface changes:
    - \* Refined naming of API while keep all original APIs, marking them as deprecated. Original API will be removed in next release. The mainin change is update API with prefix of `_PinXXX()` and `_PorortXXX`
- 2.1.0
  - Added GPIO initialize API.
- 2.0.0
  - Initial version.

## **INPUTMUX**

The current INPUTMUX driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **IOCON**

The current IOCON driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **LCDC**

The current LCDC driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **MCAN**

The current MCAN driver version is 2.0.1.

- 2.0.1
  - LPC54608 chip does not support the FD feature, so added a feature macro for it.
- 2.0.0
  - Initial version.

## **MRT**

The current MRT driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **PDB**

The current PDB driver version is 2.0.1.

- 2.0.1
  - Changed PDB register base array to const.
- 2.0.0



- Initial version.

## **PINT**

The current PINT driver version is 2.0.1.

- 2.0.1
  - Bug fix:
    - \* Updated PINT driver to clear interrupt only in Edge sensitive.
- 2.0.0
  - Initial version.

## **RIT**

The current RIT driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **RNG**

The current RNG driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **RTC**

The current RTC driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **SCTIMER**

The current SCTimer driver version is 2.0.0.

- 2.0.0
  - Initial version.

## SDIF

The current SDIF driver version is 2.0.7.

- 2.0.7
- Bug Fix:
  - Enlarge the timeout value to avoid command conflict issue..
- 2.0.6
- Bug Fix:
  - Removed assert(srcClock\_Hz <= FSL\_FEATURE\_SDIF\_MAX\_SOURCE\_CLOCK).
  - Used hardware reset instead of software reset during initialize.
- 2.0.5
- New features:
  - Added non-word aligned data address and DMA descriptor address transfer support. Once one of the above addresses is not aligned, switch to host transfer mode.
- Bug fix:
  - Fixed the DMA suspend during initialization issue.
  - Removed useless memset function call.
- 2.0.4
  - Added cardInserted/cardRemoved callback function.
  - Added host base address/user data parameter for all call back functions.
- 2.0.3
  - Improved Clock Delay macro to allow the user to redefine and remove useless delay for clock below 25 MHz.
- 2.0.2
  - Bug fix:
    - \* Fixed the issue where the status flag cannot be cleared entirely after transfer complete.
- 2.0.1
  - New features:
    - \* Improve interrupt transfer callback.
  - Bug fix:
    - \* Added assert to limit the SDIF source clock below 52 MHz.
- 2.0.0
  - Initial version.

## SPI Flash Interface

The current SPIFI driver version is 2.0.2.

- 2.0.2
  - Fixed the set command function issue. After the command set, there is no wait for the CMD flag, as it may be cleared by CS deassert.
- 2.0.1
  - Added API to read/write 1/2 Bytes data from/to SPIFI. This interface are useful for flash command, which only needs 1/2 Bytes data. The previous driver needed users to make sure the

minimum length should be 4, which may have issues in some flash commands.

- 2.0.0
  - Initial version.

## **SYSCON**

The current SYSCON driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **UTICK**

The current UTICK driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **WWDT**

The current WWDT driver version is 2.0.0.

- 2.0.0
  - Initial version.

## **CLOCK**

The current CLOCK driver version is 2.0.1.

- 2.0.1
  - some minor fixes.
- 2.0.0
  - initial version.

## **POWER**

The current POWER driver version is 2.0.0.

- 2.0.0
  - initial version.

## **RESET**

The current RESET driver version is 2.0.0.

- 2.0.0
  - initial version.

## 2 Middleware Change Log

### emWin library

The currently supported version is 5.38a.

### FatFs for MCUXpresso SDK

The current version is FatFs R0.13a\_rev0.

- R0.13a\_rev0
  - Upgraded to version 0.13a. Added patch ff\_13a\_p1.diff.
- R0.12c\_rev1
  - Add nand disk support.
- R0.12c\_rev0
  - Upgraded to version 0.12c and applied patches ff\_12c\_p1.diff and ff\_12c\_p2.diff.
- R0.12b\_rev0
  - Upgraded to version 0.12b.
- R0.11a
  - Added glue functions for low-level drivers (SDHC, SDSPI, RAM, MMC). Modified diskio.c.
  - Added RTOS wrappers to make FatFs thread safe. Modified syscall.c.
  - Renamed ffconf.h to ffconf\_template.h. Each application should contain its own ffconf.h.
  - Included ffconf.h into diskio.c to enable the selection of physical disk from ffconf.h by macro definition.
  - Conditional compilation of physical disk interfaces in diskio.c.

### lwIP for MCUXpresso SDK

The current version of lwIP is based on lwIP 2.0.3.

- 2.0.3\_rev1
  - New features:
    - \* Ported lwIP 2.0.3 (2017-09-15, SHA-1: 92f23d6ca0971a32f2085b9480e738d34174417b) to KSDK 2.0.0.
- 2.0.2\_rev1
  - New features:
    - \* Ported lwIP 2.0.2 (2017-03-13, SHA-1: c0862d60746e2d1ceae69af4c6f24e469570ecef) to KSDK 2.0.0.
- 2.0.0\_rev3
  - New features:
    - \* Ported lwIP 2.0.0 (2016-11-10, SHA-1: 216bf89491815029aa15463a18744afa04df58fe) to KSDK 2.0.0.
- 2.0.0\_rev2

- New features:
  - \* Ported lwIP 2.0.0 RC2 (2016-08-08, SHA-1: b1dfd00f9233d124514a36a8c8606990016f2ad4) to KSDK 2.0.0.
- 2.0.0\_rev1
  - New features:
    - \* Ported lwIP 2.0.0 RC0 (2016-05-26) to KSDK 2.0.0.
    - \* Changed lwIP bare-metal examples to use poll-driven approach instead of interrupt-driven one.
- 1.4.1\_rev2
  - New features:
    - \* Enabled critical sections in lwIP.
  - Bug fixes:
    - \* Fixed default lwIP packet-buffer size to be able to accept a maximum size frame from the ENET driver.
    - \* Fixed possible drop of multi-frame packets during transmission.
- 1.4.1\_rev1
  - New features:
    - \* Ported lwIP 1.4.1 to KSDK 2.0.0.

## **SDMMC**

The current driver version is 2.2.4.

- 2.2.4
  - Bug fix:
    - \* Fixed DDR mode data sequence miss issue, which is caused by NIBBLE\_POS.
  - New features:
    - \* Increased g\_sdmmc 512byte to improve the performance when application use a non-word align data buffer address.
    - \* Used OCR access mode bits to determine the mmccard high capacity flag.
    - \* Enabled auto cmd12 for SD read/write.
    - \* Disabled DDR mode frequency multiply by 2.
- 2.2.3
  - Bug fix:
    - \* Added reponse check for send operation condition command. If not checked, the card may occasionally init fail.
- 2.2.2
  - Moved set card detect priority operation before enable IRQ.
- 2.2.1
  - New features:
    - \* Improved MMC Boot feature.
    - \* Keep SD\_Init/SDIO\_Init function for forward compatibility.
- 2.2.0
  - New features:

- \* Separated the SD/MMC/SDIO init API to xxx\_CardInit/xxx\_HostInit.
- \* Allowed user register card detect callback, select card detect type, and determine the card detect timeout value.
- \* Allowed user register the power on/off function, and determine the power on/off delay time.
- \* SD\_Init/SDIO\_Init will be deprecated in the next version.
- \* Added write complete wait operation for MMC\_Write to fix command timeout issue.
- 2.1.6
  - Enhanced SD IO default driver strength.
- 2.1.5
  - Fixed coverity issue.
  - Fixed SD v1.x card write fail issue. It was caused by the block length set error.
  - Improved SDIO card init sequence and add retry option for SDIO\_SwitchToHighSpeed function.
- 2.1.4
  - Miscellaneous:
    - \* Added Host reset function for card re-initialization.
    - \* Added Go\_Idle function for SDIO card.
    - \* Added Host\_ErrorRecovery function for host error recovery procedure.
    - \* Added cache maintain operation
    - \* Added HOST\_CARD\_INSERT\_CD\_LEVEL to improve compatibility.
  - Bug fix:
    - \* Fixed card cannot detect dynamically.
- 2.1.3
  - Bug fix:
    - \* Non high-speed sdcard init fail at switch to high speed.
  - Miscellaneous:
    - \* Optimized tuning/mmc switch voltage/mmc select power class/mmc select timing function.
    - \* Added strobe dll for mmc HS400 mode.
    - \* Added Delay for SDCard power up.
- 2.1.2
  - New features:
    - \* Added fsl\_host.h to provide prototype to adapt different controller IPs(SDHC/SDIF).
    - \* Added adaptor code in SDMMC/Port folder to adapt different host controller IPs with different. transfer modes(interrupt/polling/freertos). Application includes a different adaptor code to make application more simple.
    - \* Adaptor code provides HOST\_Init/HOST\_Deinit/CardInsertDetect. APIs to do host controller initialize and transfer function configuration. SDMMC card stack uses adaptor code inside stack to wait card insert and configure host when calling card init APIs (SD\_Init/MMC\_Init/SDIO\_Init).
    - \* This change requires the user to include host adaptor code into the application. If not changed, link errors saying it cannot find the definition of HOST\_Init/HOST\_Deinit/CardInsertDetect appear.
  - New features: Improved SDMMC to support SD v3.0 and emmc v5.0.

- Bug fix:
  - \* Fixed wrong comparison between count and length in MMC\_ReadBlocks/MMC\_WriteBlocks.
- 2.1.1
  - Bug fix:
    - \* Fixed the block range boundary error when transferring data to MMC card.
    - \* Fixed the bit mask error in the SD card switch to high speed function.
  - Other changes:
    - \* Added error code to indicate that SDHC ADMA1 transfer type is not supported yet.
    - \* Optimized the SD card initialization function.
- 2.1.0
  - Bug fix:
    - \* Change the callback mechanism when sending a command.
    - \* Fix the performance low issue when transferring data.
  - Other changes:
    - \* Changed the name of some error codes returned by internal function.
    - \* Merged all host related attributes to one structure.
    - \* Optimize the function of setting maximum data bus width for MMC card.

## USB stack for MCUXpresso SDK

The current version of USB stack is 2.0.1.

- 2.0.1
  - Bug fix:
    - \* fixed some USB issues.
    - \* Change the audio codec interfaces.
- 2.0.0
  - New features:
    - \* PTN5110N support.
  - Bug fix:
    - \* Added some comments, fixed some minor USB issues.
- 1.9.0
  - New features:
    - \* Examples:
      - usb\_pd\_alt\_mode\_dp\_host
- 1.8.2
  - Updated license.
- 1.8.1
  - Bug fix:
    - \* Verified some hardware issues, support aruba\_flashless.
- 1.8.0
  - New features:
    - \* Examples:



- usb\_device\_composite\_cdc\_vcom\_cdc\_vcom
- usb\_device\_composite\_hid\_audio\_unified
- usb\_pd\_sink\_battery
- Changed usb\_pd\_battery to usb\_pd\_charger\_battery.

Bug fix:

- Code cleaned up, removed some irrelevant code.

#### 1.7.0

- New features:
  - USB PD stack support.
- Examples
  - usb\_pd
  - usb\_pd\_battery
  - usb\_pd\_source\_charger

#### 1.6.3

- Bug fix: -IP3511\_HS driver control transfer sequence issue, enabled 3511 ip cv test.

#### 1.6.2

- New features:
  - Multi instance support.

#### 1.6.1

- New features:
- Changed the struct variable address method for device\_video\_virtual\_camera and host\_phdc\_manager.

#### 1.6.0

- New features:
  - Supported Device Charger Detect feature on usb\_device\_hid\_mouse.

#### 1.5.0

- New features:
  - Supported controllers
    - \* OHCI (Full Speed, Host mode)
    - \* IP3516 (High Speed, Host mode)
    - \* IP3511 (High Speed, Device mode)
  - Examples:
    - \* usb\_lpm\_device\_hid\_mouse
    - \* usb\_lpm\_device\_hid\_mouse\_lite
    - \* usb\_lpm\_host\_hid\_mouse

#### 1.4.0

- New features:
  - Examples:

- \* usb\_device\_hid\_mouse/freertos\_static
- \* usb\_suspend\_resume\_device\_hid\_mouse\_lite

### 1.3.0

- New features:
  - Supported roles
    - \* OTG
  - Supported classes
    - \* CDC RNDIS
  - Examples
    - \* usb\_otg\_hid\_mouse
    - \* usb\_device\_cdc\_vnic
    - \* usb\_suspend\_resume\_device\_hid\_mouse
    - \* usb\_suspend\_resume\_host\_hid\_mouse

### 1.2.0

- New features:
  - Supported controllers
    - \* LPC IP3511 (Full Speed, Device mode)

### 1.1.0

- Bug fix:
  - Fixed some issues in USB certification.
  - Changed VID and Manufacturer string to NXP.
- New features:
  - Supported classes
    - \* Pinter
  - Examples:
    - \* usb\_device\_composite\_cdc\_msc\_sdcard
    - \* usb\_device\_printer\_virtual\_plain\_text
    - \* usb\_host\_printer\_plain\_text

### 1.0.1

- Bug fix:
  - Improved the efficiency of device audio speaker by changing the transfer mode from interrupt to DMA, thus providing the ability to eliminate the periodic noise.

### 1.0.0

- New features:
  - Supported roles
    - \* Device
    - \* Host
  - Supported controllers:
    - \* KHCI (Full Speed)
    - \* EHCI (High Speed)
  - Supported classes:

- \* AUDIO
- \* CCID
- \* CDC
- \* HID
- \* MSC
- \* PHDC
- \* VIDEO
- Examples:
  - \* usb\_device\_audio\_generator
  - \* usb\_device\_audio\_speaker
  - \* usb\_device\_ccid\_smart\_card
  - \* usb\_device\_cdc\_vcom
  - \* usb\_device\_cdc\_vnic
  - \* usb\_device\_composite\_cdc\_msc
  - \* usb\_device\_composite\_hid\_audio
  - \* usb\_device\_composite\_hid\_mouse\_hid\_keyboard
  - \* usb\_device\_hid\_generic
  - \* usb\_device\_hid\_mouse
  - \* usb\_device\_msc\_ramdisk
  - \* usb\_device\_msc\_sdcard
  - \* usb\_device\_phdc\_weighscale
  - \* usb\_device\_video\_flexio\_ov7670
  - \* usb\_device\_video\_virtual\_camera
  - \* usb\_host\_audio\_speaker
  - \* usb\_host\_cdc
  - \* usb\_host\_hid\_generic
  - \* usb\_host\_hid\_mouse
  - \* usb\_host\_hid\_mouse\_keyboard
  - \* usb\_host\_msd\_command
  - \* usb\_host\_msd\_fatfs
  - \* usb\_host\_phdc\_manager
  - \* usb\_keyboard2mouse
  - \* usb\_pin\_detect\_hid\_mouse

### 3 RTOS Change Log

#### FreeRTOS for MCUXpresso SDK

The current version is FreeRTOS 9.0.0. Original package is available at [freertos.org](http://freertos.org).

- 9.0.0\_rev3
  - New features:
    - \* Tickless idle mode support for Cortex-A7. Add fsl\_tickless\_epit.c and fsl\_tickless\_generic.h in portable/IAR/ARM\_CA9 folder.
    - \* Enabled float context saving in IAR for Cortex-A7. Added configUSE\_TASK\_FPU\_SUPPORT macros. Modified port.c and portmacro.h in portable/IAR/ARM\_CA9 folder.
  - Other changes:
    - \* Transformed ARM\_CM core specific tickless low power support into generic form under freertos.
- 9.0.0\_rev2
  - New features:
    - \* Enabled MCUXpresso thread aware debugging. Add freertos\_tasks\_c\_additions.h and configINCLUDE\_FREERTOS\_TASK\_C\_ADDITIONS\_H and configFRTOS\_MEMORY\_SCHEME macros.
- 9.0.0\_rev1
  - New features:
    - \* Enabled -fcto optimization in GCC by adding **attribute((used))** for vTaskSwitchContext.
    - \* Enabled KDS Task Aware Debugger. Apply FreeRTOS patch to enable configRECORD\_STACK\_HIGH\_ADDRESS macro. Modified files are task.c and FreeRTOS.h.
- 9.0.0\_rev0
  - New features:
    - \* Example freertos\_sem\_static.
    - \* Static allocation support RTOS driver wrappers.
  - Other changes:
    - \* Tickless idle rework. Support for different timers is in separated files (fsl\_tickless\_systick.c, fsl\_tickless\_lptmr.c).
    - \* Removed configuration option configSYSTICK\_USE\_LOW\_POWER\_TIMER. Low power timer is now selected by linking of appropriate file fsl\_tickless\_lptmr.c.
    - \* Removed configOVERRIDE\_DEFAULT\_TICK\_CONFIGURATION in RVDS port. Use of **attribute((weak))** is preferred solution. Not same as \_weak!
- 8.2.3
  - New features:
    - \* Tickless idle mode support.
    - \* Added template application for Kinetis Expert (KEx) tool (template\_application).
  - Other changes:
    - \* Folder structure reduction. Keep only Kinetis related parts.

**How to Reach Us:****Home Page:**[nxp.com](http://nxp.com)**Web Support:**[nxp.com/support](http://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](http://nxp.com/SalesTermsandConditions).

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, Freescale, the Freescale logo, Kinetis, and Tower are trademarks of NXP B.V. All other product or service names are the property of their respective owners. Arm, Cortex, Keil, and  $\mu$ Vision are registered trademarks of Arm Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.

© 2018 NXP B.V.

Document Number MCUXSDKLPC546XRN  
Revision 0, 05/2018

