

Kinetis SDK v.2.0.0 Release Notes Supporting FRDM-KL82 and TWR- KL82

Contents

1 Overview

The Kinetis Software Development Kit (KSDK) 2.0.0 is a collection of software enablement for Kinetis 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 and μ C/OS. In addition to the base enablement, the KSDK is augmented with demo applications and driver example projects, and API documentation to help the customers quickly leverage the support of the Kinetis SDK.

For the latest version of this and other Kinetis SDK documents, see the Kinetis SDK homepage [KINETIS-SDK: Software Development Kit for Kinetis MCUs](#).

1	Overview.....	1
2	KSDK 2.0.0.....	1
3	Development Tools.....	2
4	Supported Development Systems.....	2
5	Release Contents.....	2
6	Kinetis SDK Release Package.....	3
7	MISRA Compliance.....	6
8	Known Issues.....	6
9	Driver Change Log.....	7
10	Middleware Change Log.....	13
11	RTOS Change Log.....	16
12	Revision History.....	17

2 KSDK 2.0.0

KSDK 2.0.0 is the evolution of KSDK 1.x into a more optimized software solution. KSDK 2.0.0 eliminates the need for a separate HAL and Peripheral Driver, replacing these two layers with a single driver for each peripheral. The single driver provides both the low-level functionality of the HAL and the non-blocking interrupt-based functionality of the Peripheral Driver, enabling customers to select the right level of abstraction for their solution. Peripheral drivers in KSDK



Development Tools

2.0.0 also eliminate external software dependencies. The Operating System Abstraction, Power Manager, and Clock Manager are no longer required by the KSDK 2.0.0 drivers.

At the middleware level, RTCS and MFS have been removed, and the USB stack has been replaced with a BSD licensed solution. KSDK 2.0.0 has also aligned with ARM® architecture through the integration of mbed TLS with our accelerated cryptography drivers. This integration ensures the highest level of performance from our on-chip security peripherals.

The existing MQX™ RTOS support has been deprecated to focus on support of FreeRTOS OS and µC/OS-II and µC/OS-III.

3 Development Tools

The Kinetis SDK 2.0.0 was compiled and tested with these development tools:

- Kinetis Design Studio IDE v3.2
- IAR Embedded Workbench for ARM version 7.60
- MDK-ARM Microcontroller Development Kit (Keil)® 5.18a
- Makefiles support with GCC revision 4.9-2015-q3-update from ARM Embedded
- Atollic® TrueSTUDIO® 5.4.2

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	Kinetis MCU devices
FRDM-KL82Z	MKL82Z128VLK7 , MKL82Z128VMC7, MKL82Z128VLH7, MKL82Z128VMP7
TWR-KL82Z72M	MKL82Z128VLK7, MKL82Z128VMC7 , MKL82Z128VLH7, MKL82Z128VMP7

5 Release Contents

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

Table 2. Release contents

Deliverable	Location
Boards	<install_dir>/boards
Demo applications	<install_dir>/boards/<board_name>/demo_apps
USB demo applications	<install_dir>/boards/<board_name>/usb_examples
Driver examples	<install_dir>/boards/<board_name>/driver_examples
RTOS examples	<install_dir>/boards/<board_name>/rtos_examples
Documentation	<install_dir>/docs

Table continues on the next page...

Table 2. Release contents (continued)

USB Documentation	<install_dir>/docs/usb
Middleware	<install_dir>/middleware
DMA manager	<install_dir>/middleware/dma_manager_<version>
EMV stack	<install_dir>/middleware/emv_<version>
FatFs stack	<install_dir>/middleware/fatfs_<version>
mbedTLS stack	<install_dir>/middleware/mbedtls_<version>
SDMMC card driver	<install_dir>/middleware/sdmmc_<version>
USB stack	<install_dir>/middleware/usb_<version>
WolfSSL stack	<install_dir>/middleware/wolfssl_<version>
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
RTOS Kernel Code	<install_dir>/rtos
Tools	<install_dir>/tools

6 Kinetis SDK Release Package

The KSDK 2.0.0 release package contents are aligned with the silicon subfamily it supports. This includes the boards, CMSIS, devices, documentation, middleware, and RTOS support.

6.1 Kinetis device support

The device folder contains all available software enablement for the specific 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 Kinetis MCU peripheral registers. The device header file provides an overall System-on-Chip (SoC) memory mapped register definition. In addition to the overall device memory mapped header file, the Kinetis 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 Kinetis 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.

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 *USB Stack User's Guide* (document KSDKUSBSUG) for more information.

6.2.1.1 Peripheral devices tested with the USB Host stack

This table 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

Table continues on the next page...

Table 3. Peripheral devices (continued)

	Corporate Express 1 G TOSHIBA THUHYBS-008G 8 G Transcend JF700 8 G Netac U903 16 G SSK SFD205 8 GB Rex 4 GB SAMSUNG USB3.0 16GB
USB card reader/adapter	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
USB Printer	HP LaserJet P2055dn

6.2.2 File System

The FatFs file system is integrated with Kinetis 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.

For details, see the FatFs documentation installed at <install_dir>/middleware/fatfs_<version>/doc.

6.2.3 Security libraries

The Kinetis SDK is pre-integrated with mbedTLS and wolfSSL libraries. The integration demonstrates hardware acceleration of various cryptography algorithms and random number generation. The packages are available through separate add-on packages at kex.nxp.com, *Other Software Libraries* section.

6.2.4 RTOS

The Kinetis SDK is preintegrated with FreeRTOS OS, μ C/OS-II OS, and μ C/OS-III OS.

6.2.5 CMSIS

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

7 MISRA Compliance

All KSDK drivers and USB stack comply to MISRA 2004 rules with the following exceptions.

Exception Rules	Description
1.1	All code shall conform to ISO 9899:1990 Programming languages - C, amended and corrected by ISO/IEC 9899/COR1:1995, ISO/IEC 9899/AMD1:1995, and ISO/IEC
2.4	Sections of code should not be commented out.
5.1	Identifiers (internal and external) shall not rely on the significance of more than 31 characters.
6.3	typedefs that indicate size and signedness should be used in place of the basic types.
6.4	Bitfields shall only be defined to be of type unsigned int or signed int.
8.1	Functions shall have prototype declarations and the prototype shall be visible at both the function definition and call.
8.5	There shall be no definitions of objects or functions in a header file.
8.1	All declarations and definitions of objects or functions at file scope shall have internal linkage unless external linkage is required.
8.12	When an array is declared with external linkage, its size shall be stated explicitly or defined implicitly by initialization.
	The value of an expression of integer type shall not be implicitly converted to a different underlying type if:
	a. it is not a conversion to a wider integer type of the same signedness, or
	b. the expression is complex, or
10.1	c. the expression is not constant and is a function argument, or
	d. the expression is not constant and is a return expression.
10.3	The value of a complex expression of integer type shall only be cast to a type that is not wider and of the same signedness as the underlying type of the expression.
11.3	A cast should not be performed between a pointer type and an integral type.
11.4	A cast should not be performed between a pointer to object type and a different pointer to object type.
11.5	A cast shall not be performed that removes any const or volatile qualification from the type addressed by a pointer.
12.2	The value of an expression shall be the same under any order of evaluation that the standard permits.
12.4	The right-hand operand of a logical && or operator shall not contain side effects.
12.6	The operands of logical operators (&&, , and !) should be effectively boolean. Expressions that are effectively boolean should not be used as operands to operators other than (&&, , !, =, ==, !=, and ?).
12.13	The increment (++) and decrement (--) operators should not be mixed with other operators in an expression.
14.3	Before preprocessing, a null statement shall only occur on a line by itself; it may be followed by a comment, provided that the first character following the null statement is a whitespace character.
14.5	The continue statement shall not be used.
14.7	A function shall have a single point of exit at the end of the function.
16.1	Functions shall not be defined with a variable number of arguments.
17.4	Array indexing shall be the only allowed form of pointer arithmetic.
18.4	Unions shall not be used.
19.1	#include statements in a file should only be preceded by other preprocessor directives or comments.
19.1	In the definition of a function-like macro, each instance of a parameter shall be enclosed in parentheses unless it is used as the operand of # or ##.
20.4	Dynamic heap memory allocation shall not be used.
20.9	The input/output library <stdio.h> shall not be used in production code.

Figure 1. MISRA exceptions

8 Known Issues

8.1 P&E Micro debugger issue

Going forward, the P&E Micro OpenSDA firmware will not be shipped on any specific Freescale Freedom platform or Tower System module. A single firmware file can be used on different boards if they have the same firmware version. On FRDM-KL81Z platforms and TWR-KL81Z72M modules, to debug with P&E Micro debugger, DEBUG-FRDM-K64F_Pemicro_v108a_for_OpenSDA_v2.0.bin should be used to update the firmware because the current KL81Z MCU's OpenSDA is v2 (instead of DEBUG_OpenSDA_for_MBED_Bootloader_by_Pemicro_v108.bin, which only works with OpenSDA v1).

8.2 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 Kinetis 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.

8.3 USB HUB power supply

The external power supply of the USB HUB must be provided before it can be used. This is the result of the development board which is not designed to power a USB HUB and the devices connected to the HUB. Therefore, the external USB HUB that is connected to the development board should have its own power supply.

8.4 HS USB device MSD demo issue

If the SD card is used as the storage medium, the functionality of USB might not work correctly if the USB hot plug action is performed while transferring files.

8.5 USB noise issue

A noise occurs when a song is playing on FreeRTOS USB host speaker example. The issue is caused by the software scheduler not being aligned with the hardware scheduler as a result of the dynamic SOF threshold enablement on some boards.

9 Driver Change Log

ADC16

The current ADC16 driver version is 2.0.0

- 2.0.0
 - Initial version

CMP

The current CMP driver version is 2.0.0

- 2.0.0
 - Initial version

Driver Change Log

DAC

The current DAC driver version is 2.0.0

- 2.0.0
 - Initial version

DMAMUX

The current DMAMUX driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - Bug fix:
 - Fixed build warning while setting the DMA request source in the DMAMUX_SetSourceChange issue by changing the type of the parameter source from uint8_t to uint32_t

DSPI

The current DSPI driver version is 2.1.1

- 2.1.0
 - New features
 - Added transfer prefix to transactional APIs
- 2.1.1
 - Bug fix:
 - Set the EOQ (End Of Queue) bit to TRUE for the last transfer in transactional APIs

eDMA

The current eDMA driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - Bug fix:
 - Fixed the issue where an eDMA callback does not check a valid status in the EDMA_HandleIRQ API

FlexIO UART

The current FlexIO UART driver version is 2.1.1

- 2.1.0
 - New features:
 - Added a transfer prefix to the transactional APIs
 - Added the txSize/rxSize parameters to the handle structure to record the transfer size
 - Bug fix
 - Added an error handle to handle the data count zero or data buffer NULL situation
- 2.1.1
 - Bug fix
 - Renamed FLEXIO_UART_StopRingBuffer to FLEXIO_UART_TransferStopRingBuffer to align with the definition in the C file

FlexIO I2C

The current FlexIO I2C driver version is 2.1.1

- 2.1.0
 - New features:

- Added a transfer prefix to the transactional APIs
 - Added the txSize/rxSize parameters to the handle structure to record the transfer size
- Bug fix
 - Added an error handle to handle the data count zero or data buffer NULL situation
- 2.1.1
 - Bug fix
 - Implemented the FLEXIO_I2C_MasterTransferBlocking API which is defined in the header file but has no implementation in the C file

FlexIO SPI

The current FlexIO SPI driver version is 2.1.0

- 2.1.0
 - New features:
 - Added a transfer prefix to the transactional APIs
 - Added the txSize/rxSize parameters to the handle structure to record the transfer size
 - Bug fix
 - Fixed the error register address return for 16-bit data write in the FLEXIO_SPI_GetTxDataRegisterAddress

FlexIO I2S

The current FlexIO I2S driver version is 2.1.1

- 2.1.0
 - New features:
 - Added a transfer prefix to the transactional APIs
 - Added the txSize/rxSize parameters to the handle structure to record the transfer size
- 2.1.1
 - Bug fix
 - Fixed the FlexIO I2S RX data read error and eDMA address error
 - Fixed the FlexIO I2S slave timer comparison setting error

GPIO

The current GPIO driver version is 2.1.0

- 2.1.0
 - API Interface Change:
 - Added "pins" or "pin" to some API names
 - Renamed the "GPIO_PinConfigure" to "GPIO_PinInit"

I2C

The current I2C driver version is 2.0.1

- 2.0.1
 - New features
 - Added a double buffer enable configuration for SoCs which have the DFEN bit in S2 register
 - Added the flexible transmit/receive buffer size support in I2C_SlaveHandleIRQ
 - Added the start flag clear address match and release bus operation in I2C_SlaveWrite/ReadBlocking API
 - Bug fix:
 - Updated the kI2C_SlaveRepeatedStartEvent to kI2C_SlaveStartEvent

INTMUX

The current INTMUX driver version is 2.0.0

- 2.0.0
 - Initial version

Driver Change Log

LLWU

The current LLWU driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - Changes:
 - Updated for KL8x

LPTMR

The current LPTMR driver version is 2.0.0

- 2.0.0
 - Initial version

LPUART

The current LPUART driver version is 2.2.1

- 2.1.0
 - Updated transactional APIs
- 2.1.1
 - Removed the needless check of event flags and assert in LPUART_RTOS_Receive
 - Wait always for RX event flag in LPUART_RTOS_Receive
- 2.2.0
 - Added seven data bits and MSB support
- 2.2.1
 - Added a separate RX and TX IRQ number support

LTC

The current LTC driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - Bug fix:
 - Fixed the C++ build warning in block_to_ififo() functions

MPU

The current MPU driver version is 2.1.0

- 2.0.0
 - Initial version
- 2.1.0
 - API updates:
 - Updated the mpu_region_num_t and mpu_master_t to uint32_t
 - Updated the mpu_low_masters_access_rights_t and mpu_high_masters_access_rights_t to mpu_rwxrights_master_access_control_t and mpu_rwrights_master_access_control_t
 - Updated the MPU_SetRegionLowMasterAccessRights() and MPU_SetRegionHighMasterAccessRights() to MPU_SetRegionRwxMasterAccessRights() and MPU_SetRegionRwMasterAccessRights()

PIT

The current PIT driver version is 2.0.0

- 2.0.0
 - Initial version

PMC

The current PMC driver version is 2.0.0

- 2.0.0
 - Initial version

PORT

The current PORT driver version is 2.0.1

- 2.0.1
 - Changes:
 - Added "const" in function parameters
 - Updated enumeration variable names

QSPI

The current QSPI driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - New API:
 - QSPI_SetReadArea to set the read area
 - Bug fix:
 - Fixed the QSPI_UpdateLUT function only updating the first LUT issue
 - Fixed an issue where some functions hard code the QSPI0 as the base

RCM

The current RCM driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - [KPSDK-10249] Fixed the kRCM_SourceSw bit shift issue.

RTC

The current RTC driver version is 2.0.0

- 2.0.0
 - Initial version

SIM

The current SIM driver version is 2.0.0

- 2.0.0
 - Initial version

Smart Card

The current Smart Card driver version is 2.1.1

- 2.1.0
 - Initial version
- 2.1.1
 - New features:
 - Added a default PHY interface selection to Smart Card RTOS drivers (KPSDK-9063)
 - Replaced the smartcard_phy_ncn8025 driver with the smartcard_phy_tda8035
 - Bug fix

Driver Change Log

- Fixed the protocol timers activation sequences in the smartcard_emvsim and smartcard_phy_tda8035 drivers during the EMVL1 pre-certification tests (KPSDK-9170, KPSDK-9556)

SMC

The current SMC driver version is 2.0.2

- 2.0.0
 - Initial version
- 2.0.1
 - Changes:
 - Updated for KL8x
- 2.0.2
 - Bug fix:
 - Added DSB before WFI and ISB after WFI
 - Changes:
 - Updated the SMC_SetPowerModeVlpw implementation

TPM

The current TPM driver version is 2.0.2

- 2.0.0
 - Initial version
- 2.0.1
 - Bug fix
 - Fixed the TPM_UpdateChnEdgeLevelSelect ACK wait issue
 - Fixed the TPM_SetupDualEdgeCapture can not set FILTER register issue
 - Fixed the TPM_UpdateChnEdgeLevelSelect ACK wait issue
- 2.0.2
 - Bug fix
 - Fixed wait acknowledgement when channel is disabled issues in functions TPM_SetupPwm, TPM_UpdateChnEdgeLevelSelect, TPM_SetupInputCapture, TPM_SetupOutputCompare, and TPM_SetupDualEdgeCapture

TRNG

The current TRNG driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - New features:
 - Set a default OSCDIV for Kinetis devices KL8x and KL28Z
 - Changes:
 - Changed a default OSCDIV for KL81Z to divide by 2

VREF

The current VREF driver version is 2.1.0

- 2.1.0
 - Added new functions:
 - Added VREF_SetTrim2V1Val() and VREF_GetTrim2V1Val() functions to supply 2V1 output mode.

WDOG

The current WDOG driver version is 2.0.0

- 2.0.0
 - Initial version

CLOCK

The current CLOCK driver version is 2.2.0

- 2.0.0
 - Initial version
- 2.1.0
 - Changes:
 - Merged the fsl_mcg and fsl_osc into fsl_clock
- 2.2.0
 - New features:
 - [KPSDK-9157] Updated the CLOCK_SetFeiMode/CLOCK_SetFbiMode/CLOCK_BootToFeiMode() to support set MCG_C4[DMX32]=1 in FEI/FBI modes
 - Bug fix:
 - Updated the IP_CLOCKS array, removed unused gates, and added missing gates

10 Middleware Change Log

DMA Manager for KSDK

The current DMA Manager driver version is 2.0.0

- 2.0.0
 - Initial version

EMVL1 for KSDK

The current EMVL1 driver version is 2.0.1

- 2.0.0
 - Initial version
- 2.0.1
 - Bug fix:
 - Fixed low level driver protocol timer failures during EMVL1 pre-certification tests (KPSDK-9556)
 - Fixed incorrect T0 command response that causes long command responses (KPSDK-8707). Command case2, case3, and case4 are affected

FatFs for KSDK

The current FatFs driver version is R0.11a

- Added glue functions for low level drivers (SDHC, SDSPI, RAM, and MMC) and modified the diskio.c file
- Added RTOS wrappers to make FatFs thread-safe. Modified the syscall.c file
- Renamed ffconf.h file to ffconf_template.h file. Each application should contain its own ffconf.h file
- Included ffconf.h into diskio.c files to enable selection of a physical disk from the ffconf.h by macro definition
- Conditional compilation of physical disk interfaces in diskio.c

mbedtls for KSDK

The current mbedtls driver version is based on the mbedtls 2.2.1 released on 2016-01-05

- 2.2.1
 - New features:
 - Ported mbedtls 2.2.1 to KSDK 2.0.0
 - Added support for the mmCAU cryptographic acceleration module. Accelerated MD5, SHA, AES, and DES
 - Added support for the LTC cryptographic acceleration module. Accelerated AES, DES, and PKHA
 - Added new files:

- .c - alternative implementation of cryptographic algorithm functions using LTC and mmCAU module drivers
- .h - configuration settings used by mbedTLS KSDK bare metal examples
- Added mbedTLS KSDK bare metal examples:
 - <board name> - KSDK mbedTLS benchmark application
 - <board name> - KSDK mbedTLS self-test application
- Added the MBEDTLS_GCM_CRYPT_ALT configuration parameter to enable reloading the mbedtls_gcm_crypt_and_tag() function
- Added the MBEDTLS_ECP_MUL_COMB_ALT to enable an alternate implementation of the ecp_mul_comb() function
- Added the MBEDTLS_ECP_ADD_ALT configuration parameter to enable reloading the ecp_add() function
- Added the MBEDTLS_DES_SETKEY_DEC_ALT configuration parameter to enable reloading mbedtls_des_setkey_dec(), mbedtls_des3_set2key_dec(), and mbedtls_des3_set3key_dec() functions
- Added the MBEDTLS_DES_SETKEY_ENC_ALT configuration parameter to enable reloading mbedtls_des_setkey_enc(), mbedtls_des3_set2key_enc(), and mbedtls_des3_set3key_enc() functions
- Added the MBEDTLS_DES_CRYPT_CBC_ALT configuration parameter to enable reloading the mbedtls_des_crypt_cbc() function
- Added the MBEDTLS_DES3_CRYPT_CBC_ALT configuration parameter to enable reloading the mbedtls_des3_crypt_cbc() function
- Added the MBEDTLS_AES_CRYPT_CBC_ALT configuration parameter to enable reloading the mbedtls_aes_crypt_cbc() function
- Added the MBEDTLS_AES_CRYPT_CTR_ALT configuration parameter to enable reloading the mbedtls_aes_crypt_ctr() function
- Added the MBEDTLS_CCM_CRYPT_ALT configuration parameter to enable reloading mbedtls_ccm_encrypt_and_tag() and mbedtls_ccm_auth_decrypt() functions
- Added the MBEDTLS_MPI_ADD_ABS_ALT configuration parameter to enable reloading the mbedtls_mpi_add_abs() function
- Added the MBEDTLS_MPI_SUB_ABS_ALT configuration parameter to enable reloading the mbedtls_mpi_sub_abs() function
- Added the MBEDTLS_MPI_EXP_MOD_ALT configuration parameter to enable reloading the mbedtls_mpi_exp_mod() function
- Added the MBEDTLS_MPI_MUL_MPI_ALT configuration parameter to enable reloading the mbedtls_mpi_mul_mpi() function
- Added the MBEDTLS_MPI_MOD_MPI_ALT configuration parameter to enable reloading the mbedtls_mpi_mod_mpi() function
- Added the MBEDTLS_MPI_GCD_ALT configuration parameter to enable reloading the mbedtls_mpi_gcd() function
- Added the MBEDTLS_MPI_INV_MOD_ALT configuration parameter to enable reloading the mbedtls_mpi_inv_mod() function
- Added the MBEDTLS_MPI_IS_PRIME_ALT configuration parameter to enable reloading the mbedtls_mpi_is_prime() function
- Added encrypt/decrypt modes to the mbedtls_des_context and the mbedtls_des3_context structure
- Added a carriage return to the mbedtls_printf() in self test functions

SDMMC for KSDK

The current SDMMC driver version is 2.1.1

- 2.1.0
 - Bug fix:
 - Changed the callback mechanism when sending a command
 - Fixed the performance low issue when transferring data
 - Changes:
 - Changed the name of error codes returned by an internal function

- Merged all host-related attributes to one structure
- Optimized the function to set a maximum data bus width for the MMC card
- 2.1.1
 - Bug fix:
 - Fixed the block range boundary error when transferring data to the MMC card
 - Fixed the bit mask error in the SD card when switching to a high-speed function
 - Changes:
 - Added an error code to indicate that SDHC ADMA1 transfer type is not supported
 - Optimized the SD card initialization function

USB stack for KSDK

The current USB stack version is 1.1.0

- 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
 - vusb_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

RTOS Change Log

- 1.0.1
 - Bug fix
 - Improved the device audio speaker efficiency by changing the transfer mode from interrupt to DMA to eliminate the periodic noise
- 1.1.0
 - Bug fix:
 - Fixed issues in the USB certification
 - Updated the VID and Manufacturer string to NXP Semiconductors
 - New features:
 - Supported classes
 - Printer
 - Examples
 - usb_device_composite_cdc_msc_sdcard
 - usb_device_printer_virtual_plain_text
 - usb_host_printer_plain_text
 - Changes:
 - Renamed example usb_device_composite_cdc_msc to usb_device_composite_cdc_msc_ramdisk

wolfSSL for KSDK

The current wolfSSL is 3.8.0 based on the release 3.8.0 of wolfSSL

- 3.8.0
 - New features:
 - Added support for the Kinetis LTC hardware acceleration module, which accelerates AES, 3DES, TFM module (modular integer arithmetic), and ECC wolfSSL modules
 - Added support for the Kinetis random number generator modules TRNG and RNGA.
 - Miscellaneous changes:
 - The Kinetis mmCAU acceleration now uses the "fsl_mmcau.h" file instead of the "cau_api.h" file
 - In DSA, wc_dsaSign() function is updated to repeat the wc_RNG_GenerateBlock() until k is less than q
 - wolfssl/wolfcrypt/settings.h file is changed to remove the unused macros and add support for the KSDK 2.0.0
 - In the wolfcrypt/src/asn.c file, the ksdk_time(time_t) is changed to external, to be defined by an application

11 RTOS Change Log

FreeRTOS OS for KSDK

The current version is FreeRTOS OS 8.2.3. The original package is available at freertos.org.

- 8.2.3
 - New features:
 - Added tickless idle mode support
 - Added a template application for Kinetis Expert (KEx) tool (template_application)
 - Changes:
 - Reduced the folder structure to keep only Kinetis-related information

μC/OS-II OS for KSDK

The current version is μC/OS-II OS V2.92.11

- 2.92.11
 - New features:
 - Added a template application for the Kinetis Expert (KEx) tool (template_application)
 - Changes:

- Reduced the folder structure to keep only Kinetis-related information
- Added wrappers to adapt PendSV_Handler and SysTick_Handler. Related files are located in rtos\ucosii\uCOS-II\Ports\ARM-Cortex-Mx\Generic\<compiler>\fsl_isr_wrapper.S

μC/OS-III OS for KSDK

The current version is μC/OS-III OS V3.05.01

- V3.05.01
 - New features:
 - Added a template application for the Kinetis Expert (KEx) tool (template_application)
 - Bug fix:
 - [KPSDK-7247] Downgraded port files from V3.05.01 to V3.05.00 because of the context switch issue
 - Changes:
 - Reduced the folder structure to keep only Kinetis-related information

12 Revision History

This table summarizes revisions to this document.

Table 4. Revision history

Revision number	Date	Substantive changes
0	06/2016	Initial release

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.

NXP, the NXP logo, Freescale, the Freescale logo, Tower, and Kinetis are trademarks of NXP B.V. All other product or service names are the property of their respective owners. ARM, ARM powered logo, Keil, μ Vision, and Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. mbed is a trademark of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.

© Freescale Semiconductor, Inc.

Document Number KSDKKL82200RN
Revision 0, 06/2016

