NXP Semiconductors
Release Notes

Document identifier: MCUXSDKKW38RN
Rev. 0, May 2020

MCUXpresso SDK Release Notes for MKW39-38-37

Contents

Chapter 1
Overview

The MCUXpresso Software Development Kit (SDK) is a collection of software enablement for microcontrollers that includes peripheral drivers, high-level stacks including other middleware packages, and integrated RTOS support for FreeRTOS™ OS. In addition to the base enablement, the MCUXpresso SDK is augmented with demo applications, 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 the MCUXpresso SDK homepage MCUXpresso-SDK: Software Development Kit.

Chapter 2
MCUXpresso SDK

As part of the MCUXpresso software and tools, MCUXpresso SDK is the evolution of Kinetis SDK v2.x.x and includes support for both LPC and i.MX System-on-Chips (SoC). The same drivers, APIs, and middleware are still available with support for the 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, the support for the MCUXpresso Config Tools enables 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

To maintain compatibility with legacy Freescale code, the filenames and source code in the MCUXpresso SDK contain the legacy Freescale prefix "FSL". The "FSL" prefix is redefined as the NXP Foundation Software Library.

Chapter 3
Development tools

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

- IAR Embedded Workbench for Arm version 8.50.1
- MCUXpresso IDE version 11.1.1_3241

Chapter 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
FRDM-KW38	**MKW38A512VFT4**, MKW38Z512VFT4, MKW39A512VFT4
USB-KW38	MKW37Z512VFT4, MKW37A512VFT4

Chapter 5
What is new

- **MCUXpresso SDK Release 2.6.5 KW37A SDK RFP**

  This new release package for the KW37/38/39 platforms corresponds to the Release For Production phase of the program.

  The KW37 Bluetooth® LE v5.0 features in this release have undergone a Bluetooth SIG qualification listing process, as follows:

  — **Controller Subsystem Bluetooth LE 5** https://launchstudio.bluetooth.com/ListingDetails/102295

  — **Host Subsystem Bluetooth LE 5** https://launchstudio.bluetooth.com/ListingDetails/103280

- **Wireless**

  — Versions:

middleware/wireless/framework	release_fwk_ksdk_2.6_kw37a_5.7.6
middleware/wireless/genfsk	release_genfsk_ksdk_2.6_kw37a_3.0.3
middleware/wireless/ble_controller	release_ll_ksdk_2.6_kw37a_1.6.6
middleware/wireless/bluetooth	release_ble_ksdk_2.6_kw37a_1.6.6
middleware/wireless/refdes	release_refdes_ksdk_2.6_kw37a_1.0.6

Below are notable updates for this KW39/38/37 release as compared to previous MCUXpresso SDK releases supporting KW36/35/34:

- **Bluetooth LE Host Stack and Applications**
  - Added support for Bluetooth 5.0 Advertising extensions Optional features
  - Updated Bluetooth LE Applications to support Extended Advertising features
  - Code and RAM memory optimizations in Bluetooth LE Host Stack modules.
  - Added Advertising Extended Central and Peripheral Applications
  - Added VLLS support for Bluetooth low power applications
  - Updated Bluetooth LE GenFSK with support for extended advertising and low power
  - MISRA 2012: Compliant (Bluetooth LE Host Private files)
  - Static Code Check: Compliant (Bluetooth LE Host Private files)
  - Cyclomatic Complexity (HIS CCM <= 30): Compliant (Bluetooth LE Host Private files)
  - Unitary Testing Coverage
  - Added USB-KW38 projects
    - `Wireless_uart`, `BLE_Shell`, `OTAP_Server`, `Hid_Host`, `BLE_Fsci_Black_box`
  - Added Bluetooth LE Host Libraries build with GCC for MCUXpresso IDE projects
  - Bug fixing
- **SweynTooth Research Group Published Vulnerabilities** (https://asset-group.github.io/disclosures/sweyntooth/)
  - Link Layer Length Overflow (CVE-2019-17519): Fixed
  - LLID Deadlock (CVE-2019-17060): Fixed
- **Bluetooth LE 5.0 Link Layer**

- MISRA 2012: Compliant
- Static Code Check: Compliant
- Cyclomatic Complexity (HIS CCM <= 30): Compliant
- SweynTooth Research Group Published Vulnerabilities (https://asset-group.github.io/disclosures/sweyntooth/): Robustness verified
- Standard Supported Features:
  - LE Encryption
  - LE Ping Procedure
  - Connection Parameter Request Procedure
  - Extended Reject Indication
  - Slave-initiated Features Exchange
  - LE Data Packet Length Extension
  - Multiple PHYs
  - LE 2M PHY
  - Asymmetric Connections
  - LE Coded PHY
  - LE Extended Advertising
  - LE Periodic Advertising
  - Channel Selection Algorithm #2
  - LE Power Class 1
  - Minimum Number of Used Channels Procedure
  - LL Privacy
- Specific Supported Features:
  - Eight connections as Master role
  - Eight connections as Slave role
  - Any combination of Master/Slave roles (max connections number: 8)
  - Two Advertising Sets in parallel
  - 26 Whitelist entries
  - Eight RPA entries
- Controller States/Roles:
  - Advertising State
  - Scanning State
  - Initiating State
  - Slave Role
  - Master Role
- Device Addresses Types:
  - Public/Random/Static Addresses
  - Generation/Resolution of private addresses

    - Network Privacy Mode
    - Device Privacy Mode
- Protocol Advertising Features:
    - Low Duty Cycle Direct Advertising
    - Filtering Policies
    - Extended Advertising
    - Periodic Advertising
    - Multiple Advertising Sets
    - Sending Tx Power in Advertisements
- Protocol Scanning Features:
    - Extended Scanner filter policies
    - Periodic Sync Establishment Filtering Policies
    - Extended Scanning
    - Scanning for Periodic Advertising
- Protocol Initiating Features:
    - Requesting connections using extended advertising
- Protocol Slave Role Features:
    - Initiating Connection Parameter Request
    - Encryption Start
    - Slave Pause Encryption
    - Slave Version Exchange
    - Slave listens to multiple packets per connection event
    - LE Authenticated Payload Timeout
    - Data Length Update Procedure
    - Minimum Number Of Used Channels Procedure
    - PHY Update procedure
- Protocol Master Role Features:
    - Responding in Feature Setup
    - Initiating Connection Parameter Request
    - Accepting Connection Parameter Request
    - Encryption Start
    - Master Pause Encryption
    - Master Version Exchange
    - Minimum Number Of Used Channels Procedure
    - PHY Update procedure
- Scatternet Capabilities:
    - Act as LE Master and LE Slave at the same time
    - Act as LE Slave to more than one LE Master at the same time

- **GenFSK**
  - Radio operation using custom GFSK, FSK or MSK modulation formats.
  - Highly configurable packet structure, variable bit rate, variable lengths, bit ordering and contents of individual packets fields, defining the start and end points for whitening, CRC and some primitive parsing of the packet header.
  - RAW packet transfer mode (with only preamble detect and network address match available).
  - Instantiable, enabling use of several configurations in the same applications.
  - MISRA 2012: Compliant
  - Static Code Check: Compliant
  - Cyclomatic Complexity (HIS CCM <= 30) : Compliant
- **Framework**
  - Provides set of different services (timers, serial interfaces, OS abstraction, memory management, and others)
  - Low Power (base features):
    - Support of STOP, clock gated (LLS2/3), power gated (VLLS2/3), power gated without RAM retention (VLLS0/1) modes
    - Support of DCDC BUCK, BYPASS and PSWITCH modes
    - Support of 16k/32k/total RAM retention in low power
    - Support of advertising on wake-up
      - Support of advertising intervals larger than 10.24 seconds (using timers + advertising on wake-up)
    - Support of advertising with RAM off between advertising events (for evaluation purpose)
    - Support of FreeRTOS tickless mode
    - Support of GenFSK in single burst
    - Support of different wake-up sources (IO, low power timers, UART)
  - New memory manager: MemManagerLight
    - Dynamically allocates memory blocks in the heap
    - Blocks placed on lower memory addresses to optimize RAM retention (16 k/32 k/all)
  - Various optimizations (clocks, wake-up time)
  - MISRA 2012: Compliant
  - Static Code Check: Compliant
  - Cyclomatic Complexity: Compliant
- **Low Power Reference Design Applicatins**
  - Two new reference design applications with low power
    - `lp`: Bluetooth LE advertiser, acts like a temperature/battery sensor
    - `lp_master`: Bluetooth LE scanner, retrieves temperature/battery
  - Fully optimized low power implementation
  - Supports of all low power features based on different configuration
  - Reference for low power porting on an application
  - High verbosity to describe configuration flags
- **Bluetooth LE 5.0 Sniffer (running on USB-KW38)**

- All Bluetooth LE5.0 mandatory features supported
- Multiple PHYs
- LE 2 M PHY
- LE Coded PHY (S=2, S=8)
- LE Extended Advertising
- LE Periodic Advertising

Chapter 6
Release contents

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

Table 2. Release contents

Deliverable	Location
Bluetooth	<install_dir>/middleware/wireless/bluetooth
Bluetooth LE controller	<install_dir>/middleware/wireless/ble_controller
Boards	<install_dir>/boards
Cortex Microcontroller Software Interface Standard (CMSIS) Arm® Cortex®-M header files, DSP library source	<install_dir>/CMSIS
Demo applications	<install_dir>/boards/<board_name>/demo_apps
Documentation	<install_dir>/docs
Driver examples	<install_dir>/boards/<board_name>/driver_examples
Driver, SoC header files, extension header files and feature header files, utilities	<install_dir>/devices/<device_name>
GenFSK	<install_dir>/middleware/wireless/genfsk
LIN stack	<install_dir>/middleware/lin_stack
Middleware	<install_dir>/middleware
Peripheral Drivers	<install_dir>/devices/<device_name>/drivers
Reference design applications	<install_dir>/boards/<board_name>/reference_design
RTOS examples	<install_dir>/boards/<board_name>/rtos_examples
RTOS Kernel Code	<install_dir>/rtos
Tools	<install_dir>/tools
Utilities such as debug console	<install_dir>/devices/<device_name>/utilities
Wireless examples	<install_dir>/boards/<board_name>/wireless_examples
Wireless framework	<install_dir>/middleware/wireless/framework

Chapter 7
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.

7.1 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.

7.1.1 Board support

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

7.1.2 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.

7.2 Wireless

The following connectivity-supporting documentation is included in the documentation package:

- *Generic FSK Link Layer Quick Start Guide*
- *Generic FSK Link Layer API Reference Manual*
- *Bluetooth Low Energy Host Stack API Reference Manual*
- *Bluetooth Low Energy Application Developer's Guide*
- *Bluetooth Low Energy Demo Applications User's Guide*
- *Bluetooth Low Energy Host Stack FSCI Reference Manual*
- *Bluetooth Low Energy Quick Start Guide*
- *Bluetooth Low Energy Host Stack FSCI Application Programming*
- *Low Power Connectivity Design User’s Guide*
- *Connectivity Framework Reference Manual*

7.2.1 Radio

7.2.1.1 Tested radio modes and data rates

The following radio modes are tested in this release:

- Bluetooth LE 1 Mbit/s and 2 Mbit/s RX at 32 MHz and 26 MHz `RF_OSC`.
- Bluetooth LE 1 Mbit/s RX with Dirty TX Impairments at 32 MHz and 26 MHz `RF_OSC`.
- Bluetooth LE Long Range S=2 (500 Kbit/s) and S=8 (125 Kbit/s) at 32 MHz and 26 MHz `RF_OSC`.
- GFSK BT=0.5, h=0.5, 2 Mbit/s /1 Mbit/s / 500 Kbit/s / 250 Kbit/s rates at 32 MHz and 26 MHz `RF_OSC`.
- GFSK BT=0.5, h=0.32, 2 Mbit/s / 1 Mbit/s / 500 Kbit/s / 250 Kbit/s at 32 MHz and 26 MHz `RF_OSC`.
- GFSK BT=0.5, h=0.7, 1 Mbit/s / 2 Mbit/s at 32 MHz and 26 MHz `RF_OSC`.
- MSK 2 Mbit/s / 1 Mbit/s / 500 Kbit/s / 250 Kbit/s at 32 MHz and 26 MHz `RF_OSC`.
- GFSK BT=0.5, h=0.5, Long Range emulation for S=2 and S=8 coding rates at 32 MHz and 26 MHz `RF_OSC`. RX only, TX not yet tested.

7.2.1.2 Overview of deviating results

The following is the regression testing that did not pass in this release:

- N/A

The following is the regression testing that passes in this release, but still shows some level of impaired performance:

- Bluetooth LE Long Range with the 26 MHz RF oscillator at both S=2 and S=8 does not meet targeted RX Sensitivity in this release.
- GFSK 05032 is operational but has PER spikes at 250 Kbps data rate due to unoptimized receiver settings.
- MSK TX FSK error is slightly above target level.
- Selectivity at 6 MHz offset for Bluetooth 2 Mbps sporadically deviates from the target.

7.3 RTOS

The MCUXpresso SDK is integrated with FreeRTOS OS and bare metal.

Chapter 8
MISRA compliance

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

Table 3. MISRA exceptions

Exception rules	Description
Directive 4.2	This rule is Advisory and currently disabled in the analysis configuration.
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 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 be hidden.
Directive 4.9	A function should be used in preference to a function like a macro where they are interchangeable.
Directive 4.13	This rule is Advisory and currently disabled in the analysis configuration.
Rule 1.2	This rule is Advisory and currently disabled in the analysis configuration.
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	This rule is Advisory and currently disabled in the analysis configuration.
Rule 2.7	There should be no unused parameters in functions.
Rule 4.2	This rule is Advisory and currently disabled in the analysis configuration.
Rule 5.1	Modern compilers support much larger significant characters than the rule specifies.
Rule 5.4	Modern compilers support much larger significant characters than the rule specifies.
Rule 5.9	Identifiers that define objects or functions with external linkage shall be unique.
Rule 8.7	This rule is Advisory and currently disabled in the analysis configuration.
Rule 8.9	This rule is Advisory and currently disabled in the analysis configuration.
Rule 8.11	This rule is Advisory and currently disabled in the analysis configuration.
Rule 8.13	A pointer should point to a const-qualified type whenever possible.
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 an 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	This rule is Advisory and currently disabled in the analysis configuration.

Table continues on the next page...

Table 3. MISRA exceptions (continued)

Exception rules	Description
Rule 12.4	This rule is Advisory and currently disabled in the analysis configuration.
Rule 13.3	This rule is Advisory and currently disabled in the analysis configuration.
Rule 13.4	This rule is Advisory and currently disabled in the analysis configuration.
Rule 15.1	This rule is Advisory and currently disabled in the analysis configuration.
Rule 15.4	This rule is Advisory and currently disabled in the analysis configuration.
Rule 15.5	This rule is Advisory and currently disabled in the analysis configuration.
Rule 17.5	This rule is Advisory and currently disabled in the analysis configuration.
Rule 17.8	This rule is Advisory and currently disabled in the analysis configuration.
Rule 18.4	This rule is Advisory and currently disabled in the analysis configuration.
Rule 18.5	This rule is Advisory and currently disabled in the analysis configuration.
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	Our coding standards allow for the use of leading underscore in defining identifiers or macro names which do not conflict with identifiers or macro names in the scope of the Library section of the C standard.
Rule 21.2	Our coding standards allow for the use of leading underscore in defining identifiers or macro names which do not conflict with identifiers or macro names in the scope of the Library section of the C standard.
Rule 21.12	This rule is Advisory and currently disabled in the analysis configuration.

Chapter 9
Known issues

9.1 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.

9.2 Only FreeRTOS is tested for RTOS support

This release only supports the IAR Embedded Workbench IDE, MCUXpresso IDE, the FreeRTOS kernel, and a bare-metal non-preemptive task scheduler.

9.3 Disabled pairing and bonding for most sensor applications

Most sensor applications have pairing and bonding disabled to allow a faster interaction with mobile applications. These two security features can be enabled in the *app_preinclude.h* header file.

9.4 Toolchain functions

Toolchains functions (EX: memcpy) enabled with `gUseToolchainMemFunc_d` cannot be used with some types of optimisations (EX: IAR EWARM Speed).

9.5 Bluetooth LE extended advertising—chain packet number

When using A0 silicon version, only one Chain Packet per Advertising Set is supported. This limitation is solved when using B0 silicon version.

9.6 DTM / Rx sensitivity tests

The DTM / Rx sensitivity tests must be performed with a payload length of 37 bytes.

9.7 Bluetooth LE 5.0 Sniffer limitations (running on USB-KW38)

- One single channel decoded at a time.
- Payload decryption not supported.
- RPA not supported.
- Some blocks may be missed in case of high traffic.
- Low SPI throughput

9.8 GenFSK limitations

- BT=0.5, h=0.32 modulation is functional but has RX PER spikes present in the mid-power levels for both 26 MHz and 32 MHz RF_OSC.
- BT=0.5, h=0.5 modulation, 250 Kbps rate with 26 MHz RF_OSC is functional but has RX PER spikes present at mid-power levels.

- Fast warmup not supported (150 µS not supported between Rx and Tx).
- Sync address 1 and 2 bytes not supported.

9.9 Low power reference design applications

Features not supported :

- GenFSK Low Power modes

9.10 Device or network privacy mode configurations in resolving list

The Bluetooth LE Link Layer supports a single mode configuration (either Device or Network privacy) in the Resolving list. A mixed configuration is not supported.

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.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

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

Date of release: May 2020
Document identifier: MCUXSDKKW38RN