# Proprietary Flex SDK 3.8.3.0 GA
## Simplicity SDK Suite 2024.6.3
## April 23, 2024

The Proprietary Flex SDK is a complete software development suite for proprietary wireless applications. Per its namesake, Flex offers two implementation options.

The first uses Silicon Labs RAIL (Radio Abstraction Interface Layer), an intuitive and easily-customizable radio interface layer designed to support both proprietary and standards-based wireless protocols.

The second uses Silicon Labs Connect, an IEEE 802.15.4-based networking stack designed for customizable broad-based proprietary wireless networking solutions that require low power consumption and operates in either the sub-GHz or 2.4 GHz frequency bands. The solution is targeted towards simple network topologies.

The Flex SDK is supplied with extensive documentation and sample applications. All examples are provided in source code within the Flex SDK sample applications.

These release notes cover SDK version(s):

3.8.3.0 GA released April 23, 2025.
3.8.2.0 GA released September 18, 2024.
3.8.1.0 GA released July 24, 2024.
3.8.0.0 GA released June 5, 2024.

### Proprietary

**RAIL APPS AND LIBRARY KEY FEATURES**

- Flex-RAIL SDK support for EFR32xG22E Devices
- LCD Display usage in WM-BUS Meter Sample Application on DK2600
- Collision Detection support in RAIL Library in concurrent mode on FG25
- Added support for additional Coex TX and RX metrics events to RAIL coexistence utility
- Updated some RAIL APIs to better prepare for future API changes while retaining backwards compile-time compatibility

**CONNECT APPS AND STACK KEY FEATURES**

- Flex-Connect SDK support for EFR32xG22E Devices
- Removed support for Series 0/1

## Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Platform Release Notes installed with this SDK or on the TECH DOCS tab on https://www.silabs.com/developers/flex-sdk-connect-networking-stack. Silicon Labs also strongly recommends that you subscribe to Security Advisories for up-to-date information. For instructions, or if you are new to the Silicon Labs Flex SDK, see Using This Release.

**Compatible Compilers:**

IAR Embedded Workbench for ARM (IAR-EWARM) version 9.40.1

- Using wine to build with the IarBuild.exe command line utility or IAR Embedded Workbench GUI on macOS or Linux could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.
- Customers on macOS or Linux are advised not to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

GCC (The GNU Compiler Collection) version 12.2.1, provided with Simplicity Studio.

# Contents

# 1   Connect Applications

## 1.1    New Items

**Added in release 3.8.0.0**

- Support for EFR32xG22E Devices

## 1.2    Improvements

**Changed in release 3.8.0.0**

None.

## 1.3    Fixed Issues

**Fixed in release 3.8.0.0**

None.

## 1.4    Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on the TECH DOCS tab on https://www.silabs.com/developers/flex-sdk-connect-networking-stack.

| ID # | Description | Workaround |
|---|---|---|
| 652925 | EFR32XG21 is not supported for "Flex (Connect) - SoC Light Example DMP" and "Flex (Connect) - SoC Switch Example" | |
| 1139850 | DMP instabilities with XG27 | |

## 1.5    Deprecated Items

**Deprecated in release 3.8.0.0**

Flex SDK Flex folder is deprecated and will be removed. It has been split into Rail folder for RAIL SDK and Connect folder for Connect SDK..

## 1.6    Removed Items

**Removed in release 3.8.0.0**

None.

# 2 Connect Stack

## 2.1 New Items

### Added in release 3.8.0.0

- Added a new API `emberSetUnencryptedPacketsAcceptance()` that can be called to enable the filtering of unencrypted data packets. If enabled, all unencrypted data will be ignored by the stack and won't be visible from the application layer.

## 2.2 Improvements

### Changed in release 3.8.0.0

None.

## 2.3 Fixed Issues

### Fixed in release 3.8.2.0

| ID # | Description |
|------|-------------|
| 1334048 | Fixed an NVM issue where the Child table information was not saved through reboots. |

### Fixed in release 3.8.1.0

| ID # | Description |
|------|-------------|
| 1301334 | Fixed a memory leak when waking up from EM2. A buffer is allocated during the mac initialization and wake-up phases. It should only be allocated during initialization. |

### Fixed in release 3.8.0.0

| ID # | Description |
|------|-------------|
| 1294620 | Called Silicon Labs thread-safe malloc wrapper instead of the standard malloc. Current dynamic memory allocation could cause memory corruption. |

## 2.4 Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on the TECH DOCS tab on https://www.silabs.com/developers/gecko-software-development-kit.

| ID # | Description | Workaround |
|------|-------------|------------|
|  | When running the RAIL Multiprotocol Library (used for example when running DMP Connect+BLE), IR Calibration is not performed because of a known issue in the RAIL Multiprotocol Library. As result, there is an RX sensitivity loss in the order of 3 or 4 dBm. |  |
| 501561 | In the Legacy HAL component, the PA configuration is hard-coded regardless of the user or board settings. | Until this is changed to properly pull from the configuration header, the file ember-phy.c in the user's project will need to be modified by hand to reflect the desired PA mode, voltage, and ramp time. |

| ID # | Description | Workaround |
|------|-------------|------------|
| 711804 | Connecting multiple devices simultaneously may fail with a timeout error. | |

## 2.5    Deprecated Items

**Deprecated in release 3.8.0.0**

None

## 2.6    Removed Items

**Removed in release 3.8.0.0**

None

# 3   RAIL Applications

## 3.1     New Items

**Added in release 3.8.0.0**

- Support for EFR32xG22E Devices
- RAIL - SoC Wireless M-bus Meter:
    - CLI support
    - Three different sensors can be used:
        - Wireless M-bus virtual water meter sensor
        - Wireless M-bus thermo meter sensor
        - Wireless M-bus button pulse counter
    - DK2600 7-segment LCD support

## 3.2     Improvements

**Changed in release 3.8.0.0**

- RAIL - SoC Range Test BLE and IEEE802.15.4: Channels number can be set real-time both for the BLE and the 15.4 PHYs according to standard.
- RAIL - SoC Mode Switch: Mode switch can be triggered by BTN1.
- RAIL - SoC Wireless M-bus Meter improved timing accuracy

## 3.3     Fixed Issues

**Fixed in release 3.8.2.0**

| ID # | Description |
|---|---|
| 1336210 | Fixed a condition that made the RAIL - SoC Range Test BLE and IEEE802.15.4 start at a wrong channel when selecting the IEEE standard phy. |
| 1334642 | Fixed a run time assert, when default channel setting wasn't in the correct range. |
| 1332679 | Fixed Flex - RAIL PRS Support component to be able to add to every RAIL SDK sample application |
| 1322792 | Added back the Flex - RAIL Power Manager Sleep component for better sleep timing values. |

## 3.4   Known Issues in the Current Release

| ID # | Description | Workaround |
|---|---|---|
| 1268208 1268301 | The power consumption of *DK2600* can't be optimized if configuration time for EM2 if UART is being used, as the value of *SL_IOSTREAM_USART_{instance}_BAUDRATE* doesn't take effect. The default value used instead is 115200 bps. Typical use case is RAIL - SoC Wireless M-bus Meter with DK2600 board. | Update UART baud rate to 9600 bps at Simplicity Studio Admin Console. This is necessary for EM2. |

## 3.5  Deprecated Items

**Deprecated in release 3.8.0.0**

Flex SDK Flex folder is deprecated and will be removed. It has been split into Rail folder for RAIL SDK and Connect folder for Connect SDK..

## 3.6  Removed Items

**Removed in release 3.8.0.0**

None.

# 4 RAIL Library

## 4.1 New Items

**Added in release 3.8.3.0**

None.

**Added in release 3.8.2.0**

None.

**Added in release 3.8.1.0**

- Added support of automatic LNA bypass on the EFR32xG25 parts.

**Added in release 3.8.0.0**

- Added support of collision detection for concurrent PHYs on the EFR32xG25 parts.
- Added support for additional Coex TX and RX metrics events to the "RAIL Utility, Coexistence" component.

## 4.2 Improvements

**Changed in release 3.8.3.0**

- Improved the automatic LNA bypass feature on the EFR32xG25 part. Now, bypass signal is routed on PRS to work with other PRS signals. RAIL_EnableAutoLnaBypass(), RAIL_AutoLnaBypassConfig_t, RAIL_SupportsAutoLnaBypass(), and RAIL_SUPPORTS_AUTO_LNA_BYPASS have been renamed to RAIL_EnablePrsLnaBypass(), RAIL_PrsLnaBypassConfig_t, RAIL_SupportsPrsLnaBypass(), and RAIL_SUPPORTS_PRS_LNA_BYPASS.

**Changed in release 3.8.2.0**

- Reduced the delay between TX completion and the start of PA ramp down on the EFR32xG24.

**Changed in release 3.8.1.0**

None.

**Changed in release 3.8.0.0**

- Updated a few APIs to return RAIL_Status_t which were previously returning void.
- Updated RAIL_IEEE802154_WriteEnhAck and RAIL_WriteAutoAckFifo() which formerly took uint8_t ackDataLen parameter now take uint16_t ackDataLen parameter.
- Added missing runtime equivalent API RAIL_SupportsPathDiversity() of RAIL_SUPPORTS_PATH_DIVERSITY.

## 4.3 Fixed Issues

**Fixed in release 3.8.3.0**

| ID # | Description |
|---|---|
| 1315679 | Fixed a multiprotocol issue where a custom sync word set for one channel could override another channel's sync word after a DMP switch back to that protocol. This change reverts earlier fix attempts by documenting this limitation: if multiple protocols share the same radio configuration, the setting of custom sync words in any of these protocols should be avoided, as they can again leak into another protocol sharing the same radio configuration. |
| 1377956 | Fixed a DMP issue where the device would incorrectly switch to a background receive task with insufficient time to switch back to an upcoming high priority scheduled task. |
| 1380153 | Fixed an issue where calling RAIL_Idle() can prevent any future reception on the EFR32xG25 for OFDM-FSK concurrent PHYs. |

| | |
|---|---|
| 1389592 | Fixed an issue in "RAIL Utility, Coexistence" component where enabling Grant, Priority, and Request signals would cause hard fault in Matter + BLE multiprotocol application. |
| 1412336 | Fixed an issue where, on rare occasions, upon waking from sleep a RAIL_ASSERT_INVALID_XTAL_FREQUENCY assert would occur due to an incorrect reading of the system clock. |

**Fixed in release 3.8.2.0**

| ID # | Description |
|---|---|
| 1276280 | Fixed a DMP scheduling issue where tasks were scheduled to run with inadequate time to complete before another pending task. |
| 1306597 | Fixed a CCA issue on EFR32xG28 where, in some cases, the channel could be reported as busy even though it was free. |

**Fixed in release 3.8.1.0**

| ID # | Description |
|---|---|
| 1271542 | Fixed an issue on EFR32xG21 where an innopportunely-timed RAIL_StopTx() using RAIL_STOP_MODE_ACTIVE can hang a subsequent CSMA/LBT transmit during its clear-channel assessment (CCA). |

**Fixed in release 3.8.0.0**

| ID # | Description |
|---|---|
| 1242662 | Fixed incorrect default values for SL_RAIL_UTIL_PA_CALIBRATION_ENABLE on the EFR32xG24 and EFR32xG25 parts, along with an issue where Power Amplifier calibrations were not properly disabled when SL_RAIL_UTIL_PA_CALIBRATION_ENABLE was disabled. |
| 1255524 | Fixed a dynamic multiprotocol issue where RAIL_BLE_Init() was called (which implicitly establishes the BLE 1Mbps PHY), but after a protocol switch returns to BLE, its RAIL_BLE_ConfigChannelRadioParams() settings were not being properly reapplied. |
| 1256045 | Fixed an issue where Signal Identifier options were being errantly displayed in Simplicity Studio for the "RAIL Utility, Coexistence" component on platforms that don't support the Signal Identifier feature. |
| 1265376 | Fixed an issue in the UC components where the RAIL power manager support could be initialized after the radio was started in certain example applications. This would lead to radio state and power mode requirements being improperly tracked, as you must initialize power manager support before any radio operations. |
| 1266003 | Fixed an issue where RAIL_BLE_ConfigChannelRadioParams() was improperly overriding the BLE Quuppa PHY whitening configuration. |

## 4.4 Known Issues in the Current Release

Issues in bold were added since the previous release.

| ID # | Description | Workaround |
|---|---|---|
| | Using direct mode (or IQ) functionality on EFR32xG23 requires a specifically set radio configuration that is not yet supported by the radio configurator. For these requirements, reach out to technical support who could provide that configuration based on your specification | |
| 641705 | Infinite receive operations where the frame's fixed length is set to 0 are not working correctly on the EFR32xG23 series chips. | |
| 732659 | On EFR32xG23:<br>Wi-SUN FSK mode 1a exhibits a PER floor with frequency offsets around ± 8 to 10 KHz<br> Wi-SUN FSK mode 1b exhibits a PER floor with frequency offsets around ± 18 to 20 KHz | |

## 4.5 Deprecated Items

- The RAIL 2.x API is planned for deprecation in the 24Q4-GA release (June 2025). At that time, the new RAIL 3.0 API will be released for all supported chips along with a RAIL 2.x compatibility layer and migration guide.
    - The goal of this new API is to get rid of some unused features, add better support for concurrent listening use cases, and to simplify channel and PA configurations.
    - The migration is intended to be straight forward and simple for the majority of customers. However, in some cases manual help might be required to ease this transition.

## 4.6 Removed Items

None.

# 5  Using This Release

This release contains the following:

- Radio Abstraction Interface Layer (RAIL) stack library
- Connect Stack Library
- RAIL and Connect Sample Applications
- RAIL and Connect Components and Application Framework

This SDK depends on the Simplicity Platform. The Simplicity Platform code provides functionality that supports protocol plugins and APIs in the form of drivers and other lower layer features that interact directly with Silicon Labs chips and modules. Simplicity Platform components include EMLIB, EMDRV, RAIL Library, NVM3, and mbedTLS. Simplicity Platform release notes are available through Simplicity Studio's Documentation tab.

For more information about the Flex SDK v3.x see UG103.13: RAIL Fundamentals and UG103.12: Silicon Labs Connect Fundamentals. If you are a first time user, see QSG168: Proprietary Flex SDK v3.x Quick Start Guide.

## 5.1  Installation and Use

The Proprietary Flex SDK is provided as part of the Simplicity SDK, the suite of Silicon Labs SDKs. To quickly get started with the Simplicity SDK, install Simplicity Studio 5, which will set up your development environment and walk you through Simplicity SDK installation. Simplicity Studio 5 includes everything needed for IoT product development with Silicon Labs devices, including a resource and project launcher, software configuration tools, full IDE with GNU toolchain, and analysis tools. Installation instructions are provided in the online Simplicity Studio 5 User's Guide.

Alternatively, Simplicity SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/simplicity_sdk for more information.

Simplicity Studio installs the GSDK by default in:

- (Windows): C:\Users\<NAME>\SimplicityStudio\SDKs\simplicity_sdk
- (MacOS): /Users/<NAME>/SimplicityStudio/SDKs/simplicity_sdk

Documentation specific to the SDK version is installed with the SDK. Additional information can often be found in the knowledge base articles (KBAs). API references and other information about this and earlier releases is available on https://docs.silabs.com/.

## 5.2  Security Information

**Secure Vault Integration**

When deployed to Secure Vault High devices, sensitive keys are protected using the Secure Vault Key Management functionality. The following table shows the protected keys and their storage protection characteristics.

| Wrapped Key | Exportable / Non-Exportable | Notes |
|---|---|---|
| Thread Master Key | Exportable | Must be exportable to form the TLVs |
| PSKc | Exportable | Must be exportable to form the TLVs |
| Key Encryption Key | Exportable | Must be exportable to form the TLVs |
| MLE Key | Non-Exportable | |
| Temporary MLE Key | Non-Exportable | |
| MAC Previous Key | Non-Exportable | |
| MAC Current Key | Non-Exportable | |
| MAC Next Key | Non-Exportable | |

Wrapped keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.
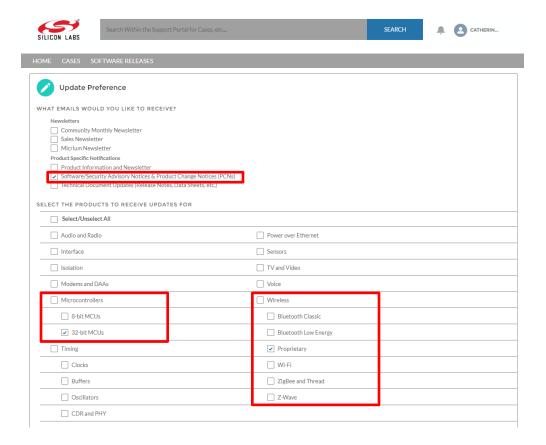
Wrapped keys that are marked as "Exportable" can be used or shared at runtime but remain encrypted while stored in flash.

For more information on Secure Vault Key Management functionality, see AN1271: Secure Key Storage.

**Security Advisories**

To subscribe to Security Advisories, log in to the Silicon Labs customer portal, then select **Account Home**. Click **HOME** to go to the portal home page and then click the **Manage Notifications** tile. Make sure that 'Software/Security Advisory Notices & Product Change Notices (PCNs)' is checked, and that you are subscribed at minimum for your platform and protocol. Click **Save** to save any changes.
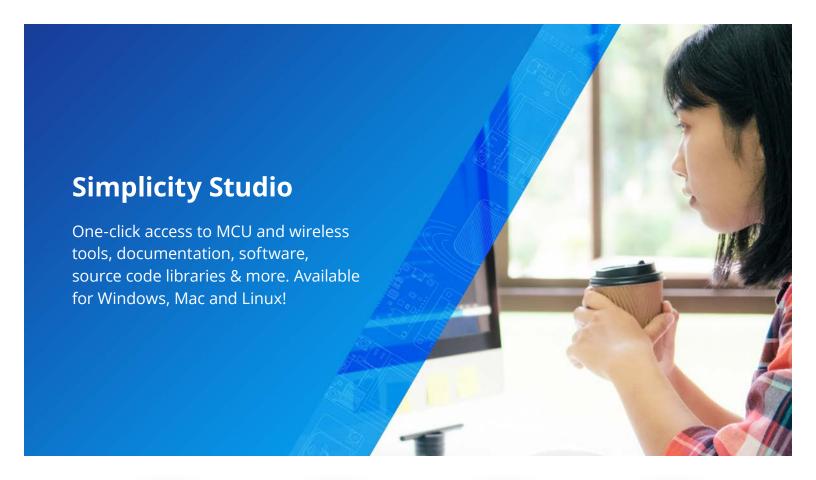
The following figure is an example:



## 5.3    Support

Development Kit customers are eligible for training and technical support. Use the Silicon Labs Flex web page to obtain information about all Silicon Labs Thread products and services, and to sign up for product support.

You can contact Silicon Laboratories support at http://www.silabs.com/support.

# Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

**IoT Portfolio**
www.silabs.com/IoT

**SW/HW**
www.silabs.com/simplicity

**Quality**
www.silabs.com/quality

**Support & Community**
www.silabs.com/community

**Silicon Laboratories Inc.**
**400 West Cesar Chavez**
**Austin, TX 78701**
**USA**

**SILICON LABS**

**www.silabs.com**