Gecko Platform 4.0.1.0 GA
Gecko SDK Suite 4.0
January 26, 2022

The Gecko Platform provides infrastructure support for applications developed with higher-level protocols, and it provides an interface with the underlying hardware. It is composed of the following modules:

CMSIS Device is a vendor-independent hardware abstraction layer for the Cortex®-M processor series.

Peripherals provides a complete peripheral API for all Silicon Labs EFM32, EZR32 and EFR32 MCUs and SoCs.

Drivers is the Gecko Platform driver library for EFM32, EZR32 and EFR32 on-chip peripherals. Drivers are typically DMA-based and use all available low-energy features.

Services includes common services such as NVM3 and Power Manager.

Common components are used throughout the SDKs.

Middleware includes the Capacitive Sensing Firmware Library and the GLIB graphics library, along with Micrium OS stacks like CAN/CANopen, File System, Networking and USB Device and Host.

Security includes mbed TLS and other security services.

Operating System includes Micrium OS Kernel as well as other things related to Operating Systems such as a CMSIS-RTOS2 layer.

The **Gecko Bootloader** is a code library configurable through Simplicity Studio's IDE to generate bootloaders that can be used with a variety of Silicon Labs protocol stacks. The Gecko Bootloader can be used with EFM32 and EFR32 Series 1 and later devices.

Machine Learning includes TensorFlow Lite Micro components, used to run neural network inference, and related preprocessing components.

Examples are example applications illustrating platform functionality.

Boards and External Devices cover supported hardware.

Other Gecko Platform Components regroups changes to documentation, project building and configuration, as well as any other aspects related to Gecko Platform.

RAIL (Radio Abstraction Interface Layer) provides a customizable radio interface layer that supports proprietary or standards-based wireless protocols. RAIL use by application protocols such as Silicon Labs Zigbee or Silicon Labs Connect is managed through the stack library. Direct RAIL use is exposed through the Flex SDK.

These release notes cover SDK version(s):

Gecko Platform 4.0.1.0 released January 26, 2022
Gecko Platform 4.0.0.0 released December 15, 2021

KEY FEATURES

CMSIS Device
- Added support for ZGM230S modules

Drivers
- Added support for RBG LEDs in the LED driver

Services
- Added password and session protection in the CLI

Middleware
- Fixed vulnerabilities in Micrium OS Net
- Deprecated Micrium OS USB Device and USB Host
- Deprecated Gecko USB

Security
- Mbed TLS is updated to version 3.0.0

Operating System
- Changed default configurations of the Micrium OS Kernel to reduce code size

Gecko Bootloader
- Projects now supported in Project Configurator
- Provided as a full-source delivery

Machine Learning
- Updated TensorFlow Lite Micro version and updated quality to production level
- Added accelerated kernels and automatic initialization of TensorFlow Lite Micro

Examples
- Added emode demo
- Added new machine learning examples

Contents

1 CMSIS Device

1.1 New Items

Added in release 4.0.0.0

- Added support for the following new OPNs:
    - ZGM230SA27HGN2
    - ZGM230SB27HGN2
    - ZGM230SA27HNN1

1.2 Improvements

Changed in release 4.0.1.0

- Added module- and part-specific DCDC initialization headers to have more control on DCDC configuration.

1.3 Fixed Issues

Fixed in release 4.0.0.0

ID #	Description
692656	A few fixes have been made in EFM32LG latest revisions: Errata CMU_E113 applies only to rev D Errata EMU_E107 applies to rev A,B,C,D,E,F Errata EMU_E110 applies only to rev E
745419	Fixed RAM SIZE on Series 2 for MSC ECC configuration. The previous implementation was leading to an infinite wait in mscEccReadWriteExistingDma while enabling ECC on devices with less RAM than the maximum possible in the family.

1.4 Known Issues in the Current Release

None

1.5 Deprecated Items

None

1.6 Removed Items

None

2 Peripherals

2.1 New Items

Added in release 4.0.0.0

- The configuration `EM_MSC_RUN_FROM_RAM` is added. It can be used to run flash write functions from RAM.

2.2 Improvements

Changed in release 4.0.0.0

- Default behavior of em_eusart is now to set the first byte force load configuration active when using the EUSART in synchronous master mode. This will improve compatibility between em_usart and em_eusart when sending the first byte in the FIFO after chip select is asserted.
- CS invert feature is added. Set csInv in USART_InitSync_TypeDef to true to make chip select active high.
- Added a new field to the structure LESENSE_ChDesc_TypeDef{}: .offset. This initialization structure is passed to LESENSE_ChannelConfig() to configure a specific LESENS E channel. .offset. This allows the user to specify information whose meaning depends on the interaction between the LESENSE/ACMP or LESENSE/IADC.
- Generate interrupt on both positive and negative edges is added to LESENSE's LESENSE_ChIntMode_TypeDef enum.
- For the PCNT module, only the first twelve PRS channels were supported as inputs. As some devices have more than twelve channels, PCNT_PRSSel_TypeDef has been extended to support all of them.

2.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
760314	Fixed to prevent FNSL reset when the same channels are selected for prs_combine api.
755808	I2C_E303 workaround is now enabled for xG21 and xG22, in addition to the previous change for xG23.

Fixed in release 4.0.0.0

ID #	Description
707597	Fixed a potential issue in some em_gpio functions that were doing non-atomic read-modify-write of some registers.
738965	Fixed conversion warning and undefined behavior in TIMER_MaxCount.
718548	Fixed an issue in VDAC settings [Channel 0 Start Reset Prescaler] and [Sine Wave Reset When inactive] where they were incorrectly configured during initialization.
743956	Set functional default value for ACMP BIASPROG on Series 2 devices. EFR32xG21 cannot use a value less than four (4). EFR32xG23 cannot use a value less than two (2). See relevant data sheets for typical delay values.

2.4 Known Issues in the Current Release

None

2.5 Deprecated Items

None

2.6 Removed Items

None

3 Drivers

3.1 New Items

Added in release 4.0.0.0

- Added support for RBG LEDs in the LED driver.

3.2 Improvements

Changed in release 4.0.0.0

- PWM and LED Instances are now added to the component catalog.
- Added a warning to SPIDRV_DeInit function to prevent calling with uninitialized instance handle.
- New function GPIO_EM4WUExtIntConfig() is added to configure EM4WU pins as level-sensitive interrupt source.
- Added new function GPIOINT_CallbackRegisterExt() to GPIOInterrupt module to allow registering a callback function with context. The function will also return the first available GPIO interrupt number, starting with interrupt number corresponding to the pin number.
- Changed simple_button API to take a pointer to the button handle as argument. This allows simple_button to use the function GPIOINT_CallbackRegisterExt() recently added to GPIOInterrupt.
- Update GPIOINT example in the documentation to remove usage of deprecated API/header.

3.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
765377	Fixed a bug in the RGB and RGBW LED Drivers that, under some conditions, prevented the LED from turning on.
757902	Fixed several issues with Keyscan driver, including missing files error, configuration file structure, pintool error and added requirements on the number of rows/columns inside the Keyscan configuration file.
772793	Fixed a bug preventing NVM3_DEFAULT_CACHE_SIZE to be set to 0.

Fixed in release 4.0.0.0

ID #	Description
725732	Fixed a bug in SPIDRV to remove the requirement on an energy mode when aborting a transfer.

3.4 Known Issues in the Current Release

None

3.5 Deprecated Items

None

3.6 Removed Items

None

4 Services

4.1 New Items

Added in release 4.0.0.0

- Added password protection option to the CLI.
- Added session protection hooks to the CLI.
- Added support for Software Flow control in the IO stream UART interfaces.

4.2 Improvements

Changed in release 4.0.0.0

- Changed the CLI instance's task priority configuration to use the cmsis_rtos priority method.
- It is now possible to change the default unity output stream in IO Stream.
- Added contribution for iostream_retaget_stdio in the component catalog.
- IO Stream's RTT stream is set by default at NO BLOCK TRIM. This allows keeping the RTT module in a release build without issue. Having blocking mode always set causes problems when no host is present to consume the data.
- Improved Sleep Timer example in the documentation.

4.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
765513	Fixed a problem where, under heavy activity in the Sleep Timer, the hardware timer could be misconfigured and timers could stop firing.
737529	Adjusted how the Power Manager provides notifications to reduce a risk of the notifications not being called atomically.
758662	Fixed a compilation error with IO Stream's SWO on IAR.

Fixed in release 4.0.0.0

ID #	Description
688476	Fixed an issue in the Power Manager where restoring from EM3 to EM2 would restore all clock trees instead of just the LF clock tree.
718554	Fixed a potential issue in IO Stream UART by adding an atomic section to protect shared variables related to reception counters.
735675	Fixed a bug in IO Stream UART initialization, where a CMSIS RTOS mutex variable was not initialized correctly.
720001	Fixed a race condition in IO Stream UART that could unbalance the Power Manager requirements leading to the system not being able to go to EM2.
747738	Fixed an issue where, in some situations, functions sl_sleeptimer_ms_to_tick() and sl_sleeptimer_ms32_to_tick() could return 1 tick more than expected.
687640	Fixed an issue in Sleep Timer module where the delta timers list could get desynchronized over time when timers are created/stopped during the Sleep Timer interrupt.

4.4 Known Issues in the Current Release

None

4.5   Deprecated Items

SL_DEVICE_INIT_HFXO_AUTOSTART and SL_DEVICE_INIT_HFXO_AUTOSELECT configurations on series 1 devices are now deprecated. Function CMU_HFXOAutostartEnable() should be used instead. Be extra careful when using HFXO auto-start and auto-select feature on Series 1, since it is not compatible with the Power Manager module.

4.6   Removed Items

None

5   Common

5.1   New Items

- Added sl_endianness.h for helping to detect the endianness of the system.

5.2   Improvements

None

5.3   Fixed Issues

None

5.4   Known Issues in the Current Release

None

5.5   Deprecated Items

None

5.6   Removed Items

None

6 Middleware

6.1 New Items

None

6.2 Improvements

None

6.3 Fixed Issues

Fixed in release 4.0.0.0

ID #	Description
746737	Fixed issue with Micrium OS USB-Device where the BSP for USB was not automatically loaded.
730445	Fixed missing dependency in the metadata for HTTP component and TCPIP core on sleeptimer
734360	Fix issue in Micrium OS Net where the message length was not validated when parsing a DNS response.
734359	Fixed a potential issue by using a random value for the DNS Query ID number in Micrium OS Net.
730679	Fixed an error path in the function FS_FAT_JournalRdWr(). The error condition was not setting the 'p_err' argument properly before returning from the macro RTOS_CRITICAL_FAIL_EXEC().

6.4 Known Issues in the Current Release

None

6.5 Deprecated Items

- Micrium OS USB-Device is deprecated and will be removed in an upcoming release. Please consider migrating to sl_usb.
- Micrium OS USB-Host is deprecated and will be removed in an upcoming release.
- Gecko USB is deprecated and will be removed in an upcoming release. Please consider migrating to sl_usb.

6.6 Removed Items

None

7 Security

7.1 New Items

Added in release 4.0.1.0

- X25519 and Ed25519 algorithms, along with related key management functionality, are now accelerated on Series-2 Secure Vault Mid devices (EFR32xG21A, EFR32xG23A, and so on). Because this functionality depends on the SE firmware being upgraded, software implementations of these algorithms are enabled by default in PSA Crypto and Mbed TLS. For code size/performance optimizations, see the new configuration option 'SL_SE_ASSUME_FW_AT_LEAST_2_1_7'.

Added in release 4.0.0.0

- Upgraded the Mbed TLS instance to be based on version 3.0.0. The upgrade comes with a set of deprecated and changed APIs, as well as security improvements. Importantly, the change also brings added functionality to the Mbed TLS and PSA Crypto APIs -- and paves the way for many future additions.

  For guidance during migration, the official changelog and migration guide can be found at: https://github.com/ARMmbed/mbed-tls/blob/v3.0.0/ChangeLog and https://github.com/ARMmbed/mbedtls/blob/development/docs/3.0-migration-guide.md, respectively.

- SE Manager APIs sl_se_read_cert_size() and sl_se_read_cert() are now available on Secure Vault Mid (previously only Secure Vault High).

7.2 Improvements

Changed in release 4.0.0.0

- Applications making use of the "built-in key" functionality available from the PSA Crypto API now must explicitly include the corresponding component ("Support for using externally provisioned keys through PSA"). By default, this support is not included, to save code size.

- Mbed TLS' strerror functions are now optionally included. If needed, these can be added to the project with the relevant component ("Mbed TLS Support for strerror").

- Due to moving to Mbed TLS 3.0, the configuration option, which was previously used to set the maximum length of both of the SSL input/output buffers (MBEDTLS_SSL_MAX_CONTENT_LEN), has been split into two individual configuration options (SL_MBEDTLS_SSL_IN_CONTENT_LEN and SL_MBEDTLS_SSL_OUT_CONTENT_LEN). Users should update their configuration file accordingly when migrating to this SDK version.

- SE Manager sl_se_derive_key_pbkdf2() now takes a pseudorandom function identifier argument instead of the previous hash function identifier. This enables the use of AES-CMAC-PRF-128 in the PBKDF2 algorithm -- a functionality that has been added to the SE firmware for newer chips. The change has been made in a backward-compatible manner.

7.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
756360	Fixes an issue causing ECDSA over secp224r1 to not work through PSA Crypto on EFR32xG23.
728573	In SE Manager and MbedTLS, fix support for static mutex allocation in order to support disabling dynamic memory allocation in FreeRTOS.

Fixed in release 4.0.0.0

ID #	Description
749633	Fixed GCM support on Series-1 devices, which was failing when the plaintext and/or the additional authentication data was 16K-16 bytes or more.
738688	Since the GSDK v3.1 release, the yield mode in SE Manager was broken when running MicriumOs because MicriumOS switched from using PRIMASK as critical region protection to using BASEPRI. This potentially allowed the SEMBRX_IRQn ISR to break critical regions in the kernel because the interrupt priority was set to 0 by the SE Manager. The fix sets the interrupt priority level of SEMBRX_IRQn to CORE_ATOMIC_BASE_PRIORITY_LEVEL (instead of 0).

7.4 Known Issues in the Current Release

None

7.5 Deprecated Items

- SE Manager API sl_se_gcm_starts() is deprecated in favor of sl_se_gcm_multipart_start()
- SE Manager API sl_se_gcm_update() is deprecated in favor of sl_se_gcm_multipart_update()
- SE Manager sl_se_gcm_finish() is deprecated in favor of sl_se_gcm_multipart_finish()

7.6 Removed Items

None

8 Operating System

8.1 New Items

Added in release 4.0.1.0

- Added a new errno module that provides its own errno.h header file and errno definition. Thread-safe when used with FreeRTOS and Micrium OS. Can also be used in interrupt context.

8.2 Improvements

Changed in release 4.0.0.0

- Added support for osMutexNew NULL attribute in Micrium OS' CMSIS-RTOS2 abstraction layer.
- Updated documentation about Micrium OS Kernel's idle hook.
- Modified Micrium OS Kernel's default configurations to reduce the code size. Also disabled DWT by default to reduce the likeliness of it being available as an attack vector.

8.3 Fixed Issues

Fixed in release 4.0.0.0

ID #	Description
748434	Fixed an issue in osMessageQueueGet() function of Micrium OS' CMSIS-RTOS2 port. Argument msg_prio was not checked for NULL value before being used.
748577	Fixed an issue in Micrium OS' CMSIS-RTOS2 port where function osEventFlagsSet() was prematurely setting the global group flags, leading to a potential race condition.
730413	Fixed a wrap-around issue in Micrium OS Kernel's OSTimeGet().
738384	Fixed a bug where, when doing floating point operation from the idle context, an ISR stack corruption rarely could occur at a later time when entering an exception/interrupt.

8.4 Known Issues in the Current Release

None

8.5 Deprecated Items

None

8.6 Removed Items

None

9 Gecko Bootloader

9.1 New Items

Added in release 4.0.0.0

In GSDK 3.0 Silicon Labs introduced a complete update to its Simplicity Studio tool suite, as well as a new, component-based Gecko Platform architecture. In this release the Gecko Bootloader has been updated to take advantage of this architecture. With Simplicity Studio 5.3 and GSDK v4.0, Gecko Bootloader developers will benefit from the following component-based project configuration features:

- Search and filter to find and discover software components that work with the target device
- Automatically pull in all component dependencies and initialization code
- Configurable software components
- All configuration settings in C header files for usage outside of Simplicity Studio
- Configuration validation to alert developers to errors or issues
- Easily manage all project source via git or other SCM tools
- Managed migration to future component and SDK versions
- Simplified transitions from Silicon Labs development kits to custom hardware

Instead of configuring project functionality through Application Builder (AppBuilder) plugins, equivalent functionality is now available through Gecko Bootloader and Platform components and configuration tools such as Project Configurator.

- Gecko Bootloader is now provided as a full-source delivery, as opposed to previously being a combination of compiled libraries and source code.
- Added **bootloader-storage-spiflash-single-1024k** sample apps for storing single image for slot size of 1024k.

9.2 Improvements

None

9.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
764405	Fixes an issue related to EFR32*BG*21 device not showing bootloader example on SDK 4.0.0.
764854	Bootloader sample apps for Series 0 devices are showing up in SDK 4.0.0.
774286	Bootloader storage slot overlap error in validation script in SDK 4.0.0.

Fixed in release 4.0.0.0

ID #	Description
685698	Fixed an issue where DFU commands did not return correct sl_status_t error codes.
690465	Added a new bootloader component bootloader_app_properties to add an instance of the AppProperties_t struct to the application. This component is automatically installed on installing the bootloader_interface component in the project.
739464	Fixed Bootloader #define macro issue with IAR compiler.
744119	Added documentation for delay_microsecond() in bootloader delay-driver for delays longer than expected.

9.4 Known Issues in the Current Release

None

9.5   Deprecated Items

Deprecated in release 4.0.0.0

Gecko USB is deprecated.

9.6   Removed Items

Removed in release 4.0.0.0

- Removed app builder flow in Gecko bootloader.
- Support for the legacy EBLv2 format has been removed.

10 Machine Learning

10.1 New Items

Added in release 4.0.0.0

- Added accelerated kernels for the following operations in TensorFlow Lite Micro: ADD, FULLY_CONNECTED, AVERAGE_POOL_2D, MAX_POOL_2D, CONV_2D, DEPTHWISE_CONV_2D, TRANSPOSE_CONV_2D
- Added component for initializing TensorFlow Lite Micro including autogenerated opsResolver based on layers used in flatbuffer.
- Added generation of C header file defines from parameters embedded in flatbuffers (.tflite files) for TensorFlow Lite Micro.
- Added Audio Feature Generator component for audio-based machine learning applications.

10.2 Improvements

Changed in release 4.0.0.0

- Updated version of TensorFlow Lite Micro to commit #3e190e5389be49c94475e509452bdae245bd4fa6 of github.com/tensorflow/tflite-micro and updated quality of TensorFlow Lite Micro components to production.

10.3 Fixed Issues

None

10.4 Known Issues in the Current Release

None

10.5 Deprecated Items

None

10.6 Removed Items

None

11 Examples

11.1 New Items

Added in release 4.0.0.0

- Added sample applications to demonstrate USB capabilities (composite device and others)

- Added an example for emode that demonstrates the power consumption using Energy Profiler for different combinations of emodes, clocks running with/without DCDC, operation performed. This example used EMU for entering EM1-4 modes and CMU for handling clocks. This example is to be used as a demo only to understand the power consumption for various configurations and **should not** be used as a starting point for implementation.

- Added sample application "SE Manager Attestation" that demonstrates how to generate, parse and print attestation tokens using the SE Manager API.

- Added the following Machine Learning sample applications: TensorFlow Model Profiler, TensorFlow Lite Micro Magic Wand, Voice Control Light.

11.2 Improvements

None

11.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
758312	Increased heap size for USB HID sample application.

11.4 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 https://www.si-labs.com/products/software.

ID #	Description	Workaround
664803	Se_manager and psa_crypto sample apps do not work correctly in Simplicity Studio 5's launch console.	In the launch console, change the line terminator selection to None.

11.5 Deprecated Items

None

11.6 Removed Items

None

12 Boards and External Devices

12.1 New Items

Added in release 4.0.0.0

- Added support for new Wireless Pro Kit Mainboard (BRD4002A).
- .Added support for the following new OPNs:
  - BRD4205B
  - ZWAVE-PK800A

12.2 Improvements

Changed in release 4.0.1.0

- Created pressure driver that abstracts bmp280 and bmp3xx.

Changed in release 4.0.0.0

- Updated si70xx_driver, si72xx_driver, si1133_driver and si7210_driver to require an instance of i2cspm so users don't have to manually add an i2cspm instance to their project.
- VCOM uses now EUSART0 and memlcd uses EUSART1. This resolves conflicts when both vcom and memlcd are running at the same time.
- Added Si446x driver to interface with Si4468.

12.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
759792	Fixed VCOM configurations for BRD4191A.

Fixed in release 4.0.0.0

ID #	Description
714788	Fixed bug to prevent an error when entering into EM2 and using the LCD by adding an LCD refresh call before writing to LCD. Refresh LCD will re-initialize SPI communication.

12.4 Known Issues in the Current Release

None

12.5 Deprecated Items

None

12.6 Removed Items

None

13 Other Gecko Platform Software Components

13.1 New Items

Added in release 4.0.0.0

- Updated Segger SystemView's version and added FreeRTOS support.
- Added API for Sync Flash Write.

13.2 Improvements

- IAR flash loader packaged in DFP packs now supports different peripheral locations (secure or non-secure), enabling first stage of TrustZone development.
- IAR and Segger flash loaders now support TrustZone-enabled devices and can write in the flash even if moved to its non-secure address.
- Updated common io I2C functions to allow a transaction with a timeout of 0. A transaction with timeout value of 0 is replaced by a default value used in common io.

13.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
756383	Fixed a circular dependency in Simplicity Studio's metadata that prevented uninstalling the FreeRTOS component.

Fixed in release 4.0.0.0

ID #	Description
739975	Fixed an issue in the Amazon FreeRTOS Libraries to make iot_timer_get_value() callable from an ISR.

13.4 Known Issues in the Current Release

None

13.5 Deprecated Items

None

13.6 Removed Items

None

14 RAIL Library

14.1 New Items

Added in release 4.0.0.0

- Added support for the ZGM230SA27HGN, ZGM230SA27HNN, and ZGM230SB27HGN modules.
- Added RAIL_GetTxPacketsRemaining() API for use when handling one of the RAIL_EVENTS_TX_COMPLETION to get a sense of how many transmits remain in a RAIL_SetNextTxRepeat() sequence.
- Added PA curves for HP, MP, LP and LLP modes on all EFR32xG23 radio boards.
- Added RAIL_PA_BAND_COUNT to count RAIL_PaBand_t.
- Added a new RAIL_RxDataSource_t to capture direct mode data on supported devices.
- Added a new RAIL_IEEE802154_Config2p4GHzRadioCustom1 API to configure an alternate IEEE 802.15.4 PHY with slightly different performance characteristics for the EFR32xG12 and EFR32xG13 parts. Use this API if instructed by Silicon Labs for your use case.

14.2 Improvements

Changed in release 4.0.1.0

- Restricted the SL_RAIL_UTIL_PA_RAMP_TIME_US to 10 µs on some EFR32 modules to match the certification conditions.
- Updated the Z-Wave PHYs for the EFR32xG23 to prevent a sensitivity degradation on the R2 (9.6 kbps) PHY.

Changed in release 4.0.0.0

- Added IEEE802.15.4 Coexistence and FEM PHYs to EFR32xG12 and EFR32xG13 based modules.
- Updated IEEE802.15.4 FEM PHYs on EFR32xG12 and EFR32xG13 based modules for improved performance.
- Updated all header files to have extern "C" when being built with C++ for compatibility.
- Made the RAIL_EnableRxDutyCycle() API safe to call in a multiprotocol application.
- To save both flash and RAM, moved information formerly contained in RAIL_Config_t::protocol, RAIL_Config_t::scheduler, and RAIL_Config_t::buffer internal to RAIL and sized appropriately for single vs. multiprotocol. RAIL multiprotocol now provides two internal state buffers for two protocols by default. An application that needs more must now call RAIL_AddStateBuffer3() or RAIL_AddStateBuffer4() to add a 3rd and 4th buffer, respectively. Otherwise RAIL_Init() will fail when trying to initialize a 3rd or 4th protocol.
- "RAIL Utility, Coexistence" component GPIO interrupt numbers are now chosen at runtime to avoid conflicts.
- A new RAIL API RAIL_GetSchedulerStatusAlt will now return more descriptive radio scheduler events as well as the RAIL_Status_t of the RAIL API invoked by the radio scheduler. As a part of the new API, new RAIL_SchedulerStatus_t events have been added while retaining the previous ones for backwards compatibility. Note that the underlying values of the existing RAIL_SchedulerStatus_t events may have changed.
- Updated the RAIL_IEEE802154_Config2p4GHzRadio*Fem PHYs on the EFR32xG12 and EFR32xG13 devices to improve performance.
- The example CSV files referred to in *AN1127: Power Amplifier Power Conversion Functions in RAIL 2.x* are updated with realistic values.

14.3 Fixed Issues

Fixed in release 4.0.1.0

ID #	Description
764234	Changed Quuppa channel 2 frequency from 2480 MHz to 2403 MHz.
773178	Fixed a compiler warning in "RAIL Utility, Callbacks" component when app_assert component ignores asserts.

Fixed in release 4.0.0.0

ID #	Description
646980	An attempt to use an unsupported built-in radio channel configuration, e.g., on a module that does not support that protocol or configuration, will now trip RAIL_ASSERT_FAILED_INVALID_CHANNEL_CONFIG rather than returning success and ignoring the configuration.
671651	Fixed timing problems with certain State_Transitions or Rx_Channel_Hopping delay values on the EFR32xG22 and newer parts.
682739	Fixed an issue with the BLE coded PHY's modulation index on the EFR32xG21 parts that could cause deviation measurements to fail.
714271	Fixed an issue where RAIL_IEEE802154_Config2p4GHzRadio*() and RAIL_IEEE802154_ConfigGB*Radio() functions were improperly clearing or setting certain RAIL_IEEE802154_EOptions_t. Also documented that these functions still implicitly clear or set certain RAIL_IEEE802154_GOptions_t suitable for that configuration.
716369	Fixed an issue where incorrect radio transition times were being applied at higher temperatures when using the high power PA on EFR32xG22 parts.
723098	Fixed RAIL_SetFixedLength(handle, RAIL_SETFIXEDLENGTH_INVALID) to restore dynamic frame length operation if the current PHY was originally configured for that.
738931	Fixed an issue with the BLE Coded PHYs on the EFR32xG22 device that could cause some packets to be improperly sent and not trigger a RAIL_EVENT_TX_PACKET_SENT event.
739594	Fix an issue with the RX_IQDATA_FILTLSB RAIL_RxDataSource_t on EFR32xG23 parts where the data did not properly saturate and was instead just the lower 16 bits of IQ sample data.
744323	Fixed an issue when using BLE AoX where non-AoX packets were transmitted on an undefined antenna. They will now always use the first antenna in the configured RAIL_BLE_AoxConfig_t::antArrayAddr pattern.
745528	Fixed some incorrect RAIL_RxPacketInfo_t::filterMask values for 802.15.4 ACKs when promiscuous, or when the PanId coordinator received a packet with only source PanId and no destination address.
753860	Fixed an issue when running IR Calibration on the EFR32xG23 (RAIL_CalibrateIrAlt) where we could compute a completely invalid IRCAL value for certain PHYs and chips.
754219	Increase maximum BLE coex request window setting, SL_RAIL_UTIL_COEX_REQ_WINDOW, in "RAIL Utility, Coexistence" component from 255 to 5000.

14.4 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 https://www.si-labs.com/products/software.

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: - Wi-SUN FSK mode 1a exhibits a PER floor with frequency offsets around ± 8 to 10 KHz - Wi-SUN FSK mode 1b exhibits a PER floor with frequency offsets around ± 18 to 20 KHz	

14.5 Deprecated Items

None

14.6 Removed Items

None

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