Gecko Platform 4.1.4.0 GA

Gecko SDK Suite 4.1

January 18, 2023

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.

CPC (Co-Processor Communication) provides a library to communicate between two processors using a serial link. CPC is used by the ACP & RCP solutions

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):

KEY FEATURES

- Initial release of CPC
- Added support for EFR32xG24; MGM240, EFR32MR21 and FGM230
- Several code size improvements related to Power Manager, HFXO Manager and em_crypto

Tools and Dependencies

- Updated compiler support to GCC 10.3-2021.10 and IAR 9.20.4
- Updated CMSIS to version 5.8.0

Drivers

- Added a new component to synchronize UART/PTI settings between WSTK mainboard and the radio board
- Added support for Analog Joystick driver to use the joystick functionality on mainboard v2

Security

- Mbed TLS is updated to version 3.1.0
- Added software support for TrustZone, BETA quality

Bootloader

- Jedec driver support for external SPI flash
- A new sample app for devices with external SPI flash

Gecko Platform 4.1.4.0 released January 18, 2023
Gecko Platform 4.1.3.0 released October 19, 2022 (early access part support, one fixed RAIL issue)
Gecko Platform 4.1.2.0 released September 28, 2022
Gecko Platform 4.1.1.0 released August 17, 2022
Gecko Platform 4.1.0.0 released June 8, 2022

Contents

1 CMSIS Device

1.1 New Items

Added in release 4.1.1.0

- Added support for EFR32FG23B021F512IM48 and EFR32FG23B021F512IM40

Added in release 4.1.0.0

- Added support for EFR32xG24 devices.
- Added support for EFR32MR21 devices.
- Added support for the following new OPNs:
  - MGM240L022VNF
  - MGM240L022RNF
  - FGM230SA27HGN
  - FGM230SB27HGN

1.2 Improvements

Changed in release 4.1.1.0

- Fix the MISRA violations:
  - em_bus.h: MISRAC2012-Rule-14.4_c
  - em_cmu.h: MISRAC2012-Rule-8.2_a, MISRAC2012-Rule-14.4_d, MISRAC2012-Rule-15.6_e
  - efr32xg2x_prs_signals.h: MISRAC2012-Dir-4.10
  - efr32xg2x_dma_descriptor.h: MISRAC2012-Dir-4.10
  - efr32xg2x_ldmaxbar_defines.h: MISRAC2012-Dir-4.10

Changed in release 4.1.0.0

- Upgraded from CMSIS 5.3.0 to 5.8.0. This upgrade includes:
  - Startup files have now become more generic (compiler-independent startup files)
  - DSP and CMSIS are separated as part of this update
  - CMSIS core include path has changed from platform/CMSIS/include to platform/CMSIS/Core/Include
  - DSP changes are now present under platform/CMSIS/DSP/Include and platform/CMSIS/DSP/include/dsp
  - DSP libs are now present under Lib/GCC
  - Stackseal support is now available in both startup and linker files

1.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
729400	Updated Flash wait states for most Series 2 devices, when the frequency is above 75 MHz.
759169	Removed some radio-related content from the CMSIS headers.

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

None

2.2 Improvements

Changed in release 4.1.1.0

- Clarified usage of the parameter 'presc' for CMU_PrescToLog2() and CMU_ClockPrescSet() to avoid confusion when the users call these functions.

Changed in release 4.1.0.0

- Renamed files em_assert.h and em_common.h to sl_assert.h and sl_common.h and moved them from platform/emlib/ to platform/common/.
- The functions for accessing crypto peripheral's data registers are no longer inlined. This reduces code size for Series 1 devices.

2.3 Fixed Issues

Fixed in release 4.1.1.0

ID #	Description
1022236	The function EMU_DCDCPowerOff was accidentally removed from the Series1 API. It has now been reinstated.
853352	Fixed GCC compilation errors in some EMLIB drivers when the GCC options -wundef and -werror are activated.

Fixed in release 4.1.0.0

ID #	Description
832829	The functions EMU_EnterEM2() and EMU_EnterEM3() were missing a call to the WFI (Wait For Interrupt) instruction following a certain code path. The application on some devices would not enter low energy modes in that case. The missing call to WFI has been added accordingly.
759090	Added EM1RUN and CLRSRC and others missing register bit field to watchdog structure.
756691	Added missing support for RT clocks in em_cmu.
764646	Clarified documentation related to iadcClkSuspend0/1.
813005	Fixed a mistake in emlib's documentation about IADC external reference voltage range.

2.4 Known Issues in the Current Release

None

2.5 Deprecated Items

Deprecated in release 4.1.0.0
Some functions are deprecated in GSDK 4.1. Users should avoid using deprecated APIs. Using a deprecated API will result in a warning. To silence the warning, the user should define SL_SUPPRESS_DEPRECATION_WARNINGS_SDK_4_1 in the project.
The following have been deprecated in platform/emlib:

- bool autoStartEm01;
- bool autoSelEm01;
- bool autoStartSelOnRacWakeup;
- __STATIC_INLINE uint32_t CMU_Log2ToDiv();
- __STATIC_INLINE CMU_HFRCOFreq_TypeDef CMU_HFRCOFreqGet(void);
- __STATIC_INLINE void CMU_HFRCOFreqSet(CMU_HFRCOFreq_TypeDef setFreq);
- __STATIC_INLINE CMU_AUXHFRCOFreq_TypeDef CMU_AUXHFRCOFreqGet(void);

- __STATIC_INLINE void CMU_AUXHFRCOFreqSet(CMU_AUXHFRCOFreq_TypeDef setFreq);
- void EMU_MemPwrDown(uint32_t blocks);
- void EMU_UpdateOscConfig(void);
- __STATIC_INLINE void AES_CBC128();
- __STATIC_INLINE void AES_CBC256();
- __STATIC_INLINE void AES_CFB128();
- __STATIC_INLINE void AES_CFB256();
- __STATIC_INLINE void AES_CTR128();
- __STATIC_INLINE void AES_CTR256();
- __STATIC_INLINE void AES_CTRUpdate32Bit(uint8_t * ctr);
- __STATIC_INLINE void AES_DecryptKey128(uint8_t * out, const uint8_t * in);
- __STATIC_INLINE void AES_DecryptKey256(uint8_t * out, const uint8_t * in);
- __STATIC_INLINE void AES_ECB128();
- __STATIC_INLINE void AES_ECB256();
- __STATIC_INLINE void AES_OFB128();
- __STATIC_INLINE void AES_OFB256();
- __STATIC_INLINE void GPIO_IntConfig();
- __STATIC_INLINE void WDOG_Enable(bool enable);
- __STATIC_INLINE void WDOG_Feed(void);
- __STATIC_INLINE void WDOG_Init(const WDOG_Init_TypeDef *init);
- __STATIC_INLINE void WDOG_Lock(void);
- __STATIC_INLINE bool WDOG_IsEnabled(void);
- __STATIC_INLINE bool WDOG_IsLocked(void);
- #define _EMLIB_VERSION 6.1.1
- #define _EMLIB_VERSION_MAJOR 6
- #define _EMLIB_VERSION_MINOR 1
- #define _EMLIB_VERSION_PATCH 1

2.6 Removed Items

None

3 Drivers

3.1 New Items

Added in release 4.1.0.0

- Added support for Analog Joystick driver to use the joystick functionality on mainboard v2.
- Added a new component **Configuration_over_swo** to synchronize UART/PTI settings between WSTK mainboard and the radio board.

3.2 Improvements

Changed in release 4.1.0.0

- Creation of an override script to generate config for RGB and RGBW drivers.
- UARTDRV handle structure now has the same size regardless of compilation configuration.
- SPIDRV handle structure now has the same size regardless of compilation configuration.

3.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
841470	Fixed SPIDRV EUSART initialization and operation that could cause bit-shifted read data.
764090	Fixed UARTDRV to reduce power consumption when going to EM2.
724551	Fixed bug that caused interrupt to trigger at incorrect temperatures when using tempdrv with Series2 devices.
674105	Fixed configurator editor issue with UART_DRV LEUART instance.

3.4 Known Issues in the Current Release

None

3.5 Deprecated Items

Deprecated in release 4.1.0.0

Some functions are deprecated in GSDK 4.1. Users should avoid using deprecated APIs. Using a deprecated API will result in a warning. To silence the warning, the user should define SL_SUPPRESS_DEPRECATION_WARNINGS_SDK_4_1 in the project.

- The following have been deprecated in platform/emdrv:
  - void SLEEP_Init(SLEEP_CbFuncPtr_t pSleepCb, SLEEP_CbFuncPtr_t pWakeUpCb);
  - typedef UARTDRV_InitUart_t UARTDRV_Init_t;
  - Ecode_t UARTDRV_Init(UARTDRV_Handle_t handle, UARTDRV_InitUart_t *initData);
  - #define EMDRV_UARTDRV_FLOW_CONTROL_ENABLE EMDRV_UARTDRV_HW_FLOW_CONTROL_ENABLE
- The following have been deprecated in platform/driver:
  - #define __SIMD32_TYPE int32_t __packed

3.6 Removed Items

None

4 Services

4.1 New Items

None

4.2 Improvements

Changed in release 4.1.1.0

- Improved C++ Compatibility in a few C headers.
- Modified SL_RBIT16() function to only accept 16 bits argument.

Changed in release 4.1.0.0

- Added a compile-time configuration in the power manager limit support to EM1, to reduce code size usage.
- HFXO Manager - SL_HFXO_MANAGER_SLEEPY_CRYSTAL_SUPPORT configuration is now disabled (0) by default. This reduces code size.

4.3 Fixed Issues

Fixed in release 4.1.1.0

ID #	Description
825836	Fixed XOFF characters being sent by IO Stream when the UART is configured in hardware flow control mode.
844862	Fixed hard fault when closing multiple instances of NVM3 on Series 2 devices.
845653	Fixed heap-based buffer overflow vulnerability in HTTP Server.
851120	Fixed macro usage when initializing IO Stream for EUSART
845550	Added a configuration option to IO Stream-SWO to print byte per byte in order to be compatible with Commander SWO Parser. The option can be disabled with third-party tools.

Fixed in release 4.1.0.0

ID #	Description
840847	Fixed sl_sleeptimer_set_datetime to configure the timezone.
760113	Fixed possible issue where the hardware timer compare value could, in extreme situations, not be updated on time when the requested value is very close to the current timer count.
758662	Fixed compiler error with iostream_swo on IAR.
816022	Fixed CLI simple password issue over Telnet.

4.4 Known Issues in the Current Release

None

4.5 Deprecated Items

None

4.6 Removed Items

None

5 CPC

5.1 New Items

Added in release 4.1.2.0

- Added support for musl libc
- Added option to configure the timeouts from the CPC Python wrapper.

Added in release 4.1.1.0

- Firmware version can be optionally specified on the secondary.
- Added option to retrieve the firmware version from cpcd.
- Added option to start cpcd conditional to a firmware version.
- Added option to allow firmware update process to be conditional to a firmware version.

Added in release 4.1.0.0

- First GA-quality release of Co-Processor Communication (CPC) module, which allows communication between a primary and a secondary via either UART or SPI through a robust protocol and optionally encrypted endpoint(s).

5.2 Improvements (since beta)

Changed in release 4.1.2.0

- Improved UART driver without hardware flow control to support higher baud rate.
- Print the full running configuration instead of only the configuration file.

Changed in release 4.1.1.0

- Added support for libgpiod.
- Added support for MbedTLS 3.1.0 and kept backwards compatibility with MbedTLS 2.7.0

Changed in release 4.1.0.0

- CPC can now encrypt traffic over the serial link. See https://stash.silabs.com/projects/EMBSW/repos/platform_service/browse for documentation. Security is enabled by default and requires binding the devices as the first step.
- Added User endpoint API for custom communication stream between a host and a target. A sample app is also available.
- Added SPI and UART drivers for EUSART peripheral.
- CPC secondary automatically configures the VCOM setting on the WPK board.
- CPC Daemon supports more bootloader modes to be more versatile.
- CPC can validate the VCOM speed on WSTK/WPK board prior to start.
- CPC can be started in a special mode to validate if the UART pins are correctly connected.
- Added protocol version validation to ensure the Daemon and the Host are compatible.
- When UART communication is selected, CPC ensures the UART Secondary's configuration matches the Host.
- Updated CPC Host configuration to use YAML format.
- Updated CPC Daemon error reporting and tracing.
- Improved CPC Daemon UART driver stability.
- Added CPC Lib version to ensure compatibility between the application and CPC Daemon.
- Added CPC Lib Python C wrapper.
- Provided IOStream bridge to access Target's terminal over CPC using a Telnet.

5.3 Fixed Issues

Fixed in release 4.1.2.0

ID #	Description
1014860	Fixed compilation error with pre-generated CPC SPI EUSART configurations.
1017514	Reduced ASYNC_LOGGER_PAGE_COUNT to use less than the Operating system's default value of 64k.

Fixed in release 4.1.1.0

ID #	Description
850891	Fixed a race condition during the endpoint open sequence where a reply can be sent to the wrong control socket.
849197	Fixed libcpc concurrency issue around opening/closing and fetching the state of the endpoint.
1022583	Fixed cpcd to exit when security session fail to initialize.
845426	Fixed daemon crash when secondary resumes after launch.
1021440	Fixed race condition where the system endpoint would try to unregister a non-registered timer.
1021172	Fixed race condition that could cause on_write_completed callback to be called twice during high interrupt latency situations.
1017162	Fixed possible socket leak between Daemon and Lib
1017166	Fixed an issue where an empty key file would be generated on a failed ECDH exchange.

Fixed in release 4.1.0.0

ID #	Description
N/A	Fixed memory leaks in CPC Secondary core and Drivers.
N/A	Improved CPC Secondary UART drivers with small packets.
N/A	Fixed issue with connection state machine, related to the closing mechanism.
N/A	Fixed issue with the bootloader support.

5.4 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.silabs.com/products/software.

ID #	Description	Workaround
N/A	CPC secondary UART driver with receive buffer larger than 2048 bytes without Hardware Flow control is unstable and it has been disabled.	Large buffer can be used if the UART Flow control is enabled or if using SPI interfaces.

5.5 Deprecated Items

None

5.6 Removed Items

None

6 Common

6.1 New Items

None

6.2 Improvements

Changed in release 4.1.0.0

- Renamed files em_assert.h and em_common.h to sl_assert.h and sl_common.h and moved them from platform/emlib/ to platform/common/.

6.3 Fixed Issues

None

6.4 Known Issues in the Current Release

None

6.5 Deprecated Items

None

6.6 Removed Items

Changed in release 4.1.2.0

- Removed internal files.

7 Middleware

7.1 New Items

None

7.2 Improvements

None

7.3 Fixed Issues

Fixed in release 4.1.1.0

ID #	Description
845653	Fixed heap-based buffer overflow vulnerability in HTTP Server.

Fixed in release 4.1.0.0

ID #	Description
818958	Fix #define name added to rtos_description.h by MicriumOS Net SNTP component.
739495	Fixed compiler error with Micrium OS I/O Silicon Labs Serial Driver with series2 devices

7.4 Known Issues in the Current Release

None

7.5 Deprecated Items

None

7.6 Removed Items

None

8 Security

8.1 New Items

Added in release 4.1.0.0

- Updated base version of Mbed TLS to v3.1.0. See the changelog at https://github.com/ARMmbed/mbedtls/blob/development/ChangeLog
- Added software support for TrustZone on devices based on Cortex-M33. The current version is BETA quality only and may change in incompatible ways in next release. The Application Note *AN1374: Series 2 TrustZone* can be delivered outside the GSDK to users who are evaluating TrustZone support.

8.2 Improvements

Changed in release 4.1.0.0

- PSA Crypto is now automatically initialized when the component is included in a project. The exception is if RAIL is relied on for entropy, because PSA Crypto cannot rely on RAIL being initialized at the time of the service init.
- X25519 and Ed25519 algorithms, along with related key management functionality, are now accelerated on Series-2 Secure Vault Mid devices (EFR32xG21A, EFR32xG23A, etc.). 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 new configuration option 'SL_SE_ASSUME_FW_AT_LEAST_2_1_7'.
- Added support for AES-CCM* without tag (i.e. IEEE 802.15.4 CCM-star in unauthenticated mode) through the PSA Crypto API.

8.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
756360	Fix an issue causing ECDSA over secp224r1 to not work through PSA Crypto on EFR32xG23.
728573	In SE Manager and Mbed TLS, fix support for static mutex allocation in order to support disabling dynamic memory allocation in FreeRTOS.
824453	The TRNG initialization routine for the VSE devices has been updated to read the conditioning key and re-initializes the TRNG if the conditioning key only contains zeros.
826942	Fix an issue where the PSA ITS driver falsely reported PSA_ERROR_INSUFFICIENT_STORAGE error.
830186	Fix the loop counters in sli_crypto_trng_soft_reset().
844823	Fix issue with fully overlapping buffers for multipart GCM for EFR32xG21 devices.

8.4 Known Issues in the Current Release

None

8.5 Deprecated Items

Deprecated in release 4.1.0.0

- Silicon Labs' PSA Crypto drivers for accelerated CBC-MAC operations are now compiled out by default, and have been scheduled for removal in an upcoming release. Until the time of removal, the driver code can be re-enabled by including the preprocessor define 'PSA_WANT_ALG_CBC_MAC' to your project. However, this should be done with caution, as this is a legacy algorithm with security requirements that cannot be enforced by the PSA Crypto API.

8.6 Removed Items

None

9 Operating System

9.1 New Items

None

9.2 Improvements

None

9.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
833210	Fixed race condition with CMSIS-RTOS2 Micrium OS Flag implementation.

9.4 Known Issues in the Current Release

None

9.5 Deprecated Items

None

9.6 Removed Items

None

10 Gecko Bootloader

10.1 New Items

Added in release 4.1.0.0

- Jedec driver support in Bootloader for external SPI flash.
- Added a new bootloader sample application bootloader-storage-spiflash-sfdp-single for devices having external SPI flash.

10.2 Improvements

None

10.3 Fixed Issues

Fixed in release 4.1.2.0

ID #	Description
00289192	Added fix for enabling GPIO clock before accessing GPIOs for Series -2 devices for EZSP SPI Bootloader.
	Added Fix for Simplicity Studio to display bootloader examples for BGM210
	Added fix for bootloader_writeStorage() api for non aligned memory access.

Fixed in release 4.1.1.0

ID #	Description
00287517	Added fix for removing the condition to check if input data length is a multiple of 4 in storage_writeRaw().
	Added Fix for Updating BOOTLOADER_STORAGE_IMPL_INFO_VERSION value to 0x210
00286863	Added fix for XMODEM bootloader update fails when SE secure boot is enabled (GSDK 4.0.2 and 4.1)
00287517	Added fix for Slot start address and slot size to display as hexadecimal values in Simplicity Studio.

Fixed in release 4.1.0.0

ID #	Description
00279315	Added SPI flash bootloader (single image) and SPI flash bootloader (multiple image) bootloader sample apps for part EFM32GG12B430F512GM64 in Simplicity Studio.
	Updated component description for App Properties component in Simplicity Studio.
00281938	Added changes in Simplicity Studio to configure the SPI speed in the SPI Bootloader storage sample app.
00282827	Added changes to install the Bootloader UART Driver component for a Simplicity Studio Bootloader UART project targeted for a custom board.
00282296	Added fix to remove disabling the GPIO clock from SPI_Deinit() routine.
00281391	Added changes in Simplicity Studio to create bootloader-spi-flash-storage-* projects for BRD4165.
00280030	Updated the postbuild step in *UG489: Gecko Bootloader User Guide.*

10.4 Known Issues in the Current Release

None

10.5 Deprecated Items

None

10.6 Removed Items

Removed in release 4.1.1.0

- Gecko USB is removed.
- USB host loader and USB device loader application removed.

11 Machine Learning

11.1 New Items

None

11.2 Improvements

Changed in release 4.1.0.0

ID #	Description
	Upgraded the TensorFlow Lite Micro version in the GSDK from git hash 3e190e5389be49c94475e509452bdae245bd4fa6 to 10b1223343303b45e6f62a433e25efb5f87958d7
	Removed TensorFlow Lite Micro's experimental "Microfrontend" component from the GSDK. Users are advised to use the "Audio Feature Generation", which contains the same functionality

11.3 Fixed Issues

None

11.4 Known Issues in the Current Release

None

11.5 Deprecated Items

None

11.6 Removed Items

None

12 Examples

12.1 New Items

Added in release 4.1.0.0

- Added sample apps to demonstrate CPC secondary application.
- Added sample app to demonstrate the functionality of the Analog Joystick driver.

12.2 Improvements

None

12.3 Fixed Issues

Fixed in release 4.1.1.0

ID #	Description
00287447	Added fix for listing MCU examples for the EFR32BG21 part number in Simplicity Studio.

Fixed in release 4.1.0.0

ID #	Description
833041	NVM3 sample application has been modified to set the CLI input buffer larger than the NVM3 max object size.

12.4 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.silabs.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.

12.5 Deprecated Items

None

12.6 Removed Items

None

13 Boards and External Devices

13.1 New Items

Added in release 4.1.1.0

Added support for the following new OPNs:

- BRD4166C
- BRD2703A

Added in release 4.1.0.0

Added support for the following new OPNs:

- BRD2204C
- BRD4330A
- BRD4331A
- BRD4319A
- BRD2603A
- BRD4328A

13.2 Improvements

None

13.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
824504	Fixed a metadata issue related to BRD4171A and 4176A, where it was wrongfully reported they supported I2CSMP, which is not the case.

13.4 Known Issues in the Current Release

None

13.5 Deprecated Items

Deprecated in release 4.1.0.0

Some functions are deprecated in GSDK 4.1. Users should avoid using deprecated APIs. Using a deprecated API will result in a warning. To silence the warning, the user should define SL_SUPPRESS_DEPRECATION_WARNINGS_SDK_4_1 in the project.

- The following have been deprecated in hardware/kit:
  - void BSP_TraceSwoSetup(void);
  - #define setupSWOForPrint RETARGET_SwoInit
- The following have been deprecated in hardware/driver:
  - sl_status_t sl_efp_set_voa_em01_peak_current();
  - sl_status_t sl_efp_set_voa_em23_peak_current();
  - sl_status_t sl_efp_set_vob_em01_peak_current();
  - sl_status_t sl_efp_set_vob_em23_peak_current();

13.6 Removed Items

None

14 Other Gecko Platform Software Components

14.1 New Items

Added in release 4.1.0.0

- Updated IAR compiler to version 9.20.4.
- Updated GCC compiler to version 10.3-2021.10.

14.2 Improvements

Changed in release 4.1.1.0

- Improved C++ compilation by adding appropriate extern “C” statements in many SDK files.

Changed in release 4.1.0.0

- Removed usage of no-builtin compiler flag option.
- Added ASSERT in linker script to validate size of data section in FLASH.

14.3 Fixed Issues

Fixed in release 4.1.0.0

ID #	Description
845280	Fixed sl_malloc to be called from C++.
828707	Fixed an issue in SVD files where reported RAM size could be off on some parts.
819104	Fixed generation of PRS signal defines in pin_config.h.

14.4 Known Issues in the Current Release

None

14.5 Deprecated Items

None

14.6 Removed Items

None

15 RAIL Library

15.1 New Items

Added in release 4.1.2.0

- Added early support for IEEE802.15.4G dynamic forward error correction PHYs on the EFR32xG12 platform. Use requires help from support to create an appropriate PHY.

Added in release 4.1.0.0

- The RAIL channel of a received packet is now available in the packet's RAIL_RxPacketDetails_t::channel field. This can be of value when scanning or hopping across multiple channels while letting packets accumulate in the receive FIFO for later processing.
- Added the RAIL_ConfigPaAutoEntry API to allow for easier configuration of PA auto mode operation in RAIL.
- Added the RAIL_SetRssiDetectThreshold API to allow the user to detect when the RSSI is at or above a configurable threshold. Once configured, the RAIL_EVENT_DETECT_RSSI_THRESHOLD event can be used to detect when this happens.
- Added support for the MGM240L022RNF module.
- Added support for the FGM230SA27HGN and FGM230SBHGN modules.
- Added the RAIL_GetChannelAlt API. This function returns the channel the radio is currently using. If using DMP and run on the inactive protocol it returns the channel that will be used when next switching to that protocol. When using channel hopping, mode switch, and other features that change channels dynamically this may be different than what is returned by RAIL_GetChannel, as this function will track what channel the radio is actually on at that moment and not what it started on.

15.2 Improvements

Changed in release 4.1.2.0

- Improved PA configurations for the xGM240 modules based on additional test data.

Changed in release 4.1.1.0

- Added support in "RAIL Utility, Coexistence" component for configuring priority options when directional priority is enabled but no static priority GPIO is defined.

Changed in release 4.1.0.0

- The "RAIL Utility, PTI" component will now validate that the correct set of pins are in use for the desired PTI mode.
- RAIL will now error if attempting to start a CSMA or LBT transmit while a scheduled RX is still in progress or vice versa.
- Added PA curves for BGM240P and MGM240P modules.
- Restricted the SL_RAIL_UTIL_PA_RAMP_TIME_US to 10us on some EFR32 modules to match the certification conditions.

15.3 Fixed Issues

Fixed in release 4.1.4.0

ID #	Description
1063152	Fixed an issue where radio reception would not be fully cleaned up when a receive error occurs with receive state transitions set to idle on error but transmit on success, a configuration mostly associated with Bluetooth LE. On the EFR32xG24 this could cause a SYNTH calibration to not be properly restored and eventually cause the radio to stop working.

Fixed in release 4.1.3.0

ID #	Description
1041997	Fixed the librail_config libraries for the following xGM240 modules: BGM240PA22VNA, BGM240PA32VNA, BGM240PA32VNN, BGM240PB22VNA, BGM240PB32VNA, BGM240PB32VNN, MGM240PA22VNA, MGM240PA32VNA, MGM240PA32VNN. Without this update these modules will assert when trying to load the supported BLE and IEEE 802.15.4 PHYs.

Fixed in release 4.1.2.0

ID #	Description
844377	Fixed a Bluetooth LE 2 Mbps AoX issue on EFR32xG24 when using a 38.4 MHz crystal.
1029710	Fixed an issue with RAIL's PA auto mode that would cause it to select an unsupported RAIL_TxPowerMode_t on chip OPNs that are missing the higher power PAs.

Fixed in release 4.1.1.0

ID #	Description
819544	Improved reception on EFR32xG23 and EFR32xG24 when using a PHY with fast detect enable (preamble sense mode).
843708	Moved function declarations from rail_features.h to rail.h to avoid a convoluted include dependency order.
844325	Fixed RAIL_SetTxFifo() to properly return 0 (error) rather than 4096 for an undersized FIFO.
844936	Fixed an issue where using RAIL_SetNextTxRepeat() could cause a brownout reset on EFR32xG23.
853714	Fixed an issue with xGM240 modules causing them to assert during initialization.
988518	Fixed an issue where the radio sequencer would leave portions of the chip enabled after AoX CTE packet reception, preventing the device from going into EM2 sleep and potentially causing missed packet receive events.

Fixed in release 4.1.0.0

ID #	Description
376658	Fixed an issue with the Bluetooth LE coded PHY on EFR32xG21 where a packet received with a corrupt coding indicator would result in an invalid start-of-packet timestamp.
759793	Fixed an issue with Bluetooth LE long-range reception on EFR32xG21 that corrupted packet data and tripped RAIL_ASSERT_FAILED_UNEXPECTED_STATE_RX_FIFO.
772769	Fixed an issue when running IR Calibration on the EFR32xG23 using RAIL_CalibrateIrAlt where we could compute an invalid IRCAL value for certain PHYs and chips.
777427	Fixed support for using the signal identifier CCA modes simultaneously with a user-enabled signal identifier trigger event.
819644	Fixed an issue with frame-type decoding PHYs running at more than 500 kbps on EFR32xG22 and later.
825083	Fixed an issue on EFR32xG23 and EFR32xG24 where PTI could merge multiple receive packets into the same transaction when interrupt latency is significant.
829499	Fixed an issue where RAIL_GetRadioStateDetail would not report the correct state information when frame detection was disabled or during an LBT operation.

ID #	Description
830214	Ensure that the RAIL_RadioConfigChangedCallback_t is called for all RAIL handles in a dynamic multiprotocol application where multiple handles use the same underlying PHY configuration.
835299	Fixed an issue with dynamic handling of whitening and FCS in FSK when only RAIL_IEEE802154_E_OPTION_GB868 was enabled.
844600	Fixed an issue of not being able to receive packets during a RAIL_ScheduleRx configured with a zero relative start time when Power Manager sleep is enabled and configured with an EM2 or lower energy requirement.

15.4 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: • 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	
1019590	When using the “RAIL Utility, Coexistence” component with Bluetooth LE the sl_bt_system_get_counters() function will always return 0 for GRANT denied counts.	Contact support for a patch to coexistence-ble.c to fix this issue.

15.5 Deprecated Items

None

15.6 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 LABS

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

www.silabs.com