Platform 5.1.1 GA

Simplicity SDK Suite 2024.12.1

February 5, 2025

The 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 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 NCP and RCP solutions.
- **Common** components are used throughout the SDKs.
- **Middleware** includes the Capacitive Sensing Firmware Library and the GLIB graphics library.
- **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 2 and later devices.
- **MVP Math Library** includes *Real* and *Complex* Matrix and Vector operations using the Matrix Vector Processor available on EFR32xG24. The library is an alternative to CMSIS-DSP for Matrix and Vector math operations.
- **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):

Platform 5.1.1 released February 5, 2025
Platform 5.1.0 released December 16, 2024

KEY FEATURES

CMSIS Device

- Added support for new EFR32xG27 OPN, EFR32xG26 OPN's, Module OPN's: MGM260PB22VNA2, MGM260PB32VNA2, MGM260PB32VNN2.

CPC

- Fixed a potential security vulnerability.

Security

- Mbed TLS upgraded to latest version 3.6.2.

Boards and External Devices

- Added support for board OPN's BRD2709A, BRD2711A, BRD4350A, BRD4351A, BRD4412A, BRD4413A.

RAIL 5.1.0

- Added support for Bluetooth Low Energy (BLE) Channel Sounding (CS) on the EFR32xG24.

  This feature enables accurate distance measurement between two devices when used with the Silicon Labs BLE stack and distance measurement libraries.
- Added support for concurrent listening feature on the EFR32xG26 part.
- Added support for 1 Mbps and 2 Mbps custom 802.15.4 SUN PHYs on the EFR32xG26 part.

Contents

1 New Items

None.

2 CMSIS Device

2.1 New Items

Added in release 5.1.1

Added support for the following module OPN's:

- BGM260PB22VNA2
- BGM260PB32VNA2
- MGM260PB32VNA5
- MGM260PB32VNN5
- MGM260PB22VNA5
- MGM260PD32VNA2
- MGM260PD32VNN2
- MGM260PD22VNA2

Added in release 5.1.0

- Added the following: EFR32xG27 OPN: EFR32BG27C320F768IJ39
- Added support for the following module OPN's:
  - MGM260PB22VNA2
  - MGM260PB32VNA2
  - MGM260PB32VNN2
- Added the following EFR32xG26 OPNs:

  EFR32MG26B211F3200IM48, EFR32MG26B221F3200IM48, EFR32MG26B211F2048IM68, EFR32MG26B221F2048IM68, EFM32PG26B301F2048IM68, EFM32PG26B101F512IM68, EFM32PG26B301F1024IM68, EFR32MG26B311F3200IL136, EFM32PG26B301F2048IL136, EFM32PG26B101F512IL136, EFM32PG26B301F1024IL136, EFR32BG26B311F2048IM48, EFR32BG26B321F2048IM48, EFR32BG26B311F2048IM68, EFR32BG26B321F2048IM68, EFR32BG26B311F2048IL136, EFR32BG26B311F1024IM68, EFR32BG26B321F1024IM68, EFR32BG26B311F1024IL136, EFR32MG26B411F3200IM48, EFR32MG26B421F3200IM48, EFR32MG26B511F3200IM48, EFR32MG26B521F3200IM48, EFR32MG26B411F3200IM68, EFR32MG26B421F3200IM68, EFR32MG26B511F3200IM68, EFR32MG26B521F3200IM68, EFR32MG26B511F3200IL136, EFR32BG26B511F3200IM48, EFR32BG26B511F3200IM68, EFR32BG26B511F3200IL136, EFM32PG26B501F3200IM68, EFM32PG26B501F3200IL136, EFM32PG26B501F3200IM48, EFM32PG26B500F3200IM48, EFR32BG26B411F3200IM48, EFR32BG26B421F3200IM48, EFR32BG26B410F3200IM48, EFR32BG26B420F3200IM48

2.2 Improvements

Changed in release 5.1.0

None.

2.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1321003	Fixed missing HFRCO bus clock enable that could lead to bus faults in SystemHFRCODPLLClockGet() and SystemHFRCOEM23ClockGet().

Fixed in release 5.1.1

ID #	Description
1391130	Changed the license text in some CMSIS headers from MSLA to Zlib. (Affected headers are <family>_dma_descriptor.h, <family>_ldmaxbar_defines.h, <family>_prs_signals.h). Removed MSLA license text from startup_<family>.c source file as Apache 2.0 license text is already there.

2.4 Known Issues in the Current Release

None.

2.5 Deprecated Items

Deprecated in release 5.1.0

None.

2.6 Removed Items

Removed in release 5.1.0

GCC and IAR specific assembly startup files have been removed. The C startup file should be used instead.

3 Peripherals

3.1 New Items

Added in release 5.1.0

None.

3.2 Improvements

Changed in release 5.1.0

- em_acmp: Added the ACMP_OutputGet function to get ACMP output value.
- em_ldma: Removed unnecessary assert in LDMA_Init and LDMA_StartTransfer.

3.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1307127	em_emu: Fixed a bug in EMU_RamPowerDown() function leading to not covering all RAM shutdown cases for series 2.
1333169	Fixed rare race-condition where DMADRV could return a transfer count of 0 instead of 1 when interrupts are masked and the channel IF is set.
1337998	em_ldma: On EFR32XG25 devices, the MATCHEN and MATCHVAL fields in the LDMA descriptors described in the em_ldma driver were inverted. This issue was fixed so that the structure properly represents the EFR32XG25 hardware.
1260083	em_prs: Added missing PRS asynchronous channels in header file for each channel in xG24 and xG26 family. Added missing channels definition in em_prs.h.
1344396	Added the following headers after they were accidentally removed from the SiSDK: peripheral_dcdc_coulomb_counter.h, peripheral_dcdc_coulomb_counter_compat.h, peripheral_etampdet.h, peripheral_etampdet_compat, peripheral_keyscan.h, peripheral_keyscan_compat.h, peripheral_sysrtc.h and peripheral_sysrtc_compat.h.
1357932	em_emu: Fixed default EM0/1 default peak current for EFR32xG27 Boost parts. The previous value was emuDcdcBoostEM01PeakCurrent_Load25mA. It has been corrected to emuDcdcBoostEM01PeakCurrent_Load23mA.
1286382	em_rmu: Fix compiler warnings for analyzer-null-dereference. em_bus: Fix compiler warnings for analyzer-shift-count-overflow.

Fixed in release 5.1.1

ID #	Description
1363972	em_iadc: Added missing input for iadcPosInputVbat and iadcNegInputVbat for EFR32xG27 devices.

3.4 Known Issues in the Current Release

None.

3.5 Deprecated Items

Deprecated in release 5.1.0

None.

3.6 Removed Items

Removed in release 5.1.0

- em_dbg: Removed DBG_DisableDebugAccess() as it was not available for Series 2 devices.

4 Drivers

4.1 New Items

Added in release 5.1.0

None.

4.2 Improvements

Changed in release 5.1.0

None.

4.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1361482	UARTDRV: Fixed a bug in UARTDRV_Abort() where the callback function was called passing a NULL pointer instead of a pointer to the data buffer.
1284781	UARTDRV: Fixed issues in UARTDRV_Abort() leading to incorrect number of EM1 requirements. Fixed handling of enableRxWhensleeping.
1332867	DMADRV: Fixed a rare issue where DMADRV_TransferRemainingCount could return 0 when the DMA had a single byte to transfer. This happened on systems with high IRQ latency, or that called the API in an atomic/critical section.
1290690	Spidrv: Fixed issue of wrong order of operations for Series 2 devices, which leads to incorrect pin configuration initialization, by enabling routing after configuring pins in SPIDRV_Init function.
1321680	Spidrv: Fixed issue of SPIDRV instantiation using USART on EFR32XG25 despite the absence of USART on these platforms.
1173169	NVM3 optimization for cache search is added, which can be enabled using configuration NVM3_OPTIMIZATION.

4.4 Known Issues in the Current Release

None.

4.5 Deprecated Items

Deprecated in release 5.1.0

None.

4.6 Removed Items

Removed in release 5.1.0

None.

5 Services

5.1 New Items

Added in release 5.1.0

- Clock Manager: Added an AUTO option to enable the HFXO and to use it as the default high frequency clock when radio is used (i.e. when RAIL library is present in the project).
- Clock Manager: Added support for CLKIN0 to the Clock Manager. The CLKIN0 was added to the SYSCLK, DPLL, and EM01GRPBCLK clock branch as a reference option in the sl_clock_manager_tree_config.h and sl_clock_manager_oscillator_config.h configuration file. A SL_CLOCK_MANAGER_CLKIN0_FREQ configuration and a pintool annotation were added to configure a GPIO to receive an external clock and use it for the above references.

Added in release 5.1.1

- Clock Manager: Added a new component clock_manager_oscillator_calibration_override that will override default oscillator calibration values with calibration values written to NVM3. For now, override calibration is only supported for the HFXO oscillator. New API's provided via the component allow users to read, write, and delete HFXO oscillator override calibration from NVM.
- NVM3: NVM3 secure writes and reads are enabled, by default, for Series-3 devices. Migrating from earlier versions of NVM3 where data was stored in plain text requires erasing of NVM3 region for Series-3 devices.

5.2 Improvements

Changed in release 5.1.0

- Sleeptimer: Changed type of overflow_counter variable to prevent the variable from overflowing when the peripheral used by sleeptimer is clocked out of a high frequency clock.
- Clock Manager: The default value of SL_CLOCK_MANAGER_PCLK_DIVIDER is now 1 on Series 2 devices to align with the default value that was present in device_init.

Changed in release 5.1.1

- Clock manager: Added an option to enable or disable the use of the CTUNE from the Manufacturing tokens/User Data page for LFXO and HFXO.

5.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1308159	Memory manager: Fix bug in realloc algorithm making sure metadata of free blocks is updated correctly. This will avoid running out of memory when using realloc() when it is actually still available. It also fixes issues related to the usage of reservations, as heap can be corrupted when using reservations. The fix makes sure reservation offsets are considered in alloc and free, realloc, reserve and release blocks.
1181978	Memory Manager: Fixed an issue where SLI_BLOCK_LEN_BYTE_TO_DWORD() would sometimes return the wrong number of words. Allocations made with an alignment that is not mulitples of 4 could, in a rare scenario, overlap each other.
1346413	Memory Manager: Fixed heap corruption in sl_memory_realloc() when reducing an allocated block.
1354963	Memory manager: Fixed issue related to statistic feature. The heap high watermark was overflowing. The fix makes sure the heap high watermark is always accurate.
1323169	Memory Manager: Added a missing closure of critical section within the function sl_memory_free() in an early exit path. An issue could arise if sl_memory_free() is called twice with the same block leading to an infinite atomic section.
1324463	sl_mpu: Fixed a bug where the first bytes of the .data section would still be marked as executable after calling sl_mpu_disable_execute_from_ram(). The fix consists in properly aligning the end of the ramfunc section (it must be 32 bytes aligned in case the MPU is used).

1251562	HFXO Manager: Fixed a corner case where the HFXO startup time measurement could be started with a wrong start measurement leading to a bad value. In a previous version of the SiSDK, the bad start-up values were discarded but the assertion in sl_hfxo_manager.c - Line 199 would fail. This correction fixes the root cause of this issue.
1305241	Fixed the FreeRTOS Power Manager/Sleeptimer integration code where the number of elapsed OS ticks during sleep was not evaluated properly, leading to the unnecessary creation of sleeptimers with very small timeout values.
1336345	Power manager: Fixed an issue where removing a requirement that was not added previously would lead to an assertion failure.
1336702	Power Manager: Fixed an issue where interrupts could be delayed until the next interrupt cycle due to a missing ISB instruction in the critical section of the power manager.
1290210	Power manager: Fixed issue of sl_power_manager_init not enabling GPIO bus clock. The GPIO bus clock may not be initialized if the project doesn't have a component that initializes it. Added sl_clock_manager_enable_bus_clock(SL_BUS_CLOCK_GPIO) function in sl_power_manager_init() to initialize the GPIO bus clock.
1289318	Sleep Timer: Defined module name for the Power Manager debug feature to correctly display the requirement set by the sleep timer.
1336478	Sleep Timer: Fixed an issue where the callback would be incorrectly called when the callback execution time exceeded the period of a periodic timer.
1356508	Clock Manager: Fixed issue where invalid LFXO and HFXO CTune MFG tokens caused clock_manager_init to attempt to initialize the LFXO and HFXO improperly on Series 2.
1351026	Clock Manager: Fixed an issue leading to an unused variable warning related to rfpll configuration.
1356859	Clock Manager: Added missing VDAC1 clock initialization for EFR32xG24 and EFR32xG26 devices.
1312889	Clock Manager: Fixed a reference to cmuClock_LCDCLK undeclared for the EFR32SG23B020F512IM40 device leading to a compilation error.
1272260	Updated documentation to make it easier for the customers to update the iostream handles for CLI.

5.4 Known Issues in the Current Release

None.

5.5 Deprecated Items

Deprecated in release 5.1.0

- The sl_ram_interrupt_vector_init component is deprecated. Interrupt Manager should be used instead.

5.6 Removed Items

Removed in release 5.1.0

None.

6 CPC

6.1 New Items

Added in release 4.6.0

- Introduced experimental support for NETLINK-SDIO on the SiWx917m platform.

6.2 Improvements

Changed in release 4.6.0

- Increased the maximum supported SPI USART bitrate, allowing the Host to configure higher data transfer speeds for improved performance.
- Addressed a few minor static code analysis issues.
- Enhanced CPCd error handling for improved management of invalid packets received.
- Refined the error messages printed by CPCd for improved clarity.

6.3 Fixed Issues

Fixed in release 4.6.1

ID #	Description
1329252	A security issue has been identified and fixed in the SPI interface, preventing a data leak vulnerability that could have allowed an attacker to receive information stored on the stack.
1355407	Fixed the CPC version reported and displayed by CPCd.

Fixed in release 4.6.0

ID #	Description
1378389	Resolved an issue in CPC Journal where the ID and timestamp were displayed incorrectly, ensuring accurate data representation.
1355407	Fixed an issue where some Linux SPI drivers would skip over a transfer of 0 bytes. This caused CPCd and secondary to lose synchronization and cause a “The IRQ it stuck abnormally long in the de-asserted state.”
1319502	Resolved an issue with the CPCd binary included in the OpenWRT asset.
1307636	Fixed a rare race condition in the SPI driver on systems with high interrupt latency which could lead to CPC system losing communication with the host.

6.4 Known Issues in the Current Release

- The sample app `CPC Secondary on Micrium OS` generates warnings at link time when LTO (Link Time Optimization) is enabled.
- CPCd NETLINK-SDIO is stable on Linux kernel v4.x but may experience stability issues on Linux kernel v6.x.

6.5 Deprecated Items

Deprecated in release 4.6.0

None.

6.6 Removed Items

Removed in release 5.1.0

None.

7 Common

7.1 New Items

Added in release 5.1.0

- Event System: Added features to unregister and free a publisher context.
- Event System: Added a feature to unsubcribe from events from a given event class.
- Event System: New sl_event API 'sl_event_queue_delete()', that accepts an event queue as a parameter, safely deletes the queue. In the process of deleting the event queue, all events that the queue was subscribed to will be unsubscribed from.
- Event System: Added a feature (sl_event_publish_static) which allows a publisher to publish events using a pre-allocated event structure.

7.2 Improvements

Changed in release 5.1.0

- Event System: Improved error handling when pushing events onto subscriber queues. Publishing an event to a subscriber queue should not be considered an error from the point of view of the publisher.

7.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1379854	Event System: Added a missing memory_manager dependency to the slcc component which could lead to compilation errors under certain circumstances.
1351448	The GNU linker template was modified to move the `.noinit` section after the `.bss` in order to fix the problem where the bootloader's `.bss` section overlaps with the `.noinit` section of the application, and initializes those variables to zero when they should not be initialized at all.

Fixed in release 5.1.1

ID #	Description
1383487	Fixed the CMSE option for CMake support with IAR compiler. This change replaces specific CMSE toolchain options with a generic compiler option.
1345860	Pin Tool: Fixed an issue where SPI signals were not displayed in the pin tool U.

7.4 Known Issues in the Current Release

None.

7.5 Deprecated Items

Deprecated in release 5.1.0

None.

7.6 Removed Items

Removed in release 5.1.0

None.

8 Middleware

8.1 New Items

Added in release 5.1.0

None.

8.2 Improvements

Changed in release 5.1.0

None.

8.3 Fixed Issues

Fixed in release 5.1.0

None.

8.4 Known Issues in the Current Release

None.

8.5 Deprecated Items

Deprecated in release 5.1.0

None.

8.6 Removed Items

Removed in release 5.1.0

None.

9 Security

9.1 New Items

Added in release 5.1.0

- Upgrade to Mbed TLS version 3.6.2 (latest).
  - Mbed TLS 3.6 is a long-term support (LTS) branch. It will be supported with bug-fixes and security fixes until at least March 2027.
  - Release Notes can be found at: https://github.com/Mbed-TLS/mbedtls/releases.
- Add SLC component to support Deterministic ECDSA using the PSA Crypto API.
  - User documentation can be found at: https://arm-software.github.io/psa-api/crypto/1.1/api/ops/sign.html.

9.2 Improvements

Changed in release 5.1.0

- Upgraded the Legacy Deterministic ECDSA component using the legacy Mbed TLS API to production quality. Users are recommended to use the corresponding PSA Crypto component for Deterministic ECDSA, since the legacy crypto APIs of Mbed TLS will be obsoleted in Mbed TLS 4.0, which is planned to be released in 2025 by Trusted-Firmware.org.
- Added call to SE Manager initialisation function sl_se_init in sl_system_init. This means applications that calls sl_system_init during initialization will not need to call sl_se_init explicitly. The sl_system_init is recommended to be included in the application configuration.
- Added support for calling SE Manager APIs from critical sections. In general, it is recommended to avoid calling the SE Manager from critical sections.

9.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1363898	Fixed the Mbed TLS ALT plugin for the API function called mbedtls_ecdsa_verify, which allowed signature components (r and s) to be negative. All SDK versions are affected. All xG2x device families are affected.
1361396	Fixed a build error in function sli_cryptoacc_driver_single_shot_pbkdf2, which is called by the PSA Crypto API function, sl_psa_key_derivation_single_shot, on VSE devices when the user selects algorithm PSA_ALG_PBKDF2_AES_CMAC_PRF_128. Affected device is xG27. Affected SDK versions are 4.4.x (23Q4) and sisdk-2024.6.x.
1354904	Fixed bug in PSA AEAD drivers which did not check output buffer size before processing context saved data. This may cause buffer overflow. Both HSE and VSE PSA AEAD drivers are affected, meaning all EFR32xG2x devices are affected. Affected SDK versions are 4.0.0 (21Q4) and later.
1378069	Fixed bug in PSA AEAD driver for VSE devices, where the function sli_cryptoacc_transparent_aead_update (called by API function psa_aead_update) did not release a mutex (used to apply mutual exclusion of CRYPTOACC hardware) before returning when the algorithm is not supported, and the algorithm is CCM and there is 16 bytes in the last AES block. This will cause all PSA VSE drivers to fail in the next calls to any PSA Crypto APIs that need to access the CRYPTOACC accelerator hardware). Affected devices are xG22 and xG27. Affected SDK versions are 4.0.0 (21Q4) and later.

Fixed in release 5.1.1

ID #	Description
1284341	Addressed a security issue in Mbed TLS prior to version 3.6.0 (not inclusive), by adding a new config option SL_MBEDTLS_PSA_ASSUME_EXCLUSIVE_BUFFERS to allow users to configure the corresponding config option called MBEDTLS_PSA_ASSUME_EXCLUSIVE_BUFFERS which is new in Mbed TLS 3.6. This option may be enabled (set to 1) if all buffers passed to any PSA function reside in memory that is accessible only to the PSA function during its execution. This option MUST be disabled (set to 0) whenever buffer arguments are in memory and shared with an untrusted party, for example where arguments to PSA calls are passed across a trust boundary like TrustZone. The default mode of MBEDTLS_PSA_ASSUME_EXCLUSIVE_BUFFERS is undefined to prepare for security by default. However, many non-TrustZone applications that are short on FLASH and RAM capacity need to consider whether or not to define this config option since it consumes extra FLASH space (added code size) and heap (in RAM) for dynamic memory allocation of input/output buffers in the PSA Crypto library. The default value of SL_MBEDTLS_PSA_ASSUME_EXCLUSIVE_BUFFERS is 0 (security by default). To enable, set SL_MBEDTLS_PSA_ASSUME_EXCLUSIVE_BUFFERS to 1. All devices and SDK versions with TrustZone support up to 5.1.0 (not inclusive) are affected.
1392116	Fixed issue in the SE Manager API function called sl_se_validate_key which returned SL_STATUS_OK when the storage buffer pointer of the key descriptor was NULL. This caused subsequent calls to the SE Manager crypto functions to send SE mailbox commands with address zero in a DMA descriptor which would cause a BUS ERROR, and corresponding status being returned. The fix checks and returns SL_STATUS_INVALID_PARAMETER if the buffer pointer is NULL. Affected devices are all xG2x HSE devices. Affected SDKs are all versions since 3.0.0.

9.4 Known Issues in the Current Release

Known Issues in release 5.1.1

ID #	Description

1399683	There is a known issue in the sample application called **psa_crypto_aead** , in the multipart function, when the returned output size is different from the input size in the calls to **psa_aead_update**. The error is triggered when the plaintext fragment size is not a multiple of the standard AES block size (16 bytes). The current version of the sample app only demonstrates plaintext fragment sizes that are multiples of 16 bytes, meaning that the sample app itself is expected to work in the released state. However if a user copies and/or adds plaintext fragment sizes that not multiples of 16 bytes the multipart AEAS function will fail. The error is in the fragmentation loop where the input pointer is not moved correctly if the output size is less than the input size. A fix is to add a new variable to keep track of the total input bytes, see proposed code snippet below:
	
	`size_t in_total = 0;  // new variable`
	`while ( input_len - (frag_size * step) >= frag_size) {`
	`  return_on_error(psa_aead_update(&op,`
	`  // input + out_total, // old code`
	`  input + in_total, // new code`
	`  frag_size,`
	`  output + out_total,`
	`  out_sz - out_total,`
	`  &bytes_out));`
	
	`  // when successful, bytes_out holds the number of actual bytes output as a result`
	`  // of the 'update' call ... which may be different from the number of bytes input`
	`  in_total += frag_size; // new code`
	`  out_total += bytes_out; // accumulate output length...`
	`  step++;`
	`}`

1403803	There is a known issue in the PSA SE AEAD driver that causes the PSA Crypto multipart AEAD API functions, psa_aead_finish and psa_aead_verify to fail unexpectedly and return **PSA_ERROR_HARDWARE_FAILURE**. The error occurs if the output buffer size argument ("out_size") for ciphertext and plaintext, respectively, are bigger than 255 and the result of a bitwise AND operations of **out_size** AND 0x000000FF, is less than the expected output size. The workaround is to pass PSA_AEAD_FINISH_OUTPUT_MAX_SIZE to psa_aead_finish and PSA_AEAD_VERIFY_OUTPUT_MAX_SIZE to psa_aead_verify. Affected devices are all xG2x HSE devices. Affected SDK versions are 5.1.1 and all earlier.

9.5 Deprecated Items

Deprecated in release 5.1.0

All APIs in em_se.h have been deprecated and will be removed in a future release. The functionality has been ported as an internal API under the SE Manager component. All usage of emlib SE should be replaced by calls to the public SE Manager API.

9.6 Removed Items

Removed in release 5.1.0

None.

10 Operating System

10.1 New Items

Added in release 5.1.0

None.

10.2 Improvements

Changed in release 5.1.0

None.

10.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1108940	Fixed a bug in the Micrium CMSIS RTOS 2 layer where a memory leak would happen when a task would call osThreadTerminate on self.

10.4 Known Issues in the Current Release

ID #	Description	Workaround
1166857	OSTaskQPost doesn't send a message to current task when called with a null pointer as argument.	N/A

10.5 Deprecated Items

Deprecated in release 5.1.0

None.

10.6 Removed Items

Removed in release 5.1.0

None.

11 Gecko Bootloader

11.1 New Items

Added in release 5.1.0

None.

11.2 Improvements

Changed in release 5.1.0

None.

11.3 Fixed Issues

Fixed in release 5.1.0

None.

11.4 Known Issues in the Current Release

None.

11.5 Deprecated Items

Deprecated in release 5.1.0

None.

11.6 Removed Items

Removed in release 5.1.0

None.

12 MVP Math Library

12.1 New Items

Added in release 5.1.0

None.

12.2 Improvements

Changed in release 5.1.0

None.

12.3 Fixed Issues

Fixed in release 5.1.0

None.

12.4 Known Issues in the Current Release

None.

12.5 Deprecated Items

Deprecated in release 5.1.0

None.

12.6 Removed Items

Removed in release 5.1.0

None.

13 Examples

13.1 New Items

Added in release 5.1.1

Added support for the following new LCD sample apps:

- segment_lcd_ldma
- segment_lcd_timer

Added in release 5.1.0

None.

13.2 Improvements

Changed in release 5.1.0

None.

13.3 Fixed Issues

Fixed in release 5.1.0

None.

13.4 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.

13.5 Deprecated Items

Deprecated in release 5.1.0

None.

13.6 Removed Items

Removed in release 5.1.0

None.

14 Boards and External Devices

14.1 New Items

Added in release 5.1.1

Added support for the following new OPN's

- BRD2505A
- BRD2713A
- BRD4120A
- BRD4121A
- BRD4402C
- BRD4403C

Added in release 5.1.0

Added support for the following new drivers

- ICM40627- IMU sensor
- SHT4X sensor- Humidity and Temperature sensor

Added support for the following new OPNs:

- BRD2709A
- BRD2711A
- BRD4350A
- BRD4351A
- BRD4412A
- BRD4413A

14.2 Improvements

Changed in release 5.1.0

None.

14.3 Fixed Issues

Fixed in release 5.1.0

ID #	Description
1320271	Microphone: Added EM1 requirement when initializing the microphones to prevent chip from entering Low Energy modes (EM2 through EM4).

14.4 Known Issues in the Current Release

None.

14.5 Deprecated Items

Deprecated in release 5.1.0

None.

14.6 Removed Items

Removed in release 5.1.0

None.

15 Other Gecko Platform Software Components

15.1 New Items

Added in release 5.1.0

None.

15.2 Improvements

Changed in release 5.1.0

None.

15.3 Fixed Issues

Fixed in release 5.1.0

None.

15.4 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
	An issue is detected when updating the GCC toolchain from version 10.3 to 12.2, which increases the RAM usage by about 400 bytes in some cases when using stdio.	--

15.5 Deprecated Items

Deprecated in release 5.1.0

None.

15.6 Removed Items

Removed in release 5.1.0

None.

16 RAIL Library

16.1 New Items

Added in release 5.1.1

- Added RAIL support for the MGM260PD32VNA2, MGM260PD32VNN2, MGM260PD22VNA2, MGM260PB32VNA5, MGM260PB32VNN5, MGM260PB22VNA5, BGM260PB22VNA2 and BGM260PB32VNA2 modules.

Added in release 5.1.0

- Added RAIL_GetRxIncomingPacketRssi to get the RSSI associated with the incoming packet.
- Added support for "RAIL Utility, CS Antenna Offset" component for BLE Channel Sounding feature on the EFR32xG24 that allows customer to configure offsets for a wired or wireless antenna configuration.
- Added support for TrustZone APIs on the EFR32xG23 and EFR32xG28 parts to allow RAIL to access the CMU, EMU, GPIO, LDMA, HFXO, PRS and SYSRTC peripherals when they are running in the secure state despite the RAIL library being non-secure.

16.2 Improvements

Changed in release 5.1.1

None.

Changed in release 5.1.0

- To prepare for RAIL 3.0 specifications, RAIL_Status_t is now sl_status_t and RAIL_STATUS_ values are mapped to similar SL_STATUS_ values.

16.3 Fixed Issues

Fixed in release 5.1.1

ID #	Description
1172597	Fixed an issue with RAIL_RX_CHANNEL_HOPPING_MODE_MANUAL where RAIL_EVENT_RX_CHANNEL_HOPPING_COMPLETE wouldn't be signaled. Added new RAIL_TriggerRxChannelHop() API for manually triggering an RX channel hop, a refinement of RAIL_CalibrateTemp(). Also noted that RAIL_RX_CHANNEL_HOPPING_OPTION_STOP was only intended for testing purposes and is not supported on the EFR32xG21 part.
1349369	Fixed an issue where power restrictions were incorrectly applied through channel configs on the platforms that support RAIL_SUPPORTS_DBM_POWERSETTING_MAPPING_TABLE.
1380153	Fixed an issue where calling RAIL_Idle() could prevent any future reception on the EFR32xG25 for OFDM-FSK concurrent PHYs.
1389189	Fixed a false channel busy issue when automatic LNA bypass is enabled on the EFR32xG25 part.

Fixed in release 5.1.0

ID #	Description
1120643	Fixed an issue where the phy was not re-configured correctly when using the RAIL_ConfigSyncWords API with certain custom modem configurations.
1273595	To better reflect Z-Wave spec naming conventions, renamed RAIL_ZWAVE_REGION_US_LR_END_DEVICE to RAIL_ZWAVE_REGION_US_LR3, RAIL_ZWAVE_REGION_EU_LR_END_DEVICE to RAIL_ZWAVE_REGION_EU_LR3, RAIL_ZWAVE_REGIONID_US_LR_END_DEVICE to RAIL_ZWAVE_REGIONID_US_LR3, and RAIL_ZWAVE_REGIONID_EU_LR_END_DEVICE to RAIL_ZWAVE_REGIONID_EU_LR3.

ID #	Description
1290731	Fixed a multiprotocol issue where RAIL_BLE_SetNextTxRepeat() or RAIL_SetNextTxRepeat() information mistakenly survived across RAIL_Idle() to the surprise of a subsequent non-repeated transmit.
1300509	Fixed an issue with IEEE 802.15.4 High Data Rate (HDR) packets such that PTI now properly indicates they have a 2-byte 802.15.4 PHY header on the EFR32xG26 part.
1306804	Fixed an issue where glitches could occur when using RAIL_RxDataSource_t values other than RX_PACKET_DATA on the EFR32xG23 and newer chips.
1309139	Fixed an issue on the EFR32xG22 and later where RAIL_StopTx() being called before the transmit has started could cause the radio to return to receive with the radio partially misconfigured for transmit.
1309208	Fixed a minor issue on the EFR32xG21 where the TX-to-RX turnaround timing could be a bit longer than specified.
1321076	Fixed an issue with setDebugSignal railtest command that broke DOUT and DCLK signals.
1352393	Fixed an issue on ZGM230S modules where PHY signatures were computed incorrectly, and the device would assert on startup.
1357948	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.
1369901	Fixed an issue where the device would become stuck when running image rejection calibration on an OFDM PHY from the event callback on the EFR32xG25 part.

16.4 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	

16.5 Deprecated Items

Deprecated in release 5.1.0

- The RAIL 2.x API is planned for deprecation in the 25Q2-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.

16.6 Removed Items

Removed in release 5.1.0

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