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):
Gecko Platform 4.2.1.0 released February 1, 2023
Gecko Platform 4.2.0.0 released December 14, 2022
1 CMSIS Device

1.1 New Items

Added in release 4.2.0.0

- Added support for EFR32xG25 devices
- Added support for EFR32xG21 Rev. C devices
- Added support for the BGM220SC23HNA module
- Added support for the MGM240SA22VNAx module
- Added support for the MGM240SD22VNAx module
- Added support for the MGM240SA22VNAx module
- Added support for the BGM240SB22VNAx module
- Added support for the BGM241SD22VNAx module

1.2 Improvements

Added in release 4.2.1.0

- The control bit DONESET in the LDMA_CH_CTRL register was renamed into DONEIEN to be consistent with other EFR32xG2x devices

1.3 Fixed Issues

Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1059733</td>
<td>ARMCC assembly startup files for Series 0 and Series 1 devices have been removed from GSDK since they have not been supported since GSDK v2.7</td>
</tr>
<tr>
<td>1064407</td>
<td>The PLL0, which provides an accurate 48 MHz clock to the USB, has been renamed for USBPLL0 in the EFR32xG25 header files.</td>
</tr>
<tr>
<td>1021831</td>
<td>Added_EMU_RSTCAUSE_SETAMPER bit in device headers of EFR32xG21, EFR32xG23, EFR32xG24 and EFR32xG24 devices. This bit is asserted following a reset due to a tampering event detection.</td>
</tr>
<tr>
<td>1015647</td>
<td>Fixed the EXT_IRQ_COUNT macro value to match the number of external NVIC internal interrupt sources for EZR32HG, EZR32LG and EZR32WG devices.</td>
</tr>
<tr>
<td>1059733</td>
<td>ARM Startup files for Series 0 and Series 1 devices have been removed from GSDK since they have not been supported since GSDK v2.7</td>
</tr>
</tbody>
</table>

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 4.2.0.0

- Added support for EFR32xG21 Rev. C in peripheral drivers (support for IADC internal voltage reference)
- Added support for the ETAMPDET peripheral
- The CMU_CLOCK_SELECT_SET macro was added as an alternative to the function CMU_ClockSelectSet, and may be used to reduce code size.
- Added new DisSyncOut field to TIMER_Init_TypeDef structure. This field can be used to disable the Timer’s ability to start/stop/reload other timers.
- Added APIs to managed EFP interrupt in the EMU driver.
- Added APIs to access the HFXO CTune value in the CMU driver.

2.2 Improvements

Changed in release 4.2.0.0

- Added documentation for EMU_DCDCXxxx() functions
- Improved Eusart_Rx function when to account for Rx FIFO watermark

2.3 Fixed Issues

Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1043460</td>
<td>Fixed IADC_calcTimebase function to use the correct clock source. Time base calculation is now based off the correct clock source CLK_SRC_ADC rather than previous CLK_CMU_ADC. IADC is initialized with the correct time base and it is no longer required to manually set the correct time base value after initializing the IADC.</td>
</tr>
<tr>
<td>1037225</td>
<td>Fixed an issue where the EM4WU GPIO pin mode was configured as InputPull instead of InputPullFilter when configuring the EM4WU External Interrupt.</td>
</tr>
<tr>
<td>1033757</td>
<td>Device can still execute instructions after executing Enter_EM4() function is addressed with a new EMU_EnterEM4Wait() function. It waits after an EM4 entry request and makes sure no other code can run between the request and the moment the CPU is actually shut down.</td>
</tr>
<tr>
<td>1020645</td>
<td>Fixed EMU_BoostExternalShutdownEnable() API, BOOSTENCTRL bit logic was reversed.</td>
</tr>
<tr>
<td>1019940</td>
<td>Fixed EMU_RamPowerDown() end parameter behavior.</td>
</tr>
<tr>
<td>845272</td>
<td>Fixed a bug where the high-accuracy IADC offset correction calculation was wrong in the IADC initialization routine.</td>
</tr>
<tr>
<td>845232</td>
<td>Fix issue where the interrupts of the EFP would still be enabled after de-initializing the EFP driver.</td>
</tr>
<tr>
<td>1022566</td>
<td>Removed dmadrv_baremetal sample application for BRD2204C because it does not use usart0 peripheral for vcom.</td>
</tr>
</tbody>
</table>

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
None

3.2 Improvements
None

3.3 Fixed Issues
Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>811009</td>
<td>Implemented KEYSCAN_E301 fix for EFR32xg2x devices (fix for Keyscan interrupts not cleared on unused rows).</td>
</tr>
</tbody>
</table>

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 4.2.0.0**
- Added a new Device Initialization RFFPLL service to handle RFFPLL peripheral initialization.
- Added new API to iostream: sl_iostream_vprintf()

4.2 Improvements

**Changed in release 4.2.1.0**
- RFFPLL configuration now defaults to use band AUTO band rather than defaulting to Band 9xx MHz in the device_init_rffpll component. This allows picking up the RFFPLL configuration parameters from the radio configurator.
- Added a new API to get the sleeptimer's clock source's accuracy.

**Changed in release 4.2.0.0**
- Added support for 64-bit timestamps in sleeptimer wall clock APIs.
- Added the ability for sleeptimer service to use high frequency TIMER/WTIMER peripherals.
- sl_device_init_hfxo() will now use the CMU_HFXOINIT_EXTERNAL_SINE default settings if the "Device Init: HFXO" component's mode configuration is set to "External sine wave".
- Changed IOStream Uart interfaces reception to use DMA instead of receiving byte per byte.

4.3 Fixed Issues

**Fixed in release 4.2.1.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1078350</td>
<td>Addressed an issue related to increased interrupt latency caused by the integration of IOstream UART DMA.</td>
</tr>
<tr>
<td>1092999</td>
<td>Resolved various static analysis issues identified in the IOstream and CLI components.</td>
</tr>
<tr>
<td>1081342</td>
<td>Corrected an issue where initial SWO data sent immediately after initializing SWO would become corrupted.</td>
</tr>
</tbody>
</table>

**Fixed in release 4.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>852037</td>
<td>Periodic sleeptimers no longer drift over time when using milliseconds as the time base.</td>
</tr>
<tr>
<td>1019822</td>
<td>Fixed sl_sleeptimer_get_datetime() so it returns time in Standard Time to match sl_sleeptimer_set_datetime() which sets the timezone.</td>
</tr>
<tr>
<td>1073997</td>
<td>Fixed sl_sleeptimer_convert_unix_time_to_ntp() to return an error when invalid time parameter is passed.</td>
</tr>
<tr>
<td>1068039</td>
<td>Fixed internal HFXO Manager settings applied when enabling SL_HFXO_MANAGER_SLEEPYCRYSTAL_SUPPORT.</td>
</tr>
<tr>
<td>818102</td>
<td>Updated critical section used in Power Manager to explicitly use PRIMASK. This fixes an issue where the CPU would not wake up from sleep if em_core critical section are configured to use BASEPRI.</td>
</tr>
<tr>
<td>937666</td>
<td>Fixed IOStream Flush UART functionality for EUSART peripheral.</td>
</tr>
<tr>
<td>1065570</td>
<td>Fixed a bad usage of an enum during the preprocessor into the IOStream UART component.</td>
</tr>
</tbody>
</table>
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 4.2.1.0**
- The bind argument (-b) can now be used together with restart argument (-r) to restart cpcd after a successful bind as well as restart cpcd if it was previously bound with a secondary.

**Added in release 4.2.0.0**
- New CPC Protocol version; CPCd and Secondary must be updated.
- Encryption can be disabled per endpoint; dictated by the secondary.
- Added new functionality to retrieve libcpc the firmware application version and the library version.
- Added new asynchronous event notification mechanism to libcpc.
- Added new API in libcpc to set endpoint options with type safety.
- libcpc’s wrapper in Rust.
- libcpc instances are limited to one per task.

5.2  Improvements

**Changed in release 4.2.1.0**
- Improved handling for missing bootloader in firmware update mode.
- Added functionality to the CPC daemon for automatically generating a plaintext binding key if one is not present before binding.

**Changed in release 4.2.0.0**
- Improved CPC Secondary UART drivers’ reliability and recovery when the hardware flow control is disabled.
- libcpc had a few changes when errors occur. Instead of returning -1 and setting “errno”, it now returns negative values; negative values of “errno”.
- Few improvements to cpcd cmake file; how to retrieve mbedTLS library and default compiler flags.
- Allow overriding socket directory / config path via CMAKE.
- Reduced minimum I-Frame retransmit timeout which also increase overall performance.
- Enhanced the tracing of the CPCd EZSP driver, used to send a firmware update to a CPC secondary using the Ember bootloader via SPI.
- CPCd can run on latest version of Android.
- Improved error reporting on bind failure.
- CPCd default binding key file location has been modified for “~/.cpcd/binding.key”.
- Prevent clients from connecting if CPCd is not started in normal mode.
- CPCd displays a warning if run with root privileges.
- Improved CPCd’s error handling when the bootloader interface is not present, and a firmware upgrade is requested.
- Improved CPCd config parser to allow comments on the same line.

5.3  Fixed Issues

**Fixed in release 4.2.1.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1085742</td>
<td>Resolved a crash issue in the secondary caused by receiving a packet before the kernel had started.</td>
</tr>
</tbody>
</table>
Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>839674</td>
<td>Fixed in libcpc, the state returned by cpc_get_endpoint_state() after cpc_endpoint_close() has been called.</td>
</tr>
<tr>
<td>1040127</td>
<td>CPC security failed to initialize on EFR32MG13 and MG14 series parts. To work around this issue, mbedtls_entropy_adc() has been added as entropy source for these parts.</td>
</tr>
<tr>
<td>1057885</td>
<td>Fixed in libcpc, an issue that prevented users to call cpc_restart() again once it failed.</td>
</tr>
<tr>
<td>1064508</td>
<td>Fixed Secondary UART-EUSART driver with Hardware Flow control disabled and the security enabled.</td>
</tr>
<tr>
<td>1066686</td>
<td>Fixed a potential socket leak into cpc_init() when the daemon was not started.</td>
</tr>
<tr>
<td>1068063</td>
<td>Fixed an issue where the CS signal would not be de-asserted after successfully sending an SPI to the secondary device.</td>
</tr>
<tr>
<td>1066273</td>
<td>Fixed an issue where a retransmit from the secondary could cause an endpoint to close because the decryption would fail.</td>
</tr>
<tr>
<td>1067736</td>
<td>Fixed race condition with security endpoint when the client is actively connecting.</td>
</tr>
</tbody>
</table>

5.4 Known Issues in the Current Release

None

5.5 Deprecated Items

None

5.6 Removed Items

None
6 Common

6.1 New Items
None

6.2 Improvements

Changed in release 4.2.0.0
- Linker sections are sorted by alignment to improve memory usage

6.3 Fixed Issues

Fixed in release 4.2.1.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1092995</td>
<td>Dedicated RAM section for SEGGER_RTT moved from the .bss section in the GCC linker script to the .data section. SEGGER_RTT section in .bss was causing .bss to be considered as initialized data and increased flash memory usage.</td>
</tr>
</tbody>
</table>

6.4 Known Issues in the Current Release
None

6.5 Deprecated Items
None

6.6 Removed Items
None
7 Middleware

7.1 New Items
None

7.2 Improvements

**Changed in release 4.2.0.0**
- Updated GLIB components to allow creation or use of a custom display implementation.
- Added configuration file to GLIB to reduce the memory footprint.

7.3 Fixed Issues

**Fixed in release 4.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1015841</td>
<td>Fixed C++ compiler error with Glib.</td>
</tr>
</tbody>
</table>

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 4.2.0.0

- Updated the Mbed TLS library to version 3.2.1 with support for Certificate-Based Authentication & Pairing (CBAP).
- Added support for the TrustZone PSA initial attestation service. The Secure Key Library PSA attestation private key is stored in ITS with the UID 0xFFFFFFFFFFFF for Vault Mid devices. Hence, this UID should not be used for other purposes when the attestation service is enabled.
- Added RTOS support for the TrustZone Secure Key Library (TZ SKL) and TrustZone Gecko Bootloader. The TrustZone NSC interfaces support only one single thread to enter the secure libraries at any time by using a mutex, implemented with the CMSIS RTOS2 API. That is, there is no multi-thread concurrency allowed inside the secure libraries. The NonSecure application can run multiple threads concurrently. Currently FreeRTOS and MicriumOS are supported.
- Added ETAMPDET tamper source to SE manager for xG25.
- Added sl_se_get_tamper_reset_cause() function to SE Manager, available for Secure Vault High devices. The function lets the user find out if the last reset was caused by a Tamper event and which Tamper source caused the reset. Requires SE firmware version 2.2.1.
- sl_se_get_otp_version() is now available for VSE devices (as well as HSE devices). Requires VSE firmware version 2.2.1 / 1.2.14.
- Added support for non-blocksize input for AEAD multipart (fixed in GSDK version 4.1.1.0)
- Added support for partially overlapping input buffers for AEAD multipart (fixed in GSDK version 4.1.1.0)
- Added driver support for CCM multipart on xG21 devices (fixed in GSDK version 4.1.1.0)

8.2 Improvements

Changed in release 4.2.0.0

- Added a new version of the ITS driver (v3) in order to optimize the key look up time, especially for applications with more than 100 keys. ITS V3 uses simple PRNG with UID as a seed to create a hash-map-like structure. Note that NVM3 ID range was changed from region starting at 0x83100 to region ending with 0x870ff. We provide an upgrade path for the devices that already use the ITS driver. Note that after upgrading to ITS driver V3, SL_PSA_ITS_USER_MAX_FILES cannot be changed. Also note, that upgrade can take some time (more than 1min for 1000 files). New config options were added:
  - SL_PSA_ITS_SUPPORT_V2_DRIVER - This can be set to 0 if the device has never used ITS driver before in GSDK 4.1.x and earlier, or the keys have been already migrated to v3, and remove the unused code.
  - SL_PSA_ITS_SUPPORT_V3_DRIVER - This can be set to 1 if the device has never used ITS driver before in GSDK 4.1.x and earlier. This configuration option will add support for V3 driver, and provide an upgrade path from V1/V2. For devices that are not able to upgrade to ITS driver V3, we recommend disabling it, and using the previous version of the ITS driver.

  The downgrade path of the ITS driver to an earlier version is not provided.

- PSA Crypto components have now been updated to reflect the addition of Curve25519 support on xG21A and xG23A devices. Users that depend on this functionality and are running SE firmware versions older than 1.2.10 and 2.1.7, respectively, are recommended to update their firmware (preferred) or manually add the “Platform\Security\Mbed TLS\Elliptic Curves\Curve25519” component to their project.
- Added components for use of PKCS #5 and #12 modules in Mbed TLS
- Add support for configuring dynamic memory allocation functions in Mbed TLS via SLC components.
- Add support for configuration of alternative threading support in Mbed TLS (aka MBEDTLS_THREADING_ALT) via SLC components.
- Changed the TrustZone root key automatic key renewal. The key is now only renewed on Device Erase.
- TRNG handling logic changed to reduce idle current consumption for xG22 in EM0/1.
- Added PSA API documentation links to Mbed TLS SLC components
8.3 Fixed Issues

Fixed in release 4.2.1.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>421759</td>
<td>Fixed EMLIB functions CRYPTO_SHA_1 and CRYPTO_SHA_256 to support input buffer pointers that are not aligned on 32-bit words. Affected devices are EFR32xG1x and EFM32xG1x. Affected SDK versions are 2.6.0-2.6.5, 2.7.0-2.7.11, 3.0.0-3.0.2, 3.1.0 - 3.1.2, 3.2.0 - 3.2.3, 4.0.0 - 4.0.2, 4.1.0 - 4.1.3 and 4.2.0</td>
</tr>
<tr>
<td>691903</td>
<td>Removed redeclarations of variables with the same name in the function sl_se_gcm_multipart_starts in the SE Manager component. Affected devices are EFR32xG21. Affected SDK versions are 3.1.0 - 3.1.2, 3.2.0 - 3.2.3, 4.0.0 - 4.0.2, 4.1.0 - 4.1.3 and 4.2.0</td>
</tr>
</tbody>
</table>

Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1074038</td>
<td>The AES-CTR one-shot encryption function of the PSA SE driver (sl_se_driver_cipher_encrypt) may fail when operating on input data that are not multiples of 16 bytes, and the alg argument is PSA_ALG_CTR or PSA_ALG_CCM_STAR_NO_TAG. The cipher oneshot encryption function of the PSA Crypto API psa_cipher_encrypt generates random IV values and subsequently calls sl_se_driver_cipher_encrypt. The failure depends on the combination of cipher key and random IV, where some random IV values may cause an incorrect ciphertext. The bug does not have exploitable properties and is not considered to be a security vulnerability. Affected devices are HSE-based devices (series-0/1 and VSE not affected). Affected SDK versions are 3.1.0 - 3.1.2, 3.2.0 - 3.2.3, 4.0.0 - 4.0.2 and 4.1.0 - 4.1.3</td>
</tr>
<tr>
<td>937813</td>
<td>Fixed an issue when the application requests random data and the radio is off, uninitialized or malfunctioning, then no entropy is available for the entropy accumulator of Mbed TLS and consequently the PRNGs (CTR-DRBG and HMAC-DRBG). The fix implements a fallback to request entropy from the ADC if the radio returns 0 bytes of entropy. The prerequisite is that the user must configure the ADC as entropy source. Affected devices are EFR32xG1x. Fixed in GSDK version 4.1.1.0</td>
</tr>
<tr>
<td>845005</td>
<td>Fixed issue on EFR32xG23 when DCI became unresponsive after running OTP initialization and Power-on-reset. In API function sl_se_init_otp reject setting reserved tamper sources to tamper levels other than &quot;ignore&quot; during OTP initialization by system manufacturer. Fixed in GSDK version 4.1.1.0</td>
</tr>
</tbody>
</table>

8.4 Known Issues in the Current Release

Issues in bold were added since the previous release.

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
</table>
|       | The platform security software support has been moved from the folder util/third_party/crypto to platform/security. The submodules mbedtls and trusted-firmware-m are now located in util/third_party/.

This should not affect any projects that are using SLC components. However, if your project is not based on SLC components (which can be the case especially for projects that were created before Gecko SDK v3.0), you may need to update the source file locations in the project files.

8.5 Deprecated Items

None
8.6 Removed Items

Removed in release 4.2.0.0

- Remove deprecated multipart APIs with tag SL_DEPRECATED_API_SDK_3_3:
  - Deprecated CMAC multipart API has been removed.
  - Deprecated GCM multipart API has been removed.
  - Deprecated HASH multipart API has been removed.
9 Operating System

No changes
10 Gecko Bootloader

10.1 New Items
None

10.2 Improvements

**Changed in release 4.2.0.0**

- Added automated support for executing post-build actions.
- Added support for upgrading from storage slots in a sequential order in a multi-slot configuration in an absence of a bootloader list as opposed to an early exit. A configuration option is available in the Bootloader Storage Slot Setup component to enable/disable validation of the slot layout in case of a multi-slot configuration.

10.3 Fixed Issues

**Fixed in release 4.2.1.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>293766</td>
<td>Added fix for Bootloader custom projects app to show error when predefines pins are not defined</td>
</tr>
</tbody>
</table>

**Fixed in release 4.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1043067</td>
<td>Added fix for XModem idle Timeout macro set to a non-default value in Bootloader - UART XMODEM sample app.</td>
</tr>
<tr>
<td>00289194</td>
<td>Added fix for Bootloader - SoC Bluetooth in-place OTA DFU application to display only for series 1 devices in Simplicity Studio.</td>
</tr>
<tr>
<td>00286863</td>
<td>Fixed an issue where the upgrade image size calculated was off by 4 bytes due to the CRC32 checksum that got appended to the image after the signature (fixed in version 4.1.1.0).</td>
</tr>
</tbody>
</table>

10.4 Known Issues in the Current Release
None

10.5 Deprecated Items
None

10.6 Removed Items
None
11 Machine Learning

11.1 New Items
None

11.2 Improvements

Changed in release 4.2.0.0
- Added support for more tflite micro operators in the slc generator script that autogenerates the OpResolver code.

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 ml_blink example showing a minimal tflite micro example blinking a led using a machine learning model. Meant to be used as a starting point for machine learning development.
- Added ml_magic_wand example. This example is based on a TensorFlow Lite micro example, but with an improved model to make gesture detection better.

12.2 Improvements

- Fixed suppression issue in voice_control_light and audio_classifier example, improving the classification stability.

12.3 Fixed Issues

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1075130</td>
<td>Fixed issue in the ml_magic_wand example where the tflite arena size was too small.</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
<th>Workaround</th>
</tr>
</thead>
<tbody>
<tr>
<td>664803</td>
<td>Se_manager and psa_crypto sample apps do not work correctly in Simplicity Studio 5’s launch console.</td>
<td>In the launch console, change the line terminator selection to None.</td>
</tr>
</tbody>
</table>

12.5 Deprecated Items

None

12.6 Removed Items

- Removed tflite_micro_hello_world example, this example was replaced by the ml_blink example.
- Removed tflite_micro_magic_wand example, this example was replaced by the ml_magic_wand example.
13 Boards and External Devices

13.1 New Items

Added in release 4.2.1.0
Added support for the following new OPNs:
- BRD2704A

Added in release 4.2.0.0
Added support for the following new OPNs:
- BRD4318A
- BRD4335A

13.2 Improvements

None

13.3 Fixed Issues

None

13.4 Known Issues in the Current Release

None

13.5 Deprecated Items

None

13.6 Removed Items

None
14 Other Gecko Platform Software Components

No changes
15 RAIL Library

15.1 New Items

**Added in release 4.2.0.0**

- Added HFXO temperature compensation in RAIL on platforms that support RAIL_SUPPORTS_HFXO_COMPENSATION. This feature can be configured with the new RAIL_ConfigHFXOCompensation() API. The user will also need to be sure to handle the new RAIL_EVENT_THERMISTOR_DONE event to trigger a call to RAIL_CalibrateHFXO to perform the compensation.
- Added options in "RAIL Utility, Protocol" component to control whether Z-Wave, 802.15.4 2.4 GHz and Sub-GHz, and Bluetooth LE are enabled so that the user can save space in their application by disabling unused protocols.
- Added a new API RAIL_ZWAVE_PerformIrcal to help perform an IR calibration across all the different PHYs used by a Z-Wave device.
- Added 40 MHz crystal support on EFR32xG24 devices to the "RAIL Utility, Built-in PHYs Across HFXO Frequencies" component.
- Added support for IEEE 802.15.4 fast RX channel switching with the new RAIL_IEEE802154_ConfigRxChannelSwitching API on supported platforms (see RAIL_IEEE802154_SupportsRxChannelSwitching). This feature allows us to simultaneously detect packets on any two 2.4 GHz 802.15.4 channels with a slight reduction in overall sensitivity of the PHY.
- Added support for the BGM220SC23HNA2 module.
- Added a new Thermal Protection feature, on platforms that support RAIL_SUPPORTS_THERMAL_PROTECTION, to track temperature and prevent transmits when the chip is too hot.
- Added new table-based OFDM and FSK PAs for EFR32xG25 based devices. The output power of these can be modified through a new customer provided look-up table. Ask support or look for an updated app note on how to configure the values in this table for your board.
- Added support for the MGM240SA22VNA, BGM240SA22VNA, and BGM241SD22VNA modules and updated the configurations for the BGM240SB22VNA, MGM240SB22VNA, and the MGM240SD22VNA.

15.2 Improvements

**Changed in release 4.2.1.0**

- Corrected the sign of the frequency error reported by RAIL_GetRxFreqOffset() when using OFDM on the EFR32xG25 to match how this was handled for other modulations (e.g. Freq_error=current_freq-expected_freq).
- The RAIL_SetTune() and RAIL_GetTune() functions now use the CMU_HFXOCTuneSet() and CMU_HFXOCTuneGet() functions respectively on EFR32xG2x and newer devices.

**Changed in release 4.2.0.0**

- The RAIL_ConfigRfSenseSelectiveOokWakeupPhy() will now return an error when run on the EFR32xG21 platform because this device cannot support the wakeup PHY.
- Updated the pa_customer_curve_fits.py helper script to accept floating point value for the maximum power argument, similar to the increment argument.
- Added support in "RAIL Utility, Coexistence" component for configuring priority options when directional priority is enabled but no static priority GPIO is defined.
- Broke up some EFR32xG12 802.15.4 dynamic FEC code to save code size for Zigbee and Bluetooth LE, which never need this functionality.
- Remove "RAIL Utility, Coexistence" component dependency from the RAIL Utility, Coulomb Counter component.
- The RAIL_PrepareChannel() function has been made dynamic multiprotocol safe and will no longer return an error if called when your protocol is inactive.

15.3 Fixed Issues

**Fixed in release 4.2.1.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1077611</td>
<td>Fixed an issue on the EFR32xG25 that would cause a 40 µs porch before an OFDM TX.</td>
</tr>
</tbody>
</table>
### Fixed in release 4.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>843708</td>
<td>Moved function declarations from rail_features.h to rail.h to avoid a convoluted include dependency order.</td>
</tr>
<tr>
<td>844325</td>
<td>Fixed RAIL_SetTxFifo() to properly return 0 (error) rather than 4096 for an undersized FIFO.</td>
</tr>
<tr>
<td>845608</td>
<td>Fixed an issue with the RAIL_ConfigSyncWords API when using certain underlying demodulator hardware on EFR32xG2x parts.</td>
</tr>
<tr>
<td>851150</td>
<td>Fixed an issue on EFR32xG2 series devices where the radio would trigger RAIL_ASSERTSEQUENCER_FAULT when PTI is used and GPIO configuration is locked. GPIO configuration can only be locked when PTI is disabled. See RAIL_EnablePti() for further information.</td>
</tr>
<tr>
<td>857267</td>
<td>Fixed an issue when using the &quot;RAIL Utility, Coexistence&quot; component with TX abort, the signal identifier feature and DMP.</td>
</tr>
<tr>
<td>1015152</td>
<td>Fixed an issue on EFR32xG2x devices where RAIL_EVENT_RX_FIFO_ALMOST_FULL or RAIL_EVENT_TX_FIFO_ALMOST_EMPTY could trigger improperly when the event is enabled or the FIFO is reset.</td>
</tr>
<tr>
<td>1017609</td>
<td>Fixed an issue where PTI appended information could be corrupted when RAIL_RX_OPTION_TRACK_ABORTED_FRAMES is in effect when RAIL_IDLE_FORCE_SHUTDOWN or RAIL_IDLE_FORCE_SHUTDOWN_CLEAR_FLAGS is used. Also clarified that RAIL_RX_OPTION_TRACK_ABORTED_FRAMES is not useful with coded PHYs.</td>
</tr>
<tr>
<td>1019590</td>
<td>Fixed an issue when using the &quot;RAIL Utility, Coexistence&quot; component with BLE where the sl_bt_system_get_counters() function would always return 0 for GRANT denied counts.</td>
</tr>
<tr>
<td>1019794</td>
<td>Eliminated compiler warning in &quot;RAIL Utility, Initialization&quot; component when few of its features are enabled.</td>
</tr>
<tr>
<td>1023016</td>
<td>Fixed an issue on EFR32xG22 and newer chips where waits in between radio activity would consume slightly more power than necessary after the first 13 ms. This was especially noticeable when using RAIL_ConfigRxDutyCycle with large off time values.</td>
</tr>
<tr>
<td>1029740</td>
<td>Fixed issue where RAIL_GetRssi()/RAIL_GetRssiAlt() could return a &quot;stale&quot; RSSI value (the value was from previous RX state instead of the current one) if called quickly upon entering receive.</td>
</tr>
<tr>
<td>1040814</td>
<td>Added support to the &quot;RAIL Utility, Coexistence&quot; component for configuring the coexistence request priority on sync detect when using BLE.</td>
</tr>
<tr>
<td>1062712</td>
<td>Fixed an issue where the &quot;RAIL Utility, Coexistence&quot; component would not always update request states correctly, which could lead to missed events triggered by new requests.</td>
</tr>
<tr>
<td>1062940</td>
<td>Prevented the &quot;RAIL Utility, Coexistence&quot; component from aborting BLE transmits when SL_RAIL_UTIL_COEX_BLE_TX_ABORT is disabled.</td>
</tr>
<tr>
<td>1063152</td>
<td>Fixed an issue where radio reception would not be fully cleaned up when a receive error occurred with receive state transitions set to idle on error but transmit on success, a configuration mostly associated with BLE. On the EFR32xG24 this could cause a SYNTH calibration to not be properly restored and eventually cause the radio to stop working.</td>
</tr>
</tbody>
</table>

### 15.4 Known Issues in the Current Release

Issues in bold were added since the previous release.

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
<th>Workaround</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>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</td>
<td></td>
</tr>
<tr>
<td>ID #</td>
<td>Description</td>
<td>Workaround</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>641705</td>
<td>Infinite receive operations where the frame's fixed length is set to 0 are not working correctly on the EFR32xG23 series chips.</td>
<td></td>
</tr>
</tbody>
</table>
| 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 |                                                                              |

### 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!