Gecko Platform 3.0.2.0 GA
Gecko SDK Suite 3.0
October 14, 2020

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

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

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

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

Services includes common services such as NVM3 and Power Manager.

Common components are used throughout the SDKs.

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

Security includes mbed TLS and other security services.

Operating System includes Micrium OS Kernel.

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.

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 SDK 3.0.2.0 released October 14, 2020
Gecko SDK 3.0.1.0 released September 30, 2020
Gecko SDK 3.0.0.2 released July 29, 2020

KEY FEATURES

Drivers
- Improved worst case repack execution time for NVM3

Services
- New Power Manager service replacing SLEEP driver
- Deprecated RTCDRV in favor of Sleep Timer service

Security
- Mbed TLS updated to v2.16.6
- Added hardware acceleration for AES-GCM

Gecko Bootloader
- Added rollback prevention feature
- Upgrade process speedup

RAIL Library
- Studio V5/GSDK 3.0 support for RAIL
- New channel hopping (Preamble Sense) and low duty cycle (LDC) modes to help performance in noisy environments
- Optional support for the Silicon Labs Power Manager
- New and updated Sample Applications

Contents

1 CMSIS Device

1.1 New Items

Added in release 3.0.0.2

- Added support for new Zen Gecko families EFR32ZG13L and EFR32ZG13S

1.2 Improvements

None

1.3 Fixed Issues

Fixed in release 3.0.2.0

ID #	Description
500351	Added missing GPIO_IF_EM4WU_MASK define in efr32mg22_gpio.h.

1.4 Known Issues in the Current Release

None

1.5 Deprecated Items

None

1.6 Removed Items

Removed in release 3.0.0.2

- Removed EFR32BG14 Blue Gecko family

2   Peripherals

2.1   New Items

Added in release 3.0.0.2

- Added support for new IADC result alignment options and IADC digital averaging on device families with hardware support for this.
- Added an API SYSTEM_GetSecurityCapability(), which can be used to find security capability of different products.
- Added an API CMU_LF_ClockPrecisionGet(), to query the clock precision.

2.2   Improvements

None

2.3   Fixed Issues

Fixed in release 3.0.1.0

ID #	Description
519432	Fixed issue where calling TIMER_Reset for a disabled timer caused a bus fault for series 2 devices.
500814	Fixed an issue in emlib's SMU module where the register mapping was not correctly done.

Fixed in release 3.0.0.2

ID #	Description
449691	Fixed issue in em_i2c where the state machine sometimes issued a NACK after last byte was read in a slave-to-master transfer. This could cause issues when auto ack was enabled. After this fix the NACK is issued before the last byte is read in all state machine transitions.
447712	On series-2 devices the VSCALEBUSY bit is not set immediately when starting EM01 voltage scaling. With this fix the VSCALEDONE interrupt flag is checked instead of the VSCALEBUSY status bit when waiting for EM01 voltage scaling to complete.
486299	Fixed a bug in em_iadc.c when writing to the TRIGGER register.
487657	Added checks and recovery mechanisms when attempting to set the frequency of HFRCODPLL to an unsupported value.

2.4   Known Issues in the Current Release

None

2.5   Deprecated Items

Deprecated in release 3.0.0.2

- All SE functionality and command defines are marked as deprecated in em_se. Use SE Manager APIs instead.

2.6   Removed Items

Removed in release 3.0.0.2

- Removed em_mpu emlib module. Please use the ARM_MPU_xxx API instead.
- Removed em_core emlib module. Please use the new em_core module instead.

3 Drivers

3.1 New Items

Added in release 3.0.1.0

- Added MPU integration to NVM3.

Added in release 3.0.0.2

- Added a new way to build project using Simplicity Studio v5 Project Configurator. If your project is using this new method (this method is used by Bluetooth and Proprietary SDKs only), it is important to understand the SDK programming model to determine if your application-specific initialization functions, or periodic processing functions need to be integrated into the new `main.c`. For more information see the sdk-programming-model section.
- Added PWM driver in platform/driver. Documentation can be found at driver/api/group-pwm.
- Added simple button driver in platform/driver. Documentation can be found at driver/api/group-button.
- Added simple and RGBW LED drivers in platform/driver. Documentation can be found at driver/api/group-led.
- Added I2C Single Polled Master (I2CSPM) driver in platform/driver. Documentation can be found at driver/api/group-i2cspm.
- Added DMA Control block to DMADRV.

3.2 Improvements

Changed in release 3.0.2.0

- The nvm3_repack function has been updated to use the cache to reduce the computational overhead to find objects that must be copied before a page erase. When the cache is valid (large enough to cache all the objects in the store), this improvement will reduce the overhead in the data copy part of nvm3_repack and reduce the maximum time for the function call.

Changed in release 3.0.1.0

- During execution, the nvm3_repack function will either do nothing, copy data or erase a page. To limit the execution time, the copy part is now split into several calls where each call will never copy more than max-object-size number of bytes. This is done to limit the time when interrupts are disabled if the default locking functions are used.

3.3 Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on https://www.si-labs.com/products/software.

ID #	Description
485403	UARTDRV operation in EM2 does not work when using EUART.
520149	When creating SPIDRV instances using the Simplicity Studio Project Configurator, code to initialize the instance is not automatically generated and must instead be added manually.

3.4 Fixed Issues

Fixed in release 3.0.1.0

ID #	Description
520149	Fixed an issue where initialization of SPIDRV instances was not generated when adding a SPIDRV instance to a project.
501179	Fixed an issue where the configuration GUI for the SPIDRV driver required a CS pin to be configured when the driver was configured to leave CS control to the application. This issue only affected the configuration GUI, and had no effect on the embedded source code.

Fixed in release 3.0.0.2

ID #	Description
497230	Fixed an issue in UARTDRV where handles where removed in UARTDRV_Abort. Handles are now instead removed in UARTDRV_DeInit.

3.5 Deprecated Items

Deprecated in release 3.0.0.2

- The SLEEP driver is deprecated and will be deleted in a later release. The new power manager service should be used instead. Note that all Silicon Labs wireless protocols included with this release as well as RAIL will also use the power manager instead of SLEEP driver. If your applications uses any of the wireless protocols, it can no longer use or include the SLEEP driver or call directly the emlib EMU functions as this is incompatible with the power manager. In that case, the power manager will need to be used. Direct EMU calls need to be called to set the device to EM4, as power manager only covers EM0 to EM3. More details can be found at service/power_manager/overview. If your application did not use SLEEP driver in the past, no action is needed on your part.

- The RTCDRV driver is deprecated and will be deleted in a later release. The sleep timer service should be used instead. Note that all Silicon Labs wireless protocols included with this release will also use the sleep timer service. In such applications the RTCDRV driver and sleep timer can only coexist if they use separate real-time clock peripherals.

- The following drivers in hardware/kit/common/drivers are marked as deprecated and will be removed in a later release:
  - eeprom
  - i2ccaptouch
  - ksz8851snl
  - ksz8851snl_spi
  - retargetfft
  - tempsens
  - tftamapped
  - tftdirect
  - tftspi
  - touch

3.6 Removed Items

Removed in release 3.0.0.2

- Deleted deprecated config files in platform/emdrv/config folder.
- Deleted deprecated NVM driver. NVM3 service should be used instead.

4   Services

4.1   New Items

Added in release 3.0.1.0

- Added deinit functionality to IOStream-UART interfaces.

Added in release 3.0.0.2

- Added a new way to build project using Simplicity Studio v5 Project Configurator. If your project is using this new method (this method is used by Bluetooth and Proprietary SDKs only), it is important to understand the SDK programming model to determine if your application-specific initialization functions, or periodic processing functions need to be integrated into the new `main.c`. For more information see the SDK Programming Model section at sdk-programming-model.

- Added Command Line Interface (CLI) service in platform/service. Documentation can be found at service/api/group-cli.

- Added Power Manager service in platform/service. Documentation can be found at service/power_manager/overview and service/api/group-power-manager. Note that all Silicon Labs wireless protocols included with this release as well as RAIL will also use the power manager instead of SLEEP driver. If your applications uses any of the wireless protocols, it can no longer use or include the SLEEP driver or call directly the emlib EMU functions as this is incompatible with the power manager. In that case, the power manager will need to be used. Direct EMU calls need to be called to set the device to EM4, as power manager only covers EM0 to EM3. More details can be found at service/power_manager/overview. If your application did not use SLEEP driver in the past, no action is needed on your part.

- Added IO Stream service in platform/service. Documentation can be found at service/api/group-iostream.

- Added Microsecond Delay service (udelay). Documentation can be found at service/api/group-udelay.

- Added device initialization service. Documentation can be found at service/api/group-device-init.

4.2   Improvements

Changed in release 3.0.1.0

- Added documentation about EM4 support in the Power Manager.

- Modified Simple MPU driver to protect entire RAM section defined in ARM memory map (0x2000 0000 - 0x3FFF FFFF) against execution.

Changed in release 3.0.0.2

- Added the option to use BURTC as the hardware timer peripheral in the Sleep Timer. More details can be found here service/api/group-sleeptimer#sleeptimer-conf.

- Added the option in CLI service to ignore case in CLI commands.

4.3   Fixed Issues

Fixed in release 3.0.2.0

ID #	Description
620789	Fixed MISRA violation in power manager by removing inclusion of stdio.h.
625372	Updated power manager handling of HFXO with Series 2. The on-demand mode will no longer be disabled when entering EM2 or EM3.

Fixed in release 3.0.1.0

ID #	Description
501561	In the Legacy HAL component the PA configuration now uses the setting from the RAIL Utility PA configuration component instead of its own hard-coded settings.

Fixed in release 3.0.0.2

ID #	Description
491275	Fixed bug where the function sl_mpu_disable_execute() could disable execution on a larger memory section than requested on ARMv6M- and ARMv7M-based devices. This could lead to unwanted memory regions marked as non-executable.
471079	Fixed a bug in Sleep Timer where, under some circumstances, a timer could expire too early when using RTC or PRORTC as hardware timers.

4.4 Known Issues in the Current Release

None

4.5 Deprecated Items

None

4.6 Removed Items

Removed in release 3.0.0.2

- Removed support for NVM3 in external flash.

5   Common

5.1   New Items

Added in release 3.0.0.2

- Added string support for the Status Codes (`sl_status_t`). Documentation can be found at common/api/group-status.

5.2   Improvements

None

5.3   Fixed Issues

None

5.4   Known Issues in the Current Release

None

5.5   Deprecated Items

None

5.6   Removed Items

None

6 Middleware

6.1 New Items

Added in release 3.0.0.2

- Added GLIB function to draw a string on a specific line on a display using a specific alignment.

6.2 Improvements

None

6.3 Fixed Issues

Fixed in release 3.0.0.2

ID #	Description
466261	Fixed an issue in Micrium OS FSs' FS_FAT_DirRd() where OEF was not reported correctly.
484320	Added a missing include to em_chip.h in bsp_fs_nor_spi.c.

6.4 Known Issues in the Current Release

None

6.5 Deprecated Items

None

6.6 Removed Items

None

7   Security

7.1   New Items

Added in release 3.0.0.2

- Added mbed TLS support for hardware acceleration of GCM on Series-1 devices including the CRYPTO peripheral.
- Initial release of SE Manager.

7.2   Improvements

Changed in release 3.0.0.2

- Mbed TLS library version upgraded from 2.7.12 to 2.16.6.

7.3   Fixed Issues

None

7.4   Known Issues in the Current Release

None

7.5   Deprecated Items

None

7.6   Removed Items

Removed in release 3.0.0.2

- The mbed TLS config options MBEDTLS_MICRIUMOS and MBEDTLS_MICRIUM have been replaced with SL_CATALOG_MICRIUMOS_KERNEL_PRESENT.
- The mbed TLS config option MBEDTLS_FREERTOS has been replaced with SL_CATALOG_FREERTOS_KERNEL_PRESENT.

8 Operating System

8.1 New Items

None

8.2 Improvements

Changed in release 3.0.0.2

- Removed the need to set RTOS_INT_CONTROLLER_SEL in Micrium OS's rtos_description.h.

8.3 Fixed Issues

Fixed in release 3.0.2.0

ID #	Description
621988	This release includes a fix for OS_OPT_TIME_PERIODIC option in OSTimeDly() function. When the kernel has been running for few days, OSTimeDly() used to mistakenly return an overflow error if the delay period overlapped with the moment when the sleep timer reset its counter. The issue was fixed by using modular arithmetic and refining the definition of maximum possible value for delay.

Fixed in release 3.0.1.0

ID #	Description
617560	Minor modification to the round-robin algorithm in Micrium OS scheduler. Before this release, if only one task is running for a long time, the round-robin sleep timer would be disabled for optimization. This resulted in undefined behavior in some special cases. To mitigate this problem, Micrium OS now resets the sleep timer of the task whenever it times out.
519121	Fixed an issue where idle context was not reached when OS_CFG_PRIO_MAX was larger than twice the data bus width.
490998	Fixed a case where OSSched() could de-reference a null pointer, in specific conditions.
468431	Some OS functions now return and set an error pointer to RTOS_ERR_NOT_READY if the kernel is not running.

Fixed in release 3.0.0.2

ID #	Description
	Fixed a bug in Micrium OS's slists that could appear on some compilers with some settings.

8.4 Known Issues in the Current Release

None

8.5 Deprecated Items

Deprecated in release 3.0.1.0

OS_TRACE_INIT is deprecated. The way to initialize SystemView is by calling SEGGER_SYSVIEW_Conf().

8.6 Removed Items

Removed in release 3.0.0.2

The deprecated Clock (under Micrium OS) module was removed.

9 Gecko Bootloader

9.1 New Items

Added in release 3.0.0.2

- Application rollback protection support added.
- Added a new internal storage bootloader sample application bootloader-storage-internal-single-352 k for devices with 352 kB internal flash.
- bootloader_secureBootEnforced() is added to check if signature verification on the application is enforced before every boot.
- bootloader_parserContextSize is added, which can be used to find size of context struct BootloaderParserContext used by bootloader image parser to store parser state.
- The application upgrade process is updated to use DMA-based MSC writes. DMA-based MSC write can be enabled for storage interface APIs by selecting DMA from the internal storage plugin. bootloader_getAllocatedDMAChannel can be used check the allocated DMA channel for the bootloader.
- Added bootloader_parseImageInfo that can be used to parse ApplicationData_t and bootloader version from a GBL stored in a buffer.

9.2 Improvements

Changed in release 3.0.0.2

- The size of ParserContext_t for Series-2 devices has been increased to 556 bytes.

9.3 Fixed Issues

None

9.4 Known Issues in the Current Release

None

9.5 Deprecated Items

None

9.6 Removed Items

None

10 Examples

10.1 New Items

Added in release 3.0.1.0

- Added sample applications to demonstrate usage of the Secure Element Manager.

Added in release 3.0.0.2

- Added common platform applications under app/common/example.
  - Empty C project
  - Blink Bare-metal
  - Blink Kernel
  - Blink PWM
  - I/O Stream USART Bare-metal
  - I/O Stream USART Kernel

10.2 Improvements

None

10.3 Fixed Issues

Fixed in release 3.0.2.0

ID #	Description
525789	Fixed metadata for retarget_swo dependency resolution and fixed incompatibility with retarget_swo and sl_memory.
623321	Device Init HFXO component will now be added automatically when a device uses RAIL library and a warning will be displayed when using LFRCO in high precision mode without initializing HFXO.

10.4 Known Issues in the Current Release

None

10.5 Deprecated Items

None

10.6 Removed Items

None

11 Boards and External Devices

11.1 New Items

Added in release 3.0.1.0

- Added VEML6035 driver to hardware/driver.

Added in release 3.0.0.2

- Added the following external device drivers to hardware/driver: BMP280, CCS811, ICM20648, IMU, Memory LCD, MX25 flash shutdown, I2S mic, PDM mic, si1133, si70xx, si7210.
- Added functions for controlling Silicon Labs board features in hardware/board.

11.2 Improvements

None

11.3 Fixed Issues

Fixed in release 3.0.0.2

ID #	Description
491599	Fixed an issue where incorrect LFXO and HFXO configurations were applied in the default configuration for Silicon Labs boards.

11.4 Known Issues in the Current Release

None

11.5 Deprecated Items

None

11.6 Removed Items

None

12 Other Gecko Platform Software Components

12.1 New Items

Added in release 3.0.0.2

- Added automatic initialization of MX25 flash shutdown component.
- Added metadata description of platform code to support Simplicity Studio Project Configurator.
- Added CMSIS annotated configuration headers for platform code to support configuration of platform components in Simplicity Studio Project Configurator.
- Added board-specific CMSIS annotated configuration header files.
- Added support for generating linker files based on component selection in Simplicity Studio Project Configurator.

12.2 Improvements

Changed in release 3.0.0.2

- Various fixes to documentation on https://docs.silabs.com/.

12.3 Fixed Issues

None

12.4 Known Issues in the Current Release

None

12.5 Deprecated Items

Deprecated in release 3.0.0.2

- The Segger emWin graphics library is marked as deprecated and will be removed in a later release
- The IEC 60335 Class B library is marked as deprecated and will be removed in a later release

12.6 Removed Items

None

13 RAIL Library

13.1 New Items

Added in release 3.0.1.0

- Added support for MGM210PB22JIA, MGM210PB32JIA, BGM210PB22JIA and BGM210PA32JIA modules.
- Added RAIL build support for BGX220P22HNA21 and BGX220S22HNA21 modules.

Added in release 3.0.0.2

- Added support for setting the default value of the FramePending bit in outgoing IEEE 802.15.4 ACKs to true. This means that the user would then be responsible for clearing this bit in the frame pending callback instead of having it default to cleared and having to set it. See the RAIL_IEEE802154_Config_t::defaultFramePendingInOutgoingAcks field for configuring this feature.
- Added a new API RAILCb_ConfigSleepTimerSync() to allow for configuration of the PRS and RTCC channels used for timer sync operations.
- Added support for a new RAIL_EVENT_SCHEDULED_RX_STARTED and RAIL_EVENT_SCHEDULED_TX_STARTED, triggered when a scheduled receive or transmit begins. These are the same value because a scheduled receive and transmit cannot occur at the same time. Note: This new event shifted the bit positions of some events in RAIL_Events_t.
- Provided new RAIL_RX_CHANNEL_HOPPING_MODE_MULTI_SENSE along with RAIL_RxChannelHoppingConfigMultiMode_t to configure its parameters. This mode can be configured to tolerate brief loss of timing and/or preamble, making it less susceptible to hopping than the single-sense modes. It can also be used with RX duty cycling.
- Provided new RAIL_RX_CHANNEL_HOPPING_OPTION_RSSI_THRESHOLD to augment each of the RAIL_RxChannelHoppingMode_t modes with one RSSI Threshold check on entering receive for the channel. If the RSSI is below that specified by RAIL_RxChannelHoppingConfigEntry_t::rssiThresholdDbm then hop (or if below RAIL_RxDutyCycleConfig_t::rssiThresholdDbm, suspend Rx). RAIL_RxDutyCycleConfig_t has been augmented not only with this new field, but also now includes RAIL_RxDutyCycleConfig_t::options field. For those options to be recognized by RAIL_ConfigRxDutyCycle() (due to backwards compatibility) RAIL_RxDutyCycleConfig_t::mode must be one of the new WITH_OPTIONS modes, e.g. RAIL_RX_CHANNEL_HOPPING_MODE_TIMEOUT_WITH_OPTIONS.
- Added support for the Silicon Labs Power Manager component. When enabled via RAIL_InitPowerManager(), RAIL will communicate directly with the power manager to configure sleep modes.
- Updated the function RAIL_GetRxPacketDetailsAlt to return the time position of the received packet timestamp corresponding to the default location in the packet.
- Added a new function, RAIL_GetTxPacketDetailsAlt2, which allows a RAIL_TxPacketDetails_t structure to be passed as an argument.
- Added a new function, RAIL_GetTxTimePreambleStartAlt, which allows a RAIL_TxPacketDetails_t structure to be passed as an argument.
- Added a new function, RAIL_GetTxTimeSyncWordEndAlt, which allows a RAIL_TxPacketDetails_t structure to be passed as an argument.
- Added a new function, RAIL_GetTxTimeFrameEndAlt, which allows a RAIL_TxPacketDetails_t structure to be passed as an argument.

13.2 Improvements

Changed in release 3.0.2.0

- Reduced the time from last modulated bit on air to ramp down on EFR32xG1x devices.

Changed in release 3.0.0.2

- Relax constraints in RAIL to allow calling RAIL_SetRxTransitions, RAIL_SetTxTransitions, RAIL_ScheduleRx, and all of the RAIL_BLE_ConfigPhy before the radio is completely IDLE.
- Updated the pa_customer_curve_fits.py helper script to work with Python 3 as well as Python 2.
- Changed pa_customer_curve_fits.py to take maxpower as a parameter to generate better curves. When maxpower and increment are different than the defaults they will now be included in the output curve. The current power curve limits can be read at runtime from the new RAIL_GetTxPowerCurveLimits API.
- The RAIL_GetRadioEntropy() API will now ensure a valid radio configuration has been loaded using RAIL_ConfigChannels() since it can cause problems if the radio is used before this.

- Changed the minimum ramp time of the RAIL_TX_POWER_MODE_2P4GIG_MP to be 3us to avoid problems with shorter ramp times.
- Improved frequency accuracy on EFR32xG1x devices when the radio configuration has an entry with a large number of channels. Previously, small errors in the channel spacing calculation could exist. If they did, then when computing the channel frequency this error would be multiplied by the channel number minus the start channel for that entry causing some drift for higher order channels. This was not much an issue in most cases, but for certain PHY and crystal combinations it could be worse.

13.3 Fixed Issues

Fixed in release 3.0.2.0

ID #	Description
623922	Fix to set the correct power on EFR32xG21 when the PA uses VDD around 1.8V.

Fixed in release 3.0.1.0

ID #	Description
493409	Fixed an issue where output power might not be set correctly when changing frequency bands within one channel configuration.
497061	Corrected an issue on EFR32xG1x chips where specifying a large ramp time could erroneously add extra delay to a transmit.
519195	EFR32xG21 will now use RTCC channel 0, as opposed to the PRORTC, to perform sleep timer synchronization. This will help lower the EM2 current consumption for this chip.

Fixed in release 3.0.0.2

ID #	Description
456338	Fixed an issue with RAIL state transitions where an internal timer wrapping could cause incorrect transition times. This error would previously affect a maximum of one packet every 15 minutes.
459581	Fixed an issue where the output power was too low on the EFR32xG21 when using the RAIL_TX_POWER_MODE_2P4GIG_MP PA and certain ramp times.
464534	Fixed issue where RAIL_StartAverageRssi() ran twice as long as it should have.
464734	Regenerated the power curves for the EFR32xG22 to allow access to the maximum power level available on the chip.
464735	Closed tiny timing window on EFR32xG13 that might corrupt PTI appended info when idling the radio.
465096	Fixed an issue where RAIL_Idle() was not properly terminating an ongoing RAIL_StartAverageRssi() process.
466012	Fixed an issue where the CRC could be disabled indefinitely on transmit when switching configs in a multiphy setup.
474678	Fixed an issue with duty cycle receive and channel hopping on the EFR32xG1x parts where some components would be left on even with long delay parameters causing extra current to be used. This allows for a noticeable improvement in power consumption when using the RAIL_ConfigRxDutyCycle() API with a delay in the hundreds of microseconds or more.
475184	Fixed an issue on the EFR32xG22 where the receiver was not automatically re-calibrated if the temperature changed significantly while sitting in receive. This could cause the radio to go off channel for significant temperature changes resulting in receive problems.
477833	Some radio configurations on the EFR32XG22 are not usable with RAIL address filtering and RAIL 802.15.4 filtering. Add an assert to catch those cases.
479539	Fixed a bug where the RAIL_ConfigTxPower() would override the PA capacitor tune values for transmit and receive without caching them. In a multiprotocol scenario this could cause us to apply incorrect PA capacitor tune values set prior to a call to RAIL_ConfigTxPower. Note that you must still call RAIL_SetPaCTune after any making any changes to the power configuration via RAIL_ConfigTxPower.
479665	Fixed an issue where RAIL_SetRxFifo() would not reject a buffer smaller than 64 bytes and would mistakenly think it is very big. In addition, RAIL_ASSERT_FAILED_RX_FIFO_BYTES can no longer occur with a 64 byte buffer when doing IR calibration.

ID #	Description
482007	Fixed a bug in multiprotocol RAIL where running an IR calibration during a protocol switch would fail. The calibration function will now return RAIL_STATUS_INVALID_STATE if called in such a scenario.
483688	Fixed an issue where the power mode selected when using RAIL_TX_POWER_MODE_2P4GIG_HIGHEST on supported chips was not saved in a multiprotocol context and could cause problems.
484374	Fixed regression from 2.8.1 on EFR32xG21 where Bluetooth LE did not include packet sync word on PTI.
489214	Fixed an issue where calling RAIL_IEEE802154_CalibrateIr2p4Ghz, RAIL_IEEE802154_CalibrateIrSubGhz, or RAIL_BLE_CalibrateIr with a NULL imageRejection parameter would result in a crash.

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.si-labs.com/products/software.

ID #	Description
500103	When using WiFi coexistence with no priority pin but a directional priority pin the high priority events will be counted with low priority events incorrectly.

13.5 Deprecated Items

None

13.6 Removed Items

None

14 Legal

14.1 Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications.

Application examples described herein are for illustrative purposes only.

Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

14.2 Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmo-dem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri, Z-Wave and others are trademarks or registered trademarks of Silicon Labs.

ARM, CORTEX, Cortex-M0+, Cortex-M3, Cortex-M33, Cortex-M4, TrustZone, Keil and Thumb are trademarks or registered trademarks of ARM Holdings.

Zigbee® and the Zigbee logo® are registered trademarks of the Zigbee Alliance.

Bluetooth® and the Bluetooth logo® are registered trademarks of Bluetooth SIG Inc.

All other products or brand names mentioned herein are trademarks of their respective holders.