



Bluetooth® SDK 2.13.1.0 GA

Gecko SDK Suite 2.7

January 24, 2020

Silicon Labs is a leading vendor in Bluetooth hardware and software technologies, used in products such as sports and fitness, consumer electronics, beacons, and smart home applications. The core SDK is an advanced Bluetooth 5-compliant stack that provides all of the core functionality along with multiple API to simplify development. The core functionality offers both standalone mode allowing a developer to create and run their application directly on the SoC, or in NCP mode allowing for the use of an external host MCU.

Extensions to the SDK include Bluetooth Mesh and Apple® HomeKit® for customers seeking the additional capabilities.

These release notes cover SDK version(s):

2.13.1.0 released on January 24, 2020

2.13.0.0 released on December 13, 2019



KEY FEATURES

- Adds directional priority PTA support for Wi-Fi Coex
- Updates to the Bluetooth Mobile app
- NVM3 support on EFR32[B|M]G1x devices

Compatibility and Use Notices

If you are new to the Silicon Labs Bluetooth SDK, see [Using This Release](#).

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 8.30.1

- Using wine to build with the `IarBuild.exe` command line utility or IAR Embedded Workbench GUI on macOS or Linux could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.
- Customers on macOS or Linux are advised not to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

GCC (The GNU Compiler Collection) version 7.2.1, provided with Simplicity Studio.

Contents

- 1 New Items 3
 - 1.1 New Features 3
 - 1.2 New APIs 3
- 2 Improvements 4
 - 2.1 Changed APIs 4
- 3 Fixed Issues 5
- 4 Known Issues in the Current Release 6
- 5 Deprecated Items 7
- 6 Removed Items 8
- 7 Using This Release 9
 - 7.1 Installation and Use 9
 - 7.2 Support 9
- 8 Legal 10
 - 8.1 Disclaimer 10
 - 8.2 Trademark Information 10

1 New Items

1.1 New Features

Added in release 2.13.0.0

NVM3 support on EFR32xG1x devices

NVM3 is now supported on EFR32xG1x devices in addition to PS Store. These two data stores cannot be used simultaneously. The sample applications in the Bluetooth SDK continue to use PS Store by default. To replace PS Store with NVM3 please refer to [AN1135: Using Third Generation NonVolatile Memory \(NVM3\) Data Storage](#).

Usage of NVM3 on EFR32xG1x devices is recommended only for new designs as there is no migration mechanism from PS to NVM3, when a new firmware is flashed via OTA or UART DFU. Doing DFU from PS to NVM3 will cause all PS data to be lost, which includes e.g., bonding information.

Sleep Timer

The sleep timer (sl_sleeptimer.h) is a platform component providing single-shot and periodic timer services. It also has APIs for reading current tick count and conversions between ticks and milliseconds.

A Bluetooth application must initialize sleep timer because it is also needed by the Bluetooth stack for deep sleep. In addition, application must configure the timer frequency divider to 1 (SL_SLEEPTIMER_FREQ_DIVIDER in sl_sleeptimer_config.h). For more details of the initialization and configuration, please refer to a Bluetooth SDK example, e.g., the soc-empty.

1.2 New APIs

For additional documentation and command descriptions please refer to the [Bluetooth Software API Reference Manual](#).

Added in release 2.13.1.0

cmd_coex_set_directional_priority_pulse

Added in release 2.13.0.0

cmd_gatt_server_get_mtu

cmd_le_connection_read_channel_map

cmd_coex_set_parameters

2 Improvements

2.1 Changed APIs

Changed in release 2.13.0.0

cmd_system_linklayer_configure

New configuration key `system_linklayer_config_key_set_priority_table` is added for changing task priorities at run time.

cmd_sm_store_bonding_configuration

The default maximum allowed bonding count is changed to 13.

3 Fixed Issues

Fixed in release 2.13.1.0

ID #	Description
448020	In <code>system_set_tx_power</code> command, the stack now sets and returns the nearest possible value if the requested value is out of range.
450903	Fix the issue when application compiles mbedTLS into source that uses AES with padding, mbedTLS context objects in Bluetooth stack may be corrupted and cause hard faults.
450922	Use of GPCRC in Bluetooth stack is now thread safe.
452831	Fix an issue in Apploader that causes application OTA update to fail if the data length in the first data packet is not multiple of four.
455666	Fix a compatibility issue in Apploader with bootloader v1.10 on EFR32xG21. This issue causes OTA update to fail and the device to stay in OTA mode indefinitely.

Fixed in release 2.13.0.0

ID #	Description
234520	In <code>cmd_gatt_read_characteristic_value_by_uuid</code> command, if multiple characteristic values are received, an <code>evt_gatt_characteristic_value</code> event is generated for each value. Previously only one event is generated for this case.
281984	First advertisement packet is now sent with the right timing in dynamic multi-protocol use cases.
335894	Documentation fix to the <code>lolen</code> field in a BGAPI command header binary. The field specifies the minimal number of bytes the parameters take. It does not include the array data length if the command has an array type parameter.
347844	Event <code>evt_user_message_to_host</code> can now be sent during user command handling in NCP target application.
358171	Fix a random advertising hang issue which was observed when the application optimization build option was disabled or the device was exposed to low temperature.
408097	Extended advertising now works in dynamic multi-protocol use cases.
415583	The stack now returns out of memory error if the application sends a characteristic notification to all connections in low memory situation. Previously wrong state error was returned.
420866	Improves robustness when streaming large data packets to and from a device which does not support data length extension.
421731	When sending a characteristic value notification to all connections using <code>cmd_gatt_server_send_characteristic_notification</code> command, the stack now returns success if devices are connected but none has enabled the notification. Previously wrong state error was returned.
425187	<code>SL_BT_</code> prefix is added into the include guard in <code>bg_version.h</code> .
425532	Fix whitelisting when an extended advertising packet does not contain an auxiliary pointer.
430225	Fix CTUNE control using PS key on EFR32xG2x devices.
430752	Fix an issue that the stack may lose a periodic advertising synchronization because the scanner uses highest priority for secondary tasks. This was solved by lowering the priority.
434406	Remove the use of variable length array in the stack.
436243	Fix a HW initialization issue in the <code>soc-dtm</code> example that causes incorrect timeout for incoming data.
441445	Fix an advertisement data update race condition which causes no advertisement is sent out.
445627	Repeated bonding attempts handling is now connection specific. After first failed attempt there is cool down period of 1 s. After second attempt this is doubled to 2 s then to 4 s etc. The maximum wait time is 64 s.
450515	Fix an issue that causes BGTool does not run on Linux.

4 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
243009	With certain events, GCC breakpoints cannot be set.	Change optimization level to none in project settings
337467	MGM12P has poor signal strength when doing OTA with Aploader.	None
360313	Default RAIL assertion (RAILCb_AssertFailed) enters an infinite loop causing the stack not responsive.	Override RAIL assertion function in application space following these instructions .
361592	The sync_data event does not report TX power.	None
368798	Example project soc-thunderboard_sense_2 does not print float values correctly.	In project C/C++ settings, enable the printf float option.
396308	In NCP, BGAPI may be out of sync between the host and target when the target is reset in the middle of sending a BGAPI message to host. The out of sync issue could cause the host unable to communicate with the target.	In NCP target, add a delay to flush NCP Tx queue after a system reset command is received.
431452	When the requested TX Power is above 0 dBm, the actual transmit power may deviate randomly.	None
444469	When the data buffer for Bluetooth connections is full, cmd_le_connection_close command returns success but the connection is not really disconnected.	Data buffer full typically could happen in cases of intensive data streaming over connections using cmd_gatt_write_characteristic_value_without_response or cmd_gatt_server_send_characteristic_notification command. In this case, stop data streaming, and disconnect the connection after a small delay.
446074	The stack may be unable to establish new connections after multiple connection drops due to weak signal.	None
456701	Before gecko_init() is called, enabling busfaults generation on access to peripherals with clocks disabled causes a crash.	Enable busfaults generation after gecko_init().
457174	Bluetooth applications cannot use RTCC on EFR32xG13 and EFR32xG21.	Use Sleep Timer if its functionalities satisfy the intended use cases. Otherwise, none.

5 Deprecated Items

None

6 Removed Items

None

7 Using This Release

This release contains the following

- Silicon Labs Bluetooth stack library
- Bluetooth sample applications

This SDK depends on Gecko Platform. The Gecko Platform code provides functionality that supports protocol plugins and APIs in the form of drivers and other lower layer features that interact directly with Silicon Labs chips and modules. Gecko Platform components include EMLIB, EMDRV, RAIL Library, NVM3, and mbedTLS. Gecko Platform release notes are available through Simplicity Studio's Launcher Perspective, under this SDK's **Release Notes** doc header.

For more information about the Bluetooth SDK see [QSG139: Getting Started with Bluetooth® Software Development](#). If you are new to Bluetooth see [UG103.14: Bluetooth LE Fundamentals](#).

7.1 Installation and Use

A registered account at Silicon Labs is required in order to download the Silicon Labs Bluetooth SDK. You can register at https://siliconlabs.force.com/apex/SL_CommunitiesSelfReg?form=short.

Stack installation instruction are covered in [QSG139: Getting Started with Bluetooth® Software Development](#).

Use the Bluetooth SDK with the Silicon Labs Simplicity Studio V4 development platform. Simplicity Studio ensures that most software and tool compatibilities are managed correctly. Install software and board firmware updates promptly when you are notified.

Documentation specific to the SDK version is installed with the SDK. Additional information can often be found in the [knowledge base articles \(KBAs\)](#). API references and other information about this and earlier releases is available on <https://docs.silabs.com/>.

7.2 Support

Development Kit customers are eligible for training and technical support. You can use the [Silicon Labs Bluetooth LE web page](#) to obtain information about all Silicon Labs Bluetooth products and services, and to sign up for product support.

You can contact Silicon Laboratories support at <http://www.silabs.com/support>.

8 Legal

8.1 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.

8.2 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®, ISOModem®, 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.

Apple and HomeKit are registered trademarks of Apple Inc.

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