Manuals+ — User Manuals Simplified.

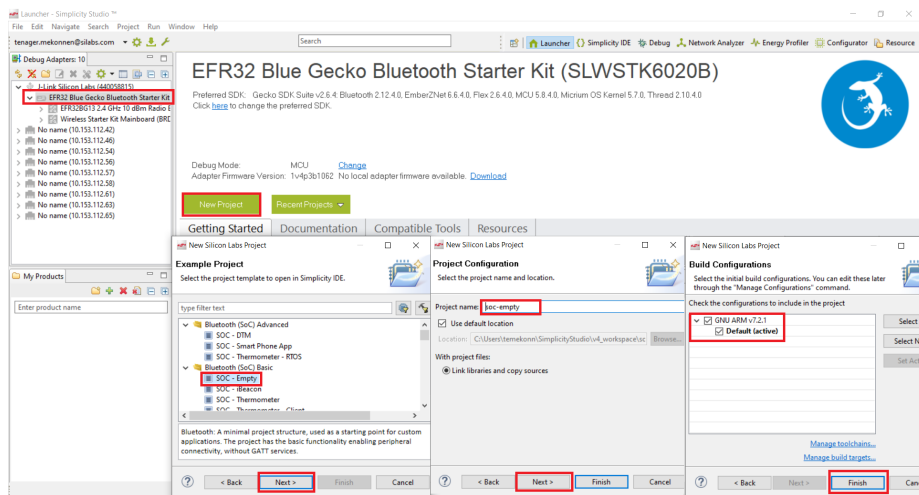# SILICON LABS Bluetooth LE SDK Software User Guide

Contents

SILICON LABS Bluetooth LE SDK Software

Release Date: September 5, 2023

Product Information

The Gecko SDK Suite 3.2 is a software development kit (SDK) provided by Silicon Labs. It is designed to facilitate the development of Bluetooth applications and offers various features and tools to enhance the development process.

Key Features:

- Compatibility and Use Notices
- Compatible Compilers

Compatibility and Use Notices:

The SDK provides compatibility and use notices to ensure proper usage and security of the software. For security updates and notices, refer to the Security chapter of the Gecko Platform Release notes installed with this SDK or visit the Silicon Labs Release Notes page. It is recommended to subscribe to Security Advisories for up-to-date information. For instructions on using Secure Vault features or if you are new to the Silicon Labs Bluetooth SDK, refer to the "Using This Release" section.

Compatible Compilers:

The G ecko SDK Suite 3.2 is compatible with the following compiler:

- GCC (The GNU Compiler Collection) version 10.2.0, provided with Simplicity Studio.

Improvements:

The following improvements have been made in version 3.2.9.0:

- Changed APIs

New Items:

The following new features have been added in previous releases:

Release 3.2.4.0:

- Python-based Host Examples: Python-based host examples for use
  with pyBGAPI are now available. You can find them at **https://github.com/SiliconLabs/pybgapi-examples**.

Release 3.2.0.0:

- Bluetooth Host Controller Interface: The Bluetooth Host Controller Interface is now supported. Refer to
  AN1328: Enabling a Radio Co-Processor using the Bluetooth HCI Function for more information.
- Dynamic GATT Database: The GATT database in the GATT server can now be created and managed
  dynamically with Bluetooth APIs. To use this feature, include the component
  "bluetooth_feature_dynamic_gattdb". See the component and configuration documentation and the Bluetooth
  API reference for further details.
- Simultaneous Scanning: The Bluetooth stack now supports simultaneous scanning on LE 1M and Coded PHY.
  This feature requires hardware support and is available only on certain devices.
- Throughput Logging: NCP host applications now support logging the throughput of the application. Use the "-l"
  option to enable this feature. The throughput value is saved in CSV format, and a logging entry is written once
  a minute.
- pyBGAPI: The pyBGAPI library, which implements the BGAPI protocol in Python, is now released in pypi.org.
  You can find it at **https://pypi.org/project/pybgapi/**.
- New Tools for Angle-of-Arrival (AoA) Development: The SDK includes an AoA Analyzer, a new 3D graphical
  tool integrated into Studio for evaluating AoA calculation with one locator and multiple tags. This tool replaces
  the previous AoA Compass Demo application.

KEY FEATURES

- Bluetooth HCI support
- Simultaneous scan on 1M and Coded-PHY
- Dynamic GATT configuration
- Release of pyBGAPI in pypi.org
- New tools for Angle-of-Arrival development

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 5.2-compliant stack that provides all of the core functionality along with multiple API to simplify development. The core func-tionality offers both standalone mode allowing a developer to create and run their appli-cation directly on the SoC, or in NCP mode allowing for the use of an external host MCU.

These release notes cover SDK version(s):

- 3.2.9.0 released September 5, 2023 (underlying platform changes only)
- 3.2.8.0 released July 13, 2023 (support for EFR32xG21, Revision C and later)
- 3.2.6.0 released March 29, 2023 (early access part support)
- 3.2.5.0 released January 11, 2023 (early access part support)
- 3.2.4.0 released October 13, 2021

- 3.2.3.0 released September 24, 2021
- 3.2.2.0 released September 8, 2021
- 3.2.1.0 released July 21, 2021
- 3.2.0.0 released June 16, 2021

Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Gecko Platform Release notes installed with this SDK or on the Silicon Labs Release Notes page. Silicon Labs also strongly recommends that you subscribe to Security Advisories for up-to-date information. For instructions as well as notes on using Secure Vault features, or if you are new to the Silicon Labs Bluetooth SDK, see Using This Release.

Compatible Compilers:
IAR Embedded Workbench for ARM (IAR-EWARM) version 8.50.9

- 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 10.2.0, provided with Simplicity Studio.

New Items

New Features

Added in release 3.2.4.0

Python-based Host Examples
Python-based host examples for use with pyBGAPI are now available (https://github.com/SiliconLabs/pybgapi-examples).

Added in release 3.2.0.0

Bluetooth Host Controller Interface
Beginning with this release, the Bluetooth Host Controller Interface is supported. See AN1328: Enabling a Radio Co-Processor using the Bluetooth HCI Function.

Dynamic GATT Database
In the GATT server, the GATT database can be created and managed dynamically with Bluetooth APIs. To use this feature, include the
component bluetooth_feature_dynamic_gattdb. See the component and configuration documentation, and the Bluetooth API reference.

Simultaneous Scanning
The Bluetooth stack supports simultaneous scanning on LE 1M and Coded PHY. This feature requires hardware support and is available only on certain devices.

New Example Applications

- Bluetooth – NCP (with Dynamic GATT support): Recommended instead of Bluetooth – NCP Empty, which has been deprecated.
- Bluetooth – RCP
- Bluetooth – SoC Blinky
- Bluetooth – SoC Light Standard DMP and Bluetooth – SoC Empty Standard DMP for EFRG32[B|M]G21 \
- Bluetooth – SoC Throughput
- Bluetooth – SoC Interoperability Test: Demo binary only, no source

Throughput Logging

NCP host applications support logging the throughput of the application. Use the -l option to enable the feature. The throughput value is saved in a CSV format. A logging entry is written once a minute.

pyBGAPI
The pyBGAPI library, implementing the BGAPI protocol in Python, is released in pypi.org (https://pypi.org/project/pybgapi/).

New Tools for Angle-of-Arrival (AoA) Development
AoA Analyzer: A new 3D graphical tool integrated into Studio to quickly evaluate AoA calculation with one locator and multiple tags. This tool replaces the previous AoA Compass Demo application.
AoA Configurator: A new 3D graphical tool to help customers create a valid multi-locator configuration file for multi-locator use cases.

New APIs
For additional documentation and command descriptions please refer to the Bluetooth API reference in the SDK installation or the online API reference specific to the SDK version you are using. The most up-to-date version is at **https://docs.silabs.com/bluetooth/latest/.**

Added in release 3.2.0.0

- sl_bt_connection_read_remote_used_features command: Read link layer features supported by a remote device.
- sl_bt_evt_connection_remote_used_features event: Indicate the link layer features supported by a remote device.
- sl_bt_gatt_server_read_client_supported_features command: Read GATT client supported features.
- sl_bt_gattdb_new_session command: Start a new GATT database update session.
- sl_bt_gattdb_add_service command: Add a service into the GATT database.
- sl_bt_gattdb_remove_service command: Remove a service from the GATT database.
- sl_bt_gattdb_add_included_service command: Add an included-service attribute into a service.
- sl_bt_gattdb_remove_included_service command: Remove an included-service attribute from a service.
- sl_bt_gattdb_add_uuid16_characteristic command: Add a 16-bit UUID characteristic into a service.
- sl_bt_gattdb_add_uuid128_characteristic command: Add a 128-bit UUID characteristic into a service.
- sl_bt_gattdb_remove_characteristic command: Remove a characteristic from a service.
- sl_bt_gattdb_add_uuid16_descriptor command: Add a 16-bit UUID descriptor into a characteristic.
- sl_bt_gattdb_add_uuid128_descriptor command: Add a 128-bit UUID descriptor into a characteristic.
- sl_bt_gattdb_remove_descriptor command: Remove a descriptor from a characteristic.
- sl_bt_gattdb_start_service command: Start a service so that it becomes visible to remote GATT clients.

- sl_bt_gattdb_stop_service command: Stop a service so that it becomes invisible to remote GATT clients.
- sl_bt_gattdb_start_characteristic command: Start a characteristic so that it becomes visible to remote GATT clients.
- sl_bt_gattdb_stop_characteristic command: Stop a characteristic so that it becomes invisible to remote GATT clients.
- sl_bt_gattdb_commit command: Save all changes performed in the current session to the GATT database and close the session. sl_bt_gattdb_abort command: Abort all changes performed in the current session to the GATT
- database and close the session.
- sl_bt_sm_get_bonding_handles command: Get the handles in the bonding database.
- sl_bt_sm_get_bonding_details command: Get the detailed information about a bonding.
- sl_bt_sm_find_bonding_by_address command: Find the bonding information by a Bluetooth device address.
- sl_bt_sm_set_legacy_oob command: Set the OOB data for legacy pairing.
- sl_bt_sm_set_oob command: Enable the use of OOB data for secure connections pairing.
- sl_bt_sm_set_remote_oob command: Set the OOB data and confirmation values received from the remote device for secure connec-tions pairing.
- SL_BT_COMPONENT_CONNECTIONS configuration: can be used by a component to configure the amount of Bluetooth connections it additionally needs.

Improvements

Changed APIs

Changed in release 3.2.2.0

- sl_bt_gap_set_privacy_mode() command: When the privacy mode is enabled with this command, advertiser addresses set with the sl_bt_advertiser_set_random_address() command are no longer updated by the stack autonomously. For each advertiser that uses the device's identity address, the stack periodically generates a new resolvable or non-resolvable private address for it in the privacy mode.
- sl_bt_advertiser_set_configuration() command: A new configuration item (value 16) has been added to allow an advertiser to use the global device identity address in the privacy mode. This configuration has no effect if the advertiser address has been set by the user application with the sl_bt_advertiser_set_random_address() command.
- sl_bt_sm_configure() command: A new option to select if the pairing should prefer just works or authenticated pairing when both options are possible based on the settings.

Changed in release 3.2.1.0
sl_bt_gattdb_commit() command: Previously, the stack removed the client characteristic configurations of all GATT clients except the service-changed configuration when the local GATT database was changed. This behavior has been changed so that, for connected GATT clients, the stack only removes the configurations of the removed characteristics.

Changed in release 3.2.0.0

- SL_BT_CONFIG_MAX_CONNECTIONS configuration: Moved to the bluetooth_feature_connection component

configuration file sl_bluetooth_connection_config.h.

- SL_BT_CONFIG_USER_ADVERTISERS configuration: Moved to the bluetooth_feature_advertiser component configuration file sl_bluetooth_advertiser_config.h.
- SL_BT_CONFIG_MAX_PERIODIC_ADVERTISING_SYNC configuration: Moved to the bluetooth_feature_sync component configu-ration file sl_bluetooth_periodic_sync_config.h.
- CTE Service UUIDs: values are updated according to the Bluetooth SIG specification.

Fixed Issues

Fixed in release 3.2.4.0

ID #	Description
735638	Fix a memory access violation when closing a Bluetooth connection if the Security Manager is uninitialized (i.e., the bluetooth_feature_sm component is unused). The violation has not caused any known functionality issue in released SDK versions.
736501	Add the app_properties.c file into RCP example projects for supporting firmware updates.
737292	Fix the issue that causes failures of connection establishments and scanning on the LE Coded PHY on EFR32[B|M]G21 devices.
740185	Fix a memory access violation when closing a Bluetooth connection that had a failed bonding operation. The violation has not caused any known functionality issue in released SDK versions.
740421	The Bluetooth controller now sends the correct number of bytes per packet for all connection intervals.
741923	Fix the issue that causes a failure of booting to bootloader from the HCI interface using the vendor-specific command 0xfc18.

Fixed in release 3.2.3.0

ID #	Description
738646	Fix a memory leak that occurs when a Bluetooth connection is opened. The issue was first introduced in Bluetooth SDK 3.2.0.

<ins>Fixed in release 3.2.2.0</ins>

ID #	Description
683223	Fix the issue that the TX power value passed to the sl_bt_test_dtm_tx_v4() command has no effect when testing a unmodulated carrier mode.
708049	Fix the issue that the DTM commands for unmodulated carrier wave transmission does not work before a DTM TX command for a modulated signal has been used.
714913	Fix a Bluetooth controller's task scheduling issue that causes Bluetooth disconnections during the scanning.
725480	Fix the issue that the connectionless aoa_locator app sometimes fails to synchronize to a tag.
728452	Fix the issue that the Bluetooth HCI component does not respond to the HCI reset command.
730386	LE Read Maximum Data Length HCI command now returns the correct maximum values that the controller supports.
731566	Fix a RTOS task hanging issue when initiating a disconnection while the Bluetooth connection is encrypted.
733857	Bluetooth HCI now reports completed ACL packets correctly to the host.

Fixed in release 3.2.1.0

ID #	Description
707252	Improvements in LE Power Control feature.
712526	Fix an issue with CTE (AoA/AoD) where device may enter into a hard fault if connectionless CTE or Silicon Labs CTE was enabled before connection creation.
714406	Fix for LL/DDI/SCN/BV-25-C.
715016	Fixed LE Power Control initialization.
715286	Now raising an error when subscribing to notifications or indications fail on characteristics that do not support those.
715414	Fix an issue in the HCI that advertisers cannot be disabled with number of sets setting to 0 in the LE Set Extended Advertising Enable command.
717381	Fix for Throughput example application to handle indication data correctly.
718466	Bluetooth 'NCP Interface' Component now defines the SL_BT_API_FULL macro, enabling all BGAPI command tables to be linked. This is needed by NCP target applications.
718867	Re-enabled whitelisting component support for soc_empty example app.
723935	Improvements in SoC Throughput example app.

Fixed in release 3.2.0.0

ID #	Description
649254	Previously user applications could set TX power higher than +10dBm even if the AFH (Adaptive Frequency Hopping) is not enabled. This has been fixed that the maximum usable TX power level is properly set and returned back to the user application if the AFH has not been enabled.
651247	Previously the Bluetooth stack on EFR32MG21 occasionally did not recognize a disconnection. This is a very rare case and the probability may increase with more RF noise in the environment. This issue has been fixed.
679431	Previously the DEBUG_EFM assert was triggered in Series 2 devices when creating a Bluetooth application from an empty project. This issue no longer exists in this release.
686213	Previously the Bluetooth stack occasionally could get stuck in an eternal loop. Assume an application has multiple GATT client connections performing simultaneous GATT procedures from both the sleep-timer interrupt context and the application main loop. In this case, a rare race condition might cause a stack memory corruption, which in turn causes a GATT procedure to fail to start. The issue does not exist if Bluetooth APIs are only called from the main loop (in bare metal mode) or an OS task (in RTOS mode).

The memory corruption issue in the use case above has been fixed. However, Bluetooth API commands cannot be called from interrupt contexts. Doing this might result in other unknown problems. This is described in **UG434: Silicon Labs Bluetooth® C Application Developers Guide for SDK v3.x.** |
696220	Fix an initialization issue that may cause another protocol using wrong RAIL configurations in a dynamic multiple protocol application.
696283	Fix a connection opening issue with an extended advertiser while the scanning is enabled.
697200	Fix a notation error in the Bluetooth stack RTOS configuration.
698227	Fix an issue that a task in Link Layer does not complete when radio gets stuck. This issue occurs very rarely and it could be reproducible in a busy environment with many advertisers, scanners and Bluetooth connections. The solution to the issue is the introduction of a radio watchdog (new feature component bluetooth_feature_radio_watchdog). A task will be aborted if the watchdog detects that the radio gets stuck. By default this feature is disabled to save memory.
700422	Fix a connection opening issue in central role when scanning simultaneously on different LE PHYs.
703303	Fix the firmware image filename extension in Bluetooth API sl_bt_dfu_flash_upload documentation.
703613	Fix compilation warnings using IAR, which are related to the usage of mbedTLS component in Bluetooth applications.
705969	Now the Radio can be initialized with VSCALE enabled on EFR32[B|M]G22 devices.
708029	Fix a Bluetooth connection issue that was caused by a defect on EFR32[B|M]G2[1|2] where the Power Manager fails to wake up from EM2 in certain situations.
714411	Fix an issue where connectionless CTE was transmitted on both AUX_ADV_IND and AUX_SYNC_IND packets. The correct behavior is to transmit it only on AUX_SYNC_IND packets.

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	Workaround
337467	MGM12P has poor signal strength when doing OTA with Apploader.	None
361592	The sync_data event does not report TX power.	None
368403	If setting CTE interval to 1, a CTE request should be sent in every connection interval. But it is sent only in every second connection interval.	None
641122	The Bluetooth stack component does not provide a configuration for RF antenna path.	This is an issue specifically for BGM210P. One workaround is to manually update the configuration in sl_bluetooth_config.h in text edit mode. If the OTA with Apploader is used, include the bluetooth_feature_ota_config component in application project. Call command sl_bt_ota_set_rf_path() to set the RF path for OTA mode.
650079	LE 2M PHY on EFR32[B|M]G12 and EFR32[B|M]G13 doesn't work with smartphones using the Mediatek Helio chip due to an interoperability issue.	No workaround exists. For application development and testing, the disconnection can be avoided by disabling 2M PHY with sl_bt_connection_set_preferred_phy() or sl_bt_connection_set_default_preferred_phy().
682198	The Bluetooth stack has an interoperability issue on the 2M PHY with a Windows PC.	No workaround exists. For application development and testing, the disconnection can be avoided by disabling 2M PHY with sl_bt_connection_set_preferred_phy() or sl_bt_connection_set_default_preferred_phy().
695148	Bluetooth soft timer doesn't work when Bluetooth on-demand start feature is enabled.	Use the simple timer component in the Bluetooth SDK or the sleeptimer platform service.
725498	**The connection-based aoa_locator application sometimes crashes with error message Failed to enable CTE.**	**None**
730692	**4-7% packet error rate is observed on EFR32[B|M]G13 devices when RSSI is between -25 and -10 dBm. The PER is nominal (as per the datasheet) both above and below this range.**	**None**

Deprecated Items

Deprecated in release 3.2.1.0

- **API enum sl_bt_gap_phy_type_t**
  This enum type is replaced by sl_bt_gap_phy_t.

- **API enum sl_bt_gap_phy_and_coding_type_t**

  This enum type is replaced by sl_bt_gap_phy_coding_t.

  Old types are still valid and can be used in applications. It is recommend to migrate to the new types as soon as possible. Old types will be removed in no less than one year in a future major SDK release.

Deprecated in release 3.2.0.0

- **API command sl_bt_sm_list_bonding_entry**

  This command is replaced by sl_bt_sm_get_bonding_handles and sl_bt_sm_get_bonding_details commands.
- **API command sl_bt_sm_set_oob_data**

  This command is replaced by the command sl_bt_sm_set_legacy_oob.
- **API command sl_bt_sm_use_sc_oob**

  This command is replaced by the command sl_bt_sm_set_oob.
- **API command sl_bt_sm_set_sc_remote_oob_data**

  This command is replaced by the command sl_bt_sm_set_remote_oob.
- **API commands sl_bt_system_set_soft_timer and sl_bt_system_set_lazy_soft_timer**

  Bluetooth APIs do not provide a replacement. Use the simple timer component in the Bluetooth SDK or the sleeptimer platform service for timers.

AoA Compass Demo
To be removed in a future release. This demo is replaced by the AoA Analyzer.

ncp_empty example application
To be removed in a future release. This example is replaced by the ncp example.

Removed Items

Removed in release 3.2.0.0
BGTool
BGTool is removed in this release and replaced by the Bluetooth NCP Commander which includes a modern, intuitive, web-based user interface as well as a smart console with IntelliSense and built-in API documentation.

Using This Release

This release contains the following

- Silicon Labs Bluetooth stack library
- Bluetooth sample applications

For more information about the Bluetooth SDK see QSG169: Bluetooth® SDK v3.x Quick Start Guide. If you are new to Bluetooth see UG103.14: Bluetooth LE Fundamentals.

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://sili-conlabs.force.com/apex/SL_CommunitiesSelfReg?form=short**.
Stack installation instruction are covered in the Simplicity Studio 5 online User's Guide.
Use the Bluetooth SDK v3.x with the Silicon Labs Simplicity Studio 5 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. Only use Simplicity Studio 4 with Bluetooth SDK v2.13.x and lower. 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**/.

Security Information

Secure Vault Integration

When deployed to Secure Vault High devices, sensitive keys such as the Long Term Key (LTK) are protected using the Secure Vault Key Management functionality. The table below shows the protected keys and their storage protection characteristics.

Wrapped Key	Exportable / Non-Exportable	Notes
Remote Long Term Key (LTK)	Non-Exportable	
Local Long Term Key (legacy only)	Non-Exportable	
Remote Identity Resolving Key (IRK)	Exportable	Must be Exportable for future compatibility reasons
Local Identity Resolving Key	Exportable	Must be Exportable because the key is shared with other devices.

Wrapped keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime. Wrapped keys that are marked as "Exportable" can be used or shared at runtime but remain encrypted while stored in flash. For more information on Secure Vault Key Management functionality, see AN1271: Secure Key Storage.

Security Advisories

To subscribe to Security Advisories, log in to the Silicon Labs customer portal, then select Account Home. Click HOME to go to the portal home page and then click the Manage Notifications tile. Make sure that 'Software/Security Advisory Notices & Product Change Notices (PCNs)' is checked, and that you are subscribed at minimum for your platform and protocol. Click Save to save any changes.

Support

Development Kit customers are eligible for training and technical support. Use the Silicon Labs Bluetooth LE web page to obtain infor-mation 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**.

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

- **IoT Portfolio**

  www.silabs.com/IoT
- **SW/HW**

  www.silabs.com/simplicity
- **Quality**

  www.silabs.com/quality

- **Support & Community**

  www.silabs.com/community

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 imple-menters 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 to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the infor-mation supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. 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. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon Labs product in such unauthorized applications. Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit **www.silabs.com/about-us/inclusive-lexicon-project**

Trademark Information
Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals®, WiSeConnect , n-Link, ThreadArch®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, Gecko OS, Gecko OS Studio, Precision32®, Simplicity Studio®, Telegesis, the Telegesis Logo®, USBXpress® , Zentri, the Zentri logo and Zentri DMS, Z-Wave®, and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.

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

Documents / Resources

SILICON LABS Bluetooth LE SDK Software [pdf] User Guide
Bluetooth LE SDK Software, Bluetooth LE, SDK Software, Software

References

- **[PyPI · The Python Package Index](#)**
- **[Silicon Labs](#)**
- **[Silicon Labs](#)**
- **[About Us - Silicon Labs](#)**
- **[Silicon Labs Community](#)**
- **[Internet of Things (IoT) - Silicon Labs](#)**
- **[Quality - Silicon Labs](#)**
- **[Simplicity Studio - Silicon Labs](#)**
- **[Technical Support - Silicon Labs](#)**
- **[Software Developer Docs - Silicon Labs](#)**
- **[Developing with Bluetooth - latest - Bluetooth LE Silicon Labs](#)**
- **[Overview - latest - Simplicity Studio 5 Users Guide Silicon Labs](#)**
- **[GitHub - SiliconLabs/pybgapi-examples](#)**
- **[pybgapi · PyPI](#)**
- **[Silicon Labs New User Registration](#)**
- **[Silicon Labs Community](#)**
- **[Simplicity Studio - Silicon Labs](#)**
- **[Bluetooth Low Energy (LE) Software Development Kit - Silicon Labs](#)**
- **[Simplicity Studio - Silicon Labs](#)**
- **[User Manual](#)**