



Bluetooth[®] mesh SDK 6.0.0.0 GA

Gecko SDK Suite 4.4

December 13, 2023

Bluetooth mesh is a new topology available for Bluetooth Low Energy (LE) devices that enables many-to-many (m:m) communication. It's optimized for creating large-scale device networks, and is ideally suited for building automation, sensor networks, and asset tracking. Our software and SDK for Bluetooth development supports Bluetooth Mesh and Bluetooth 5.3 functionality. Developers can add mesh networking communication to LE devices such as connected lights, home automation, and asset tracking systems. The software also supports Bluetooth beaconing, beacon scanning, and GATT connections so Bluetooth mesh can connect to smart phones, tablets, and other Bluetooth LE devices.

This release includes features supported by the Bluetooth mesh specification version 1.1.

These release notes cover SDK versions:

6.0.0.0 released December 13, 2023



KEY FEATURES

- Qualified implementation of Mesh 1.1
- Added Network Lighting Control (NLC) profiles

Compatibility and Use Notices

For more 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, or if you are new to the Silicon Labs Bluetooth mesh SDK, see [Using This Release](#).

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 9.40.1

- 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 12.2.1, provided with Simplicity Studio.

- Link-time optimization feature of GCC has been disabled, resulting in a slight increase of image size.

Contents

- 1 New Items 2
 - 1.1 New Features 2
 - 1.2 New APIs..... 3
- 2 Improvements 5
- 3 Fixed Issues 6
- 4 Known Issues in the Current Release 7
- 5 Deprecated Items 8
- 6 Removed Items 9
- 7 Using This Release 10
 - 7.1 Installation and Use 10
 - 7.2 Security Information..... 10
 - 7.3 Support..... 11

1 New Items

This release of the Gecko SDK (GSDK) will be the last with combined support for all EFM and EFR devices, except for patches to this version as needed. Starting in mid-2024 we will introduce separate SDKs:

- The existing Gecko SDK will continue with support for Series 0 and 1 devices.
- A new SDK will cater specifically to Series 2 and 3 devices.

The Gecko SDK will continue to support all Series 0 and 1 devices with no change to the long-term support, maintenance, quality, and responsiveness provided under our software policy.

The new SDK will branch from Gecko SDK and begin to offer new features that help developers take advantage of the advanced capabilities of our Series 2 and 3 products.

This decision aligns with customer feedback, reflecting our commitment to elevate quality, ensure stability, and enhance performance for an exceptional user experience across our software SDKs.

1.1 New Features

Added in release 6.0.0.0

New Networked Lighting Control (NLC) example apps:

btmesh_soc_nlc_basic_lightness_controller for demonstration of the BT Mesh NLC Basic Lightness Controller Profile

btmesh_soc_nlc_basic_scene_selector for demonstration of the BT Mesh NLC Basic Scene Selector Profile

btmesh_soc_nlc_dimming_control for demonstration of the BT Mesh NLC Dimming Controller Profile

btmesh_soc_nlc_sensor_ambient_light for demonstration of the BT Mesh NLC Ambient Light Sensor Profile

btmesh_soc_nlc_sensor_occupancy for demonstration of the BT Mesh NLC Occupancy Sensor Profile (People count)

Changes in example apps:

btmesh_soc_sensor_server was deleted and its functionality was split in 3 examples:

- **btmesh_soc_sensor_thermometer** for demonstration of Sensor Server Model with a thermometer
- **btmesh_soc_nlc_sensor_occupancy** for demonstration of the BT Mesh NLC Occupancy Sensor Profile (people count)
- **btmesh_soc_nlc_sensor_ambient_light** for demonstration of the BT Mesh NLC Ambient Light Sensor Profile

btmesh_soc_switch was renamed to **btmesh_soc_switch_ctl**, whose purpose is to demonstrate the usage of the Light CTL Client Model. The example no longer controls scenes (Scene Client)

btmesh_soc_light was renamed to **btmesh_soc_light_ctl**

The example no longer demonstrates the LC Server model and Scene Server, Scheduler Server and Time Server Models

btmesh_soc_hsl was renamed to **btmesh_soc_light_hsl**

The example no longer demonstrates the LC Server model and Scene Server, Scheduler Server and Time Server Models

Changes in all example apps:

DFU image updates are generated by a Python script instead of create_bl_files.bat/.sh files

Support for Mesh Composition Data Pages 1, 2, 128, 129, 130 was added for all examples, these pages are automatically generated by the BT Mesh Configurator tool.

New SLC components:

btmesh_nlc_basic_lightness_controller for demonstration of the BT Mesh NLC Basic Lightness Controller Profile

btmesh_nlc_basic_lightness_controller_profile_metadata for Composition Data Page 2 NLC support for Basic Lightness Controller Profile

btmesh_nlc_basic_scene_selector for demonstration of the BT Mesh NLC Basic Scene Selector Profile

btmesh_nlc_basic_scene_selector_profile_metadata for Composition Data Page 2 NLC support for Basic Scene Selector Profile

btmesh_nlc_dimming_control for demonstration of the BT Mesh NLC Dimming Controller Profile

btmesh_nlc_dimming_control_profile_metadata for Composition Data Page 2 NLC support for Dimming Controller Profile

btmesh_nlc_ambient_light_sensor for demonstration of the BT Mesh NLC Ambient Light Sensor Profile

btmesh_nlc_ambient_light_sensor_profile_metadata for Composition Data Page 2 NLC support for Ambient Light Sensor Profile

btmesh_nlc_occupancy_sensor for demonstration of the BT Mesh NLC Occupancy Sensor Profile (People count)

btmesh_nlc_occupancy_sensor_profile_metadata for Composition Data Page 2 NLC support for Occupancy Sensor Profile

btmesh_generic_level_client_ext for extending the Generic Base component with Generic Move Unacknowledged and Generic Delta Unacknowledged messages

ncp_btmesh_ae_server for enabling the Silabs Configuration Server vendor model for the node to allow data transfer over Advertisement Extension

ncp_btmesh_ae_server for enabling the Silabs Configuration Client vendor model for the node.

ncp_btmesh_user_cmd for demonstrating the communication between an NCP host and NCP target using BGAPI user messages, responses and events.

1.2 New APIs**Added in release 6.0.0.0****Changes in SLC components:**

ncp_btmesh_dfu component's `ncp_btmesh_dfu.h` has a new API

```
void sl_btmesh_ncp_dfu_handle_cmd(void *data, bool *cmd_handled);
```

btmesh_provisioning_decorator component doesn't restart provisioning after provisioning is failed

btmesh_lighting_server's `sl_btmesh_lighting_server.h` has a new API

```
void sl_btmesh_update_lightness(uint16_t lightness, uint32_t remaining_ms);
```

btmesh_event_log has more granular configurability options

btmesh_ctl_client's `sl_btmesh_ctl_client.h` has an API change

instead of

```
void sl_btmesh_set_temperature(uint8_t new_color_temperature_percentage);
```

the new API is

```
void sl_btmesh_ctl_client_set_temperature(uint8_t temperature_percent);
```

```
void sl_btmesh_ctl_client_set_lightness(uint8_t lightness_percent);
```

BGAPI additions:

A new BGAPI class for device diagnostics has been added. It provides the application with Mesh stack statistics counters and an event-based reporting of network PDU relaying and proxying, which can be activated and deactivated as needed.

The BGAPI commands in the diagnostic class are:

- `sl_btmesh_diagnostic_init`: Initialize the diagnostic component
- `sl_btmesh_diagnostic_deinit`: Deinitialize the diagnostic component
- `sl_btmesh_diagnostic_enable_relay`: Enable event-based reporting of network PDU relaying/proxying activity
- `sl_btmesh_diagnostic_disable_relay`: Disable event-based reporting of network PDU relaying/proxying activity
- `sl_btmesh_diagnostic_get_relay`: Get the number of relayed/proxied network PDUs so far
- `sl_btmesh_diagnostic_get_statistics`: Get mesh stack statistics counters
- `sl_btmesh_diagnostic_clear_statistics`: Zero mesh stack statistics counters

The BGAPI event in the diagnostic class is:

- `sl_btmesh_diagnostic_relay`: Event reporting that a network PDU has been relayed or proxied by the stack

2 Improvements

Changed in release 6.0.0.0

A provisioner or a node can now configure itself using the configuration client model and its own primary address as the destination for the messages. This can replace self-configuration by test BGAPI commands.

Code optimization can result in slightly smaller firmware images than before, depending on the feature set used.

Code optimization can result in slightly smaller RAM usage than before, depending on the feature set used.

The Mesh stack no longer requires or supports the deprecated BLE advertiser and scanner components. Instead, it uses the current versions of each (legacy advertiser and legacy scanner for non-extended advertisements, and extended advertiser and extended scanner for extended advertisements). Applications that use both the BLE and the Mesh BGAPIs should no longer use the deprecated BLE advertiser and scanner components either.

3 Fixed Issues

Fixed in release 6.0.0.0

ID #	Description
360955	The interval between first and second attention timer event could be other than one second.
1198887	Private beacon random advertiser address is the same for all subnets while it should be different.
1202073	Btmesh_ncp_empty example does not have enough RAM on BRD4182 with GCC compiler.
1202088	Btmesh_soc_switch example does not have enough RAM on BRD4311 and BRD4312 with IAR compiler
1206714	Proxy server should emit a beacon over proxy connection when a subnet is added to the proxy server
1206715, 1211012, 1211022	Support for device composition data page 2, 129 and 130 should be present in configuration server model as well as large composition data server model when remote provisioning is supported
1211017	Periodic publishing of location information should alternate between global and local location when both are known
1212373	Resource leak in proxy connection handling after several hundred proxy connections have been opened and closed
1212854	Pull mode MBT transfer to an LPN does not complete successfully
1197398, 1194443	DFU distributor application is currently not able to handle more than 60 nodes successfully
1202088	Btmesh_soc_switch_ctl example compiles on all boards with IAR compiler.

4 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
401550	No BGAPI event for segmented message handling failure.	Application needs to deduce failure from timeout / lack of application layer response; for vendor models an API has been provided.
454059	A large number of key refresh state change events are generated at the end of KR process, and that may flood NCP queue.	Increase NCP queue length in the project.
454061	Slight performance degradation compared to 1.5 in round-trip latency tests was observed.	
624514	Issue with re-establishing connectable advertising if all connections have been active and GATT proxy is in use.	Allocate one more connection than is needed.
841360	Poor performance of segmented message transmission over GATT bearer.	Ensure that the underlying BLE connection's Connection interval is short; ensure that ATT MTU is large enough to fit a full Mesh PDU; tune the minimum connection event length to allow multiple LL packets to be transmitted per connection event.
1121605	Rounding errors may cause scheduled events to trigger at very slightly different times than expected.	
1226000	BGAPI command for checking node identity advertisements only works with regular node identity advertisements, not private node identity advertisements.	
1206620	Insufficient buffers can cause some DFU target nodes to fail DFU with the error "verification failed" in large network updates (50+ nodes) when background traffic is sufficiently high.	Increase the configuration value <code>SL_BT_CONFIG_BUFFER_SIZE</code> to 4000 or more. Alternatively, the nodes that report image verification failure can be updated in a second DFU process. After the traffic from the first DFU process has stopped, and most nodes are already updated, it is unlikely for the nodes to run out of buffers anymore.
1226127	Host provisioner example can be stuck when it starts to provision a second node	Restart the host provisioner app before provisioning the second node
1204017	Distributor is not able to handle parallel self FW Update and FW Upload	Don't run self FW update and FW upload in parallel

5 Deprecated Items

Deprecated in release 6.0.0.0

The BGAPI command `sl_btmesh_node_get_networks()` has been deprecated. Use `sl_btmesh_node_key_key_count()` and `sl_btmesh_node_get_key()` instead.

The BGAPI commands `sl_btmesh_test_set_segment_send_delay()` and `sl_btmesh_test_set_sar_config()` have been deprecated. Use `sl_btmesh_sar_config_set_sar_transmitter()` and `sl_btmesh_sar_config_server_set_sar_receiver()` instead.

6 Removed Items

Removed in release 6.0.0.0

The BGAPI commands `sl_btmesh_test_set_local_config()` and `sl_btmesh_test_get_local_config()` have been removed.

The BGAPI commands `sl_btmesh_node_get_statistics()` and `sl_btmesh_node_clear_statistics()` have been removed.

7 Using This Release

This release contains the following

- Silicon Labs Bluetooth mesh stack library
- Bluetooth mesh sample applications

If you are a first time user, see *QSG176: Silicon Labs Bluetooth Mesh SDK v2.x Quick-Start Guide*.

7.1 Installation and Use

The Bluetooth mesh SDK is provided as part of the Gecko SDK (GSDK), the suite of Silicon Labs SDKs. To quickly get started with the GSDK, install [Simplicity Studio 5](#), which will set up your development environment and walk you through GSDK installation. Simplicity Studio 5 includes everything needed for IoT product development with Silicon Labs devices, including a resource and project launcher, software configuration tools, full IDE with GNU toolchain, and analysis tools. Installation instructions are provided in the online [Simplicity Studio 5 User's Guide](#).

Alternatively, Gecko SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/gecko_sdk for more information.

The GSDK default install location has changed with Simplicity Studio 5.3 and higher.

- Windows: C:\Users\\SimplicityStudio\SDKs\gecko_sdk
- MacOS: /Users/<NAME>/SimplicityStudio/SDKs/gecko_sdk

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 Security Information

Secure Vault Integration

This version of the stack is integrated with Secure Vault Key Management. When deployed to Secure Vault High devices, mesh encryption keys are protected using the Secure Vault Key Management functionality. The table below shows the protected keys and their storage protection characteristics.

Key	Exportability on a node	Exportability on Provisioner	Notes
Network key	Exportable	Exportable	Derivations of the network key exist only in RAM while network keys are stored on flash
Application key	Non-exportable	Exportable	
Device key	Non-exportable	Exportable	In Provisioner's case, applied to Provisioner's own device key as well as other devices' keys

Keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.

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.

SILICON LABS Search Within the Support Portal for Cases, etc... SEARCH CATHERIN...

HOME CASES SOFTWARE RELEASES

Update Preference

WHAT EMAILS WOULD YOU LIKE TO RECEIVE?

Newsletters

- Community Monthly Newsletter
- Sales Newsletter
- Micrium Newsletter

Product Specific Notifications

- Product Information and Newsletter
- Software/Security Advisory Notices & Product Change Notices (PCNs)
- Technical Document Updates (Release Notes, Data Sheets, etc.)

SELECT THE PRODUCTS TO RECEIVE UPDATES FOR

Select/Unselect All

<input type="checkbox"/> Audio and Radio	<input type="checkbox"/> Power over Ethernet
<input type="checkbox"/> Interface	<input type="checkbox"/> Sensors
<input type="checkbox"/> Isolation	<input type="checkbox"/> TV and Video
<input type="checkbox"/> Modems and DAAs	<input type="checkbox"/> Voice
<input type="checkbox"/> Microcontrollers	<input type="checkbox"/> Wireless
<input type="checkbox"/> 8-bit MCUs	<input type="checkbox"/> Bluetooth Classic
<input checked="" type="checkbox"/> 32-bit MCUs	<input type="checkbox"/> Bluetooth Low Energy
<input type="checkbox"/> Timing	<input checked="" type="checkbox"/> Proprietary
<input type="checkbox"/> Clocks	<input type="checkbox"/> Wi-Fi
<input type="checkbox"/> Buffers	<input type="checkbox"/> ZigBee and Thread
<input type="checkbox"/> Oscillators	<input type="checkbox"/> Z-Wave
<input type="checkbox"/> CDR and PHY	

7.3 Support

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

Contact Silicon Laboratories support at <http://www.silabs.com/support>.

Simplicity Studio

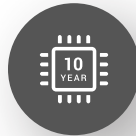
One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT



SW/HW
www.silabs.com/simplicity



Quality
www.silabs.com/quality



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 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 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 information 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