Silicon Labs *Bluetooth*® mesh SDK 1.5.0.1 GA

19Q2 Gecko SDK

June 25, 2019

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 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.

These release notes cover SDK version:

1.5.0.1 released June 25, 2019
1.5.0.0 released June 14, 2019

KEY FEATURES

- Added support for EFR32BG13 and EFR32MG13 Rev D parts
- Added support for EFR32BG12 QFN68 I-Grade parts
- Added sensor model API, documentation, and sample applications
- Selected quality improvements and bug fixes

Compatibility and Use Notices

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 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.

- Customers on macOS may experience linking errors due to a bug in GNU binutils used with GCC 7.2.1

Contents

1 New Items

1.1 New Features

Added in release 1.5.0.0

Models: support for sensor models (sensor client, sensor server, sensor setup server) has been added

1.2 New APIs

For additional documentation please refer to the Bluetooth Mesh Software API Reference Manual installed with the Bluetooth Mesh SDK.

Added in release 1.5.0.0

BGAPI commands and events for sensor models have been added

Sensor client model commands and events:

mesh_sensor_client_init(),
mesh_sensor_client_deinit(),
mesh_sensor_client_get(),
mesh_sensor_client_get_cadence(),
mesh_sensor_client_get_column(),
mesh_sensor_client_get_descriptior(),
mesh_sensor_client_get_series(),
mesh_sensor_client_get_setting(),
mesh_sensor_client_get_settings(),
mesh_sensor_client_set_cadence(),
mesh_sensor_client_set_setting(),
mesh_sensor_client_descriptor_status(),
mesh_sensor_client_cadence_status(),
mesh_sensor_client_settings_status(),
mesh_sensor_client_setting_status(),
mesh_sensor_client_status(),
mesh_sensor_client_column_status(),
mesh_sensor_client_series_status(),
mesh_sensor_client_publish()

Sensor server model commands and events:

mesh_sensor_server_init(),
mesh_sensor_server_deinit(),
mesh_sensor_server_send_column_status(),
mesh_sensor_server_send_descriptor_status(),
mesh_sensor_server_send_series_status(),
mesh_sensor_server_send_status(),
mesh_sensor_server_get_request(),
mesh_sensor_server_get_column_request(),
mesh_sensor_server_get_series_request(),
mesh_sensor_server_publish()

Sensor setup server model commands and events:

mesh_sensor_setup_server_send_cadence_status(),
mesh_sensor_setup_server_send_settings_status(),
mesh_sensor_setup_server_send_setting_status(),
mesh_sensor_setup_server_get_cadence_request(),
mesh_sensor_setup_server_set_cadence_request(),
mesh_sensor_setup_server_get_settings_request(),
mesh_sensor_setup_server_get_setting_request(),
mesh_sensor_setup_server_set_setting_request(),
mesh_sensor_setup_server_publish()

2 Improvements

2.1 Changed APIs

Changed in release 1.5.0.0

A number of new BGAPI commands and events have been added to various command classes. One LPN command has been deprecated.

Added provisioner commands and events:

mesh_prov_stop_scan_unprov_beacons()

Added node commands and events:

mesh_node_key_removed(),
mesh_node_key_updated(),
mesh_node_reset()

Added LPN commands and events:

mesh_lpn_config()

Deprecated LPN commands and events:

mesh_lpn_configure() – use mesh_lpn_config() instead

Added testing BGAPI commands and events:

mesh_test_set_element_seqnum()

2.2 Changed Documents

Changed in release 1.5.0.1

The quick start guide e QSG148: Getting Started with Bluetooth® Mesh Software Development has been updated.

3 Fixed Issues

Fixed in release 1.5.0.0

ID #	Description
3046, 6166	Node now erases all internal state when told to reset by the Provisioner, instead of only key material
4800	Initial publishing of model state is deferred by a pseudorandom amount to avoid collisions when multiple devices are powered on at the same time
5601	Fixed issue with activating GATT provisioning service advertisements even if only ADV provisioning bearer was selected
5770, 6203	Fixed issue with GATT proxy state change events missing on proxy server side
6005	Fixed issue with allowing multiple concurrent GATT proxy connections
6166	Stack now erases all mesh-related persistent storage when told to factory reset, instead of just encryption keys
6167	Provisioner replies with an immediate error if asked to provision another device with an address already in use
6196	Fixed two's logarithm representation in heartbeat publication count
6222	Fixed issue with sequence number updating to persistent storage, triggered by query for element remaining sequence numbers
6267	Fixed issue with unit conversion in LPN poll timeout value
6295	Fixed DCD data passing to application when mesh configuration client BGAPI was used
6304	Fixed sequence number clearing in persistent storage when IV index update takes place
6363	Fixed issue with battery model state serialization

4 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
N/A	Implementation of time, scene, scheduler and some lighting models not yet done	
3878	Mesh GATT events visible to the application	Application can ignore BGAPI events related to GATT provisioning and proxying based on service and characteristic parameters
4975	GCC linking with link-time optimization may fail on OSX due to an issue in GNU binutils	
5662	Default device UUID does not conform to RFC4122	Customer needs to explicitly set UUID to a conformant one
6460	**Default LPN parameters may result in friendship loss**	**Override LPN default parameters with mesh_lpn_config(), setting mesh_lpn_retry_interval to 50 and mesh_lpn_request_retries to 6**

5 Deprecated Items

Deprecated in release 1.5.0.0

As of June 2019 Simplicity Studio 3.0 is being deprecated. All access will be removed from Silicon Labs' website at the end of 2019.

6 Removed Items

Removed in release 1.5.0.0

None.

7 Using This Release

This release contains the following

- Silicon Labs Bluetooth mesh stack library
- Bluetooth sample applications

If you are a first time user, see QSG148: Getting Started with Bluetooth® Mesh Software Development.

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 QSG148: Getting Started with Bluetooth® Mesh Software Development.

Use the Bluetooth mesh 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 mesh 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-M3 and THUMB are trademarks or registered trademarks of ARM Holdings.

Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.