



Silicon Labs OpenThread SDK 2.5.2.0 GA

Simplicity SDK Suite 2024.6.2

September 18, 2024

Thread is a secure, reliable, scalable, and upgradeable wireless IPv6 mesh networking protocol. It provides low-cost bridging to other IP networks while optimized for low-power / battery-backed operation. The Thread stack is designed specifically for Connected Home applications where IP-based networking is desired, and a variety of application layers may be required.

OpenThread released by Google is an open-source implementation of Thread. Google has released OpenThread to accelerate the development of products for the connected home and commercial buildings. With a narrow platform abstraction layer and a small memory footprint, OpenThread is highly portable. It supports system-on-chip (SoC), network co-processor (NCP), and radio co-processor (RCP) designs.

Silicon Labs has developed an OpenThread-based SDK tailored to work with Silicon Labs hardware. The Silicon Labs OpenThread SDK is a fully tested enhanced version of the GitHub source. It supports a broader range of hardware than does the GitHub version, and includes documentation and example applications not available on GitHub.

These release notes cover SDK version(s):

2.5.2.0 GA released on September 18, 2024

2.5.1.0 GA released on July 24, 2024

2.5.0.0 GA released on June 5, 2024



KEY FEATURES

OpenThread

- Thread 1.3.0 certification compliance with Thread Test Harness v59.0 for SoC and Host-RCP architectures
- Thread 1.4 features - Alpha / Experimental
- Coex Metrics enhancements
- OpenThread Mesh Performance Results (AN1408)
- New Part support:
 - MG26 support – Alpha
- Removed support for Series 0/1

Multi-Protocol

- ZigbeeD and OTBR support on OpenWRT – Alpha
- DMP BLE + CMP ZB & Matter/OT – Alpha
- Removed support for Series 0/1

Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Platform Release Notes installed with this SDK or on the TECH DOCS tab on <https://www.silabs.com/developers/thread>. 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 OpenThread SDK, see [Using This Release](#).

Compatible Compilers:

GCC (The GNU Compiler Collection) version 12.2.1, provided with Simplicity Studio.

Contents

1	New Items	3
1.1	New Components	3
1.2	New Features	3
1.3	New APIs	3
1.4	New Radio Board Support	3
2	Improvements	4
3	Fixed Issues	5
4	Known Issues in the Current Release	6
5	Deprecated Items	7
6	Removed Items	8
7	Multiprotocol Gateway and RCP	9
7.1	New Items	9
7.2	Improvements	9
7.3	Fixed Issues	9
7.4	Known Issues in the Current Release	10
7.5	Deprecated Items	10
7.6	Removed Items	10
8	Using This Release	11
8.1	Installation and Use	11
8.2	OpenThread GitHub Repository	11
8.3	OpenThread Border Router GitHub Repository	11
8.4	Using the Border Router	11
8.5	NCP/RCP Support	12
8.6	Security Information	12
8.7	Support	13
8.8	Thread Certification	13

1 New Items

Simplicity SDK is an embedded software development platform for building IoT products based on our Series 2 and Series 3 wireless and MCU devices. It integrates wireless protocol stacks, middleware, peripheral drivers, a bootloader, and application examples – a solid framework for building power-optimized and secure IoT devices.

The Simplicity SDK offers powerful features such as ultra-low power consumption, strong network reliability, support for a large number of nodes, and abstraction of complex requirements like multiprotocol and pre-certification. Additionally, Silicon Labs provides over-the-air (OTA) software and security updates to remotely update devices, minimize maintenance costs, and enhance the end-user product experience.

Simplicity SDK is a follow-on from our popular Gecko SDK, which will continue to be available providing long-term support for our Series 0 and Series 1 devices. For additional information on the Series 0 and Series 1 devices please reference: [Series 0 and Series 1 EFM32/EZR32/EFR32 device \(silabs.com\)](#).

1.1 New Components

Added in release 2.5.0.0

- `ot_core_vendor_extension` - This component implements the `ot::Extension::ExtensionBase` interface for EFR32. When used with the OT Crash Handler component, crash info will be printed after the OpenThread instance is initialized.

1.2 New Features

Added in release 2.5.0.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- Libraries and sample applications in this SDK default to Thread 1.3. Thread 1.4 version and features, including credential sharing and ephemeral key support, are supported but are not ready for certification and still have ongoing development.
- Optimization to the OpenThread API to allow for thread-safe access to the OpenThread stack in a multi-threaded RTOS environment.
- A new platform component, Clock Manager, is integrated to all the openthread sample applications for clock configuration.
- A new platform component, Memory Manager, is integrated to all the openthread sample applications for dynamic memory management.
- OTBR on OpenWRT. A new multiprotocol openthread-br ipk package has been added for use in the OpenWRT host environment.
- Support for OpenThread Coex metrics

1.3 New APIs

Added in release 2.5.0.0

- Support for OpenThread diag channel and transmit power API and CLI commands: `otPlatDiagTxPowerSet()` and `otPlatDiagChannelSet()`.

1.4 New Radio Board Support

Added in release 2.5.0.0

Support has been added for the following radio boards:

- BRD4116A - EFR32MG26B410F3200IM48-A

2 Improvements

Changed in release 2.5.0.0

- The `ot_rcp_gp` interface component has been renamed to `ot_gp_interface`.
- The openthread API includes all changes up to and including the updates described here: https://openthread.io/reference/api-updates#may_17_2024.
- The set of prebuilt demo sample applications has been updated as follows:
 - `ot-ble-dmp` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `ot-ble-dmp-no-buttons` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `ot-cli-ftd` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `ot-cli-mtd` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `ot-rcp` – (brd2703a, brd4116a, brd4180a, brd4180b, brd4186c)
 - `rcp-spi-802154` – (brd4116a, brd4180a, brd4180b, brd4186c)
 - `rcp-spi-802154-blehci` – (brd4116a, brd4180a, brd4180b, brd4186c)
 - `rcp-uart-802154` – (brd2703a, brd4116a, brd4180a, brd4180b, brd4186c)
 - `rcp-uart-802154-blehci` – (brd2703a, brd4116a, brd4180a, brd4180b, brd4186c)
 - `sleepy-demo-ftd` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `sleepy-demo-mtd` – (brd2703a, brd4116a, brd4180a, brd4186c)
 - `sleepy-demo-ssed` – (brd2703a, brd4116a, brd4180a, brd4186c)
- Increased `OPENTHREAD_CONFIG_CSL_RECEIVE_TIME_AHEAD` to 750 us. for default FTD and MTD certification libraries for use with devices under test using library-based certification.
- For older project files (.slcps) the default config value of `CIRCULAR_QUEUE_LEN_MAX` must be greater than or equal to `SL_OPENTHREAD_RADIO_RX_BUFFER_COUNT` (i.e. 16). Otherwise, the validation script will fail the project generation. This can be done by adding the config value entry in the .slcp directly or from Simplicity Studio: go to the 'Circular Queue' component, then go to configuration, and set the max length as per `SL_OPENTHREAD_RADIO_RX_BUFFER_COUNT`.
- For multipan RCP applications, replace existing `app.c` with `<GSDK_DIR>/protocol/openthread/sample-apps/ot-ncp/app.c` in existing project or port changes from `<GSDK_DIR>/protocol/openthread/sample-apps/ot-ncp/app.c` to custom `app.c` if you have one.

3 Fixed Issues

Fixed in release 2.5.2.0

ID #	Description
1238120 / 1334227	In a multiprotocol RCP environment certain incoming packets were wrongly interpreted as Zigbee Green Power packets triggering a random outgoing packet. The issue is fixed by strengthening the Zigbee Green Power packet detection on the RCP.
1289835 / 1334618	Fixed issue with the openthread border router dropping more packets than expected when running within a docker container and under duress.
1295848 / 1334987	Fixed occasional occurrence of duplicate header bytes that were causing rcp resets in busy networks with a lot of traffic over SPI. Also, it is recommended to use a higher spi-bus speed (such as 4Mhz - depending on the network traffic) with a reduced debug logging level to have stable communication between host and rcp. Higher spi speed on OTBR can be set in radio-url as 'spi-speed=<>' argument.
1327029 / 1337439	Added support for uartdrv_eusart:vcom instance in OpenThread applications.
1329286 / 1334038	Removed the "-Werror=unused-parameter" GCC compiler option so that applications that have unused parameters can build.

Fixed in release 2.5.1.0

ID #	Description
1295833	The following was added to NCP README to better reflect the actual status of the NCP samples apps. "Note that the NCP design model is not supported or tested, although experimental support is available with the OpenThread stack."

Fixed in release 2.5.0.0

ID #	Description
1208578	Added support for linking the CPC library to Posix host apps by using provided paths, and by using pkg-config.
1235923	Fixed a wrapping bug in calls to otPlatAlarmMilliStartAt and otPlatAlarmMicroStartAt.
1238120	In a multiprotocol RCP environment, certain incoming packets were wrongly interpreted as Zigbee Green Power packets, triggering a random outgoing packet. The issue is fixed by strengthening the Zigbee Green Power packet detection on the RCP.
1243597	Removed extra ot-ble-dmp-no-buttons sample apps from demos folder.
1249346 / 1255247	Addressed an issue where the RCP could incorrectly dequeue packets destined for the host, resulting in a parse error in the OTBR and unexpected termination.
1249492	The dbus FactoryReset command no longer causes the openthread border router to terminate.
1251926	When using the crash handler feature in a Host / RCP environment, the RCP was attempting to send the crash info to the Host too early after reboot, causing the info to be dropped prior to logging. This issue is addressed by introducing a new spinel property to the upstream openthread stack. The property allows the Host to request crash logs from the RCP once the Host is ready for them. See https://github.com/openthread/openthread/pull/10061 for more information on the new spinel property.
1251952	Fixed undefined reference to otInstanceResetToBootloader when building with ot_cert_libs and bootloader_interface.
1255595	Fix a type promotion issue when handling large values in otPlatAlarm APIs
1263222	Fixed coex metric "Average Request To Grant Time".
1277790	Fixed coex metric "Grant Immediate".
1287331	The ot-ble-dmp sample application is now only supported on parts with at least 768k of flash.

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.silabs.com/developers/thread> in the Tech Docs tab.

ID #	Description	Workaround
482915 495241 1295252	A known limitation with the UART driver can cause characters to be lost on CLI input or output. This can happen during particularly long critical sections that may disable interrupts, so it can be alleviated by repeating the CLI or waiting long enough for state changes.	No known workaround
815275	Ability to modify the Radio CCA Modes at compile-time using a configuration option in Simplicity Studio is currently not supported.	Use the SL_OPENTHREAD_RADIO_CCA_MODE configuration option defined in openthread-core-efr32-config.h header file included with your project.
1286531 1295725	CSL/SSSED instability.	Adjust settings of CSL related configuration macros.

5 Deprecated Items

Deprecated in release 2.5.0.0

None.

6 Removed Items

Removed in release 2.5.0.0

- Support for Series 0 and Series 1 parts
- Support for Proprietary Sub-GHz Support

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 2.5.0.0

OpenWRT alpha support has been added for zigbeed, OTBR and Z3Gateway applications. Zigbeed and OTBR are now provided in IPK package format as well. See *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol Co-Processor* for details.

7.2 Improvements

Changed in release 2.5.2.0

In Zigbeed, the `halCommonGetInt32uMillisecondTick()` tick API is now updated to use MONOTONIC clock so that it does not get affected by the NTP in a host system.

Changed in release 2.5.1.0

The `zb_ble_dmp_print_ble_connections()` API is defined in the `zigbee_ble_event_handler` component and referenced in the `zigbee_ble_dmp_cli` component. For applications that use the `zigbee_ble_dmp_cli` component, but NOT the `zigbee_ble_event_handler`, you will need to add an empty stub for this function in your `app.c` file as follows:

```
void zb_ble_dmp_print_ble_connections(void)
{
}
```

Changed in release 2.5.0.0

None.

7.3 Fixed Issues

Fixed in release 2.5.2.0

ID #	Description
1332330	Fixed an issue where a 15.4+BLE RCP operating in an environment with heavy network traffic could occasionally encounter a race condition that would leave it unable to send messages up to CPCd until rebooting the device. (Other ref: 1333150)
1337228	In Zigbeed the <code>halCommonGetInt32uMillisecondTick()</code> tick API is now updated to use MONOTONIC clock, so that it does not get affected by the NTP in a host system. (Other ref: 1346711)

Fixed in release 2.5.1.0

ID #	Description
1300848	An issue was fixed where Z3Gateway in OpenWRT environment couldn't start EZSP communication caused by mismatching termios control characters running on OpenWRT and other environments.

Fixed in release 2.5.0.0

ID #	Description
1231021	Fixed an issue to avoid an OTBR assert by recovering the RCP rather than passing unhandled transmit errors to the sub mac.
1242948	Removed spurious test asserts from zigbeed.
1244459	Fixed issue where MAC-retried Indirect transmissions via RCP can result in a source match table entry for child being removed despite messages pending.
1245988	Fixed an issue where Zigbeed did not restart when performing a Trust Center Backup and Restore Reset Node action.
1282264	Fixed issue that could have interrupted radio transmit operations by clearing the transmit fifo prematurely, causing underflow.
1288653	Zigbee/OT/BLE SOC app will now print connection info upon receiving the CLI command "plugin ble gap print-connections".
1292537	Zigbee/BLE NCP application is now properly showing up in Simplicity Studio UI.
1252365	Added Coexistence plugin back into Zigbee NCP/OpenThread RCP sample application.
1293853	Reduced RAM footprint for zigbee_ncp-ot_rcp-spi and zigbee_ncp-ot_rcp_uart on MG21.
1124140	Fixed an issue where setting SL_OPENTHREAD_RADIO_RX_BUFFER_COUNT to a value other than 1 caused z3-light_ot-ftd_soc to not send Zigbee beacons after the OpenThread network is up.

7.4 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.silabs.com/developers/simplicity-software-development-kit>.

ID #	Description	Workaround
937562	Bluetoothctl 'advertise on' command fails with rcp-uart-802154-blehci app on Raspberry Pi OS 11.	Use btmgmt app instead of bluetoothctl.
1074205	The CMP RCP does not support two networks on the same PAN id.	Use different PAN ids for each network. Support is planned in a future release.
1122723	In a busy environment, the CLI may become unresponsive in the z3-light_ot-ftd_soc app.	No known workaround.
1209958	The ZB/OT/BLE RCP on MG24 and MG21 can stop working after a few minutes when running all three protocols.	Will be addressed in a future release.
1221299	Mfglib RSSI readings differ between RCP and NCP.	Will be addressed in a future release.

7.5 Deprecated Items

None.

7.6 Removed Items**Removed in release 2.5.0.0**

None.

8 Using This Release

This release contains the following

- Silicon Labs OpenThread stack
- Silicon Labs OpenThread sample applications
- Silicon Labs OpenThread border router

For more information about the OpenThread SDK see [QSG170: Silicon Labs OpenThread QuickStart Guide](#). If you are new to Thread see [UG103.11: Thread Fundamentals](#).

8.1 Installation and Use

The OpenThread SDK is part of the Simplicity SDK, the suite of Silicon Labs SDKs. To quickly get started with OpenThread and the Simplicity SDK, start by installing [Simplicity Studio 5](#), which will set up your development environment and walk you through Simplicity SDK 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, Simplicity SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/simplicity_sdk for more information.

The GSDK default installation location has changed beginning with Simplicity Studio 5.3.

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

Documentation specific to the SDK version is installed with the SDK. API references and other information about this release are available on <https://docs.silabs.com/openthread/latest/>. Select your SDK version in the upper right.

8.2 OpenThread GitHub Repository

The Silicon Labs OpenThread SDK includes all changes from the OpenThread GitHub repo (<https://github.com/openthread/openthread>) up to and including commit **1fceb225b**. An enhanced version of the OpenThread repo can be found in the following Simplicity Studio 5 GSDK location:

<GSDK Installation Location>\util\third_party\openthread

8.3 OpenThread Border Router GitHub Repository

The Silicon Labs OpenThread SDK includes all changes from the OpenThread border router GitHub repo (<https://github.com/openthread/ot-br-posix>) up to and including commit **e56c02006**. An enhanced version of the OpenThread border router repo can be found in the following Simplicity Studio 5 GSDK location:

<GSDK Installation Location>\util\third_party\ot-br-posix

8.4 Using the Border Router

For ease of use, Silicon Labs recommends the use of a Docker container for your OpenThread border router. Refer to [AN1256: Using the Silicon Labs RCP with the OpenThread Border Router](#) for details on how to set up the correct version of OpenThread border router Docker container. It is available at <https://hub.docker.com/r/siliconlabsinc/openthread-border-router>.

If you are manually installing a border router, using the copies provided with the Silicon Labs OpenThread SDK, refer to [AN1256: Using the Silicon Labs RCP with the OpenThread Border Router](#) for more details.

Although updating the border router environment to a later GitHub version is supported on the OpenThread website, it may make the border router incompatible with the OpenThread RCP stack in the SDK.

8.5 NCP/RCP Support

The OpenThread NCP support is included with OpenThread SDK but any use of this support should be considered experimental. The OpenThread RCP is fully implemented and supported.

8.6 Security Information

Secure Vault Integration

When deployed to Secure Vault High devices, sensitive keys are protected using the Secure Vault Key Management functionality. The following table shows the protected keys and their storage protection characteristics.

Wrapped Key	Exportable / Non-Exportable	Notes
Thread Master Key	Exportable	Must be exportable to form the TLVs
PSKc	Exportable	Must be exportable to form the TLVs
Key Encryption Key	Exportable	Must be exportable to form the TLVs
MLE Key	Non-Exportable	
Temporary MLE Key	Non-Exportable	
MAC Previous Key	Non-Exportable	
MAC Current Key	Non-Exportable	
MAC Next Key	Non-Exportable	

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.

The following figure is an example:

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

☐ Audio and Radio	☐ Power over Ethernet
☐ Interface	☐ Sensors
☐ Isolation	☐ TV and Video
☐ Modems and DAAs	☐ Voice
☐ Microcontrollers	☐ Wireless
☐ 8-bit MCUs	☐ Bluetooth Classic
☒ 32-bit MCUs	☐ Bluetooth Low Energy
☐ Timing	☒ Proprietary
☐ Clocks	☐ Wi-Fi
☐ Buffers	☐ ZigBee and Thread
☐ Oscillators	☐ Z-Wave
☐ CDR and PHY	

8.7 Support

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

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

8.8 Thread Certification

This release has been qualified for Thread 1.3.0 with Thread Test Harness v60.0 (Member Release). For Thread Product certifications tied to this major release and associated patch releases (with no OpenThread stack updates), Silicon Labs recommends using the above TH version for qualification. Also included with this release is a set of OpenThread stack and PAL libraries that may be used for Thread certification by inheritance.

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.

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®, 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