



Silicon Labs OpenThread SDK 2.2.7.0 GA

Gecko SDK Suite 4.2

January 15, 2025

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 in order 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):

- 2.2.7.0 GA released on January 15, 2025
- 2.2.6.0 GA released on July 3, 2024
- 2.2.5.0 GA released on January 24, 2024
- 2.2.4.0 GA released on August 16, 2023 (support for EFR32xG21, Revision C and later plus multiprotocol updates)
- 2.2.3.0 GA released on May 3, 2023
- 2.2.2.0 GA released on March 8, 2023
- 2.2.1.0 GA released on February 1, 2023
- 2.2.0.0 GA released on December 14, 2022



KEY FEATURES

OpenThread

- SPI support for OpenThread RCP without CPC
- Thread 1.3.0 GA, and 1.3.0.1 support for OpenThread and Matter 1.0 - experimental
- Support for CPC on Android Host - experimental
- MGM240S SiP Module support
- MG24 Explorer kit support
- BRD2704A board support

Multiprotocol

- Dynamic Multiprotocol Bluetooth and multi-PAN 802.15.4 in RCP mode
- Dynamic Multiprotocol Bluetooth and Zigbee NCP - experimental
- Manufacturing Library (MfgLib) support for Concurrent Multiprotocol RCP
- Zigbee + OpenThread Concurrent Listening on MG24 parts - experimental

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 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

Contents

1	New Items	3
1.1	New Components	3
1.2	New Features	3
1.3	New Radio Board Support	3
2	Improvements	4
3	Fixed Issues	5
4	Known Issues in the Current Release	7
5	Deprecated Items	8
6	Removed Items	9
7	Multiprotocol Gateway and RCP	10
7.1	New Items	10
7.2	Improvements	10
7.3	Fixed Issues	10
7.4	Known Issues in the Current Release	12
7.5	Deprecated Items	12
7.6	Removed Items	12
8	Using This Release	13
8.1	Installation and Use	13
8.2	OpenThread GitHub Repository	13
8.3	OpenThread Border Router GitHub Repository	13
8.4	Using the Border Router	13
8.5	NCP/RCP Support	14
8.6	Security Information	14
8.7	Support	15

1 New Items

1.1 New Components

None

1.2 New Features

Added in release 2.2.2.0

- New configuration setting `SL_ENABLE_MULTI_RX_BUFFER_SUPPORT` to enable an experimental multi buffer-rx feature to address a bug with transmitting fragmented messages in a DMP scenario.

Added in release 2.2.1.0

- Sample application `ot-ble-dmp-no-buttons`. This new application can be built and run on boards that do not have button support.

Added in release 2.2.0.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- Thread 1.3.0 GA and 1.3.0.1 support for OpenThread and Matter 1.0 (experimental).
 - Our OpenThread sample apps are built with 1.3.0 and 1.3.0.1 features enabled by default.
- SPI support for OpenThread RCP without CPC
 - SPI is now supported for communication between a host and an RCP. In previous releases UART was the only protocol supported for this communication when not using CPC. Refer to [AN1256: Using the Silicon Labs RCP with the OpenThread Border Router](#) for additional information.
- Support for the OpenThread Border router with CPC on Android Host (experimental).
 - The OpenThread Border Router can now be used with CPC on an Android host. To build, download the Android NDK toolchain, define the environment variable "NDK" to point to the toolchain, and run the `script/cmake-build-android` script instead of `script/cmake-build`.

1.3 New Radio Board Support

Added in release 2.2.1.0

Support has been added for the following radio boards:

- BRD2704A - MGM240PB32VNA2

Added in release 2.2.0.0

Support has been added for the following radio boards:

- BRD4318A - MGM240SD22VNA2
- BRD2703A - EFR32MG24 Explorer Kit

2 Improvements

Changed in release 2.2.0.0

Starting with 22Q4 GA release, OpenThread defaults to a native implementation of NAT64. To prevent collision with previously installed NAT64 configuration, please remove or backup your tayga configuration file, usually located in `/etc/tayga.conf`. This step is required for NAT64 to work for OTBR, especially when run in containers that turn on host networking.

3 Fixed Issues

Fixed in release 2.2.6.0

ID #	Description
1243597	Removed extra ot-ble-dmp-no-buttons sample apps from demos folder.

Fixed in release 2.2.5.0

ID #	Description
1193597	OpenThread Radio PAL now maintains Max channel power table.

Fixed in release 2.2.3.0

ID #	Description
1126570	Addressed a memory leak associated with PSA keys which occurs when <code>otInstanceFinalise()</code> is called without power cycling.
1133240	Fixed a bug in setting link parameters in the meshcop forwarding layer.

Fixed in release 2.2.2.0

ID #	Description
1084368	Addressed a bug with fragmented messages in a DMP scenario, where a receive complete callback was not sent for all acknowledged received fragments. The fix requires enabling a new configuration setting <code>SL_ENABLE_MULTI_RX_BUFFER_SUPPORT</code>

Fixed in release 2.2.1.0

ID #	Description
1074144	To prevent a single child, which maybe offline, from taking many and potentially all entries in the source match table we now check for duplicates before adding a new entry.
1085732	Addressed compilation issue where return values from function calls were not being checked. Logic was added to check the return values from these function calls in <code>factory_diags.cpp</code> : <code>otPlatDiagTxStreamStop</code> , <code>otPlatDiagTxStreamTone</code> , <code>otPlatDiagTxStreamRandom</code> , <code>otPlatDiagTxStreamAddrMatch</code> , and <code>otPlatDiagTxStreamAutoAck</code> .
1085743	Fixed an issue with the logic that allows for building the posix executables with <code>multipan_rcp</code> support. The previous logic assumed that, if an argument followed the platform argument it must be <code>multipan_rcp</code> , but if something other than <code>multipan_rcp</code> was passed the following error occurred: *** ERROR: OpenThread CMake doesn't support platform"
1085753	Added new config item, <code>OPENTHREAD_SPINEL_CONFIG_RCP_TX_WAIT_TIME_SECS</code> , to specify the wait time for reception of the <code>TxDone</code> callback from RCP.
1092864	Created a new sample application, <code>ot-ble-dmp-no-buttons</code> , which can be built and run on boards that do not have button support.

Fixed in release 2.2.0.0

ID #	Description
829618	The sample apps no longer default to compile as a reference device.
830554	RAIL PA ramp time is no longer hardcoded to 10 and instead now references the compile time defined macro SL_RAIL_UTIL_PA_RAMP_TIME_US.
1015604	Issues with NetworkTimeSync have been resolved.
1017551	<p>The following configuration values are now set by default for all OpenThread sample applications. Please note that if you want different values for these parameters, they need to be overridden in your app's .slcp file.</p> <ul style="list-style-type: none"> • OPENTHREAD_CONFIG_BACKBONE_ROUTER_ENABLE=0 • OPENTHREAD_CONFIG_BORDER_ROUTER_ENABLE=0 • OPENTHREAD_CONFIG_COAP_API_ENABLE=0 • OPENTHREAD_CONFIG_COAP_OBSERVE_API_ENABLE=0 • OPENTHREAD_CONFIG_COAP_SECURE_API_ENABLE=0 • OPENTHREAD_CONFIG_COMMISSIONER_ENABLE=0 • OPENTHREAD_CONFIG_DHCP6_CLIENT_ENABLE=0 • OPENTHREAD_CONFIG_DHCP6_SERVER_ENABLE=0 • OPENTHREAD_CONFIG_DNSSD_SERVER_ENABLE=0 • OPENTHREAD_CONFIG_JOINER_ENABLE=0 • OPENTHREAD_CONFIG_MAC_FILTER_ENABLE=0 • OPENTHREAD_CONFIG_REFERENCE_DEVICE_ENABLE=0 • OPENTHREAD_CONFIG_SRP_SERVER_ENABLE=0 • OPENTHREAD_CONFIG_TMF_NETDATA_SERVICE_ENABLE=0 • OPENTHREAD_CONFIG_LOG_OUTPUT= OPENTHREAD_CONFIG_LOG_OUTPUT_PLATFORM_DEFINED
1019947	Added support for building RCP projects for efr32mg1b and efr32mg1v parts.
1021181	Error when using external heap and having message buffers use the heap allocator have been resolved. See https://github.com/openthread/openthread/pull/7933
1026506	Addressed a linker issue that resulted when selecting Thread version 1.1 in the stack configuration.
1030815	The OpenThread Border Router no longer displays an incorrect version string for otbr-agent (`sudo otbr-agent --version`) or for the POSIX stack (`sudo ot-ctl version`) when previous build artifacts were present in the build/ folder when re-installing the OpenThread Border Router.
1058102	Fixed issue preventing 'coexistence get-pta-option' CLI from working.
1067632	Increased the CPC restart timeout interval to 100 msec up to 30 seconds to address an issue that resulted when restarting too quickly.

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	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
754514	Double ping reply observed for OTBR ALOC address.	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 <code>SL_OPENTHREAD_RADIO_CCA_MODE</code> configuration option defined in <code>openthread-core-efr32-config.h</code> header file included with your project.
1023725	If an OTBR distributes a DUA prefix on a network and does not restore previous prefix configuration after a reboot, previously detached MTDs on the Thread network can hit an assert while re-attaching to the OTBR.	Restore previously configured prefixes on an OTBR during initialization if rebooted. The prefix information is not stored across reboots.
1041112	OTBR / EFR32 RCP can miss forwarding packets from a CSL child if it configures an alternate channel for CSL communication. Due to this issue, OTBRs based on GSDK 4.2.0.0 are not expected to pass Thread 1.2 certification unless the customer use cases demand a waiver to exclude all tests that require changing the primary channel.	Avoid configuring alternate CSL channels until this issue is addressed.
1064242	OpenThread prefix commands sometimes fail to add prefix for OTBR over CPC.	No known workaround
1079667	Thread device can no longer communicate after reporting transient out-of-buffers condition.	No known workaround

5 Deprecated Items

Deprecated in release 2.2.0.0

Using Tayga as the NAT64 service with the openthread border router is now deprecated in favor of OpenThread's native NAT64 service. Refer to <https://github.com/openthread/ot-br-posix/pull/1539> and <https://github.com/openthread/ot-br-posix/pull/1577> for additional information.

6 Removed Items

None

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 2.2.2.0

Zigbeed now loads the CREATOR_STACK_RESTORED_EUI64, if present, from the host tokens file, and uses it as the EUI64, overriding the EUI64 stored on the EFR32.

Added in release 2.2.1.0

Zigbeed now supports coex EZSP commands.

Added in release 2.2.0.0

Added Dynamic Multiprotocol BLE and Zigbee NCP project (zigbee_ncp-ble_ncp-xxx.slcp). Released as experimental quality.

Added 802.15.4 concurrent listening for EFR32MG24 CMP RCP. This is the ability to run Zigbee and OpenThread simultaneously on different channels using a single RCP (rcp-802154-xxx.slcp and rcp-802154-blehci-xxx.slcp). Released as experimental quality.

Added Zigbeed support for 32-bit x86 architecture.

Added support for BLE to de-init in multiprotocol use cases, freeing up memory resources for use by other protocol stacks.

The Stack API Trace now can be enabled for Zigbeed by setting the debug-level to 4 or 5 in the zigbeed.conf file.

Zigbeed stack version as well as build date and time are now printed in the logs.

7.2 Improvements

Changed in release 2.2.2.0

Reduced CPC Tx and Rx queue sizes to fit the Zigbee BLE DMP NCP onto the MG13 family.

Changed zigbee_ble_event_handler to print scan responses from legacy advertisements in DMPLight app.

The rcp-xxx-802154 and rcp-xxx-802154-blehci apps now use 192 µsec turnaround time for non-enhanced acks while still using 256 µsec turnaround time for enhanced acks required by CSL.

7.3 Fixed Issues

Fixed in release 2.2.6.0

ID #	Description
1244459	<p>When sending Zigbee data packets to sleepy children of an RCP-based device, the framePending state for that child on the RCP could be corrupted and forced False under these conditions:</p> <ul style="list-style-type: none"> • if the delivery to the child fails due to no MAC Ack received, and • additional data packets for that child are waiting in the MAC Indirect queue. <p>This corruption would leading to no further data being picked up by the sleepy end device until all those pending data have timed out in the host stack after the MAC Indirect Transmission Timeout (7680ms by default) has elapsed for each queued message.</p> <p>This issue has been fixed. Other ref. 1243036, 1294248</p>
1199706	<p>Fixed an issue where data polls from forgotten end device children weren't properly setting a pending frame on the RCP to queue a Leave and Rejoin command to the former child.</p> <p>Other ref. 1305071</p>

Fixed in release 2.2.5.0

ID #	Description
1188521	Fixed an RCP hang issue when running BLE Scan on with notification and OpenThread ping traffic.

Fixed in release 2.2.4.0

ID #	Description
1118077	In the CMP RCP, Spinel messages were being dropped under heavy traffic load due to CPC not keeping up with the incoming packets. Fixed this by bundling all Spinel messages ready to be sent over CPC into one payload on the RCP, and unbundling them on the host. This dramatically improves the efficiency of CPC so that it can keep up with the incoming radio traffic.
1113498, 1135805, 1139990, 1143344	Fixed multiple intermittent Zigbeed crashes and asserts that could be triggered when joining many Zigbee devices simultaneously to the CMP RCP.

Fixed in release 2.2.3.0

ID #	Description
1130226	Fixed issue in which the RCP would not recover if CPC became temporarily busy.
1129821	Fixed null pointer dereference in Zigbeed when receiving a packet if no buffers are available.

Fixed in release 2.2.1.0

ID #	Description
1036645	Solved a bug in BLE CPC NCP which prevented a client app from reconnecting after the first disconnection.
1068435	Fixed Green Power bidirectional commissioning timing issue. Certification test case GPP 5.4.1.23 passes.
1074593	Fixed issue in which Just-in-time (JIT) messages to sleepy end devices were not sent correctly by Zigbeed + RCP.
1076235	Fixed issue where ot-cli failed to run in the multiprotocol docker container.
1080517	Z3GatewayCPC now automatically handles a reset of the NCP (CPC secondary).
1085498	Fixed an issue where Zigbeed was not sending rejoin responses to sleepy end devices indirectly.
1090915	Fixed issue where multiple 0x38 errors appeared when attempting to either open a Zigbee endpoint on the Z3GatewayCPC OR to set EZSP parameters without resetting the CPC NCP.

Fixed in release 2.2.0.0

ID #	Description
828785	Fixed a bug in cpc-hci-bridge that caused an HCI packet to be dropped if BlueZ sent two at once.
834191	Improved the CPU utilization of the cpc-hci-bridge helper application.
1025713	Increased max length of Zigbeed device path to 4096.
1036622	Fixed a problem using cmake to build ot-cli using the multi-PAN RCP.
1040127	CPC security was failing to initialize for the rcp-uart-802154 and rcp-spi-802154 projects on MG13 and MG14 series parts. To work around this issue, mbedtls_entropy_adc has been added as entropy source for these parts. That might prevent the ADC from being used in combination with CPC security.
1066422	Fixed an intermittent buffer leak in Zigbeed.
1068429	Fixed a race condition that could cause the CMP RCP to assert.
1068435	Added capability on the RCP node to check and buffer a single bidirectional Green Power data frame and send it out upon rx offset timeout.
1068942	Fixed a leak in the RCP source match table that could prevent Zigbee devices from joining.

ID #	Description
1074172	Fixed sending leave request from Zigbeed when receiving a poll from a non-child.
1074290	Stopped Zigbeed from processing un-acked polls.
1079903	Fixed a bug in the CMP RCP that could cause SPINEL messages to be dispatched incorrectly, resulting in Zigbeed and OTBR crashing or exiting.

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/gecko-software-development-kit>.

ID #	Description	Workaround
811732	Custom token support is not available when using Zigbeed.	Support is planned in a future release.
937562	Bluetoothctl 'advertise on' command fails with rcp-uart-802154-blehci app on Raspberry Pi OS 11.	Use btmgmt app instead of bluetoothctl.
1031607	The rcp-uart-802154.slcpc project is running low on RAM on an MG1 part. Adding components may reduce the heap size below what is needed to support ECDH binding in CPC.	A workaround is to disable CPC security via the SL_CPC_SECURITY_ENABLED configuration.
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.

7.5 Deprecated Items

None

7.6 Removed Items

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 Gecko SDK (GSDK), the suite of Silicon Labs SDKs. To quickly get started with OpenThread and the GSDK, start by installing [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 installation location has changed beginning with Simplicity Studio 5.3.

- 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. API references and other information about this release are available on <https://docs.silabs.com/openthread/2.1/>.

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 **91fa1f455**. 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 **d9103922a**. 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.

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	

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>.

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