Wi-SUN SDK 2.2.0.0 GA
Simplicity SDK Suite 2024.6.2
September 18, 2024

Wireless Smart Ubiquitous Network (Wi-SUN) is the leading IPv6 sub-GHz mesh technology for smart city and smart utility applications. Wi-SUN brings Smart Ubiquitous Networks to service providers, utilities, municipalities/local government, and other enterprises, by enabling interoperable, multi-service, and secure wireless mesh networks. Wi-SUN can be used for large-scale outdoor IoT wireless communication networks in a wide range of applications covering both line-powered and battery-powered nodes.

Silicon Labs' Wi-SUN hardware is certified by the Wi-SUN Alliance, a global industry association devoted to seamless LPWAN connectivity. Wi-SUN builds upon open standard internet protocols (IP) and APIs, enabling developers to extend existing infrastructure platforms to add new capabilities. Built to scale with long-range capabilities, high-data throughput and IPv6 support, Wi-SUN simplifies wireless infrastructure for industrial applications and the evolution of smart cities.

These release notes cover SDK versions:

2.2.0.0 released September 18, 2024.
2.1.0.0 released July 24, 2024.
2.0.0.0 released June 5, 2024.

KEY FEATURES

Wi-SUN Stack

- Dropped the legacy socket API
- Stability improvements
- Removed support for Series 0/1

Wi-SUN Applications

- Underlying platform changes only

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/wi-sun-protocol-stack. 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 Wi-SUN SDK, 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.

Contents

1 Wi-SUN Stack

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 Items

Added in release 2.2.0.0

- Added sl_wisun_set_leaf() API to make an FFN act as RPL leaf.
- Replaced wisun_test_certificates component with wisun_keychain and wisun_silabs_certificates.

Added in release 2.1.0.0

- Added sl_wisun_set_tx_power_ddbm() API to set the Tx power with a 0.1 dB resolution.
- Deprecated sl_wisun_set_tx_power() that is replaced by sl_wisun_set_tx_power_ddbm(). It is still possible to call that API, but it will be removed in a future release.

Added in release 2.0.0.0

- Dropped the support of our previous socket API. Only the Posix-like socket API remains.
- Split the stack into smaller software modules, offering more granularity to the selection of the feature set selected and reducing the flash and RAM footprints. A device can now either act as an FFN when only the FFN-support component is installed, an LFN when only the LFN-support component is installed, or can decide which profile will be used thanks to an API call during the initialization phase if both are installed. As a direct consequence, new libraries with new filenames are released.
- Added an example of select() implementation. It only works with socket file descriptors but is distributed in source and could be enriched to support new file descriptor types.
- Added support for sendmsg(), recvmsg(), getpeername() and getsockname(). Extended the list of supported socket options.

1.2 Improvements

Changed in release 2.2.0.0

- Improved ETX mean value computation and reduced the weight given to the first measurement.

Changed in release 2.1.0.0

- Improved network stability by reducing the number of inappropriate disconnections.

Changed in release 2.0.0.0

- Added a random delay before starting an LFN join sequence. A significant number of over-the-air collisions occurred when starting multiple LFN at the exact same time.

1.3 Fixed Issues

Fixed in release 2.2.0.0

ID #	Description
1333646	Fixed parent selection issue causing frequent parent switching and infinite rank advertising.

ID #	Description
1320129	Fixed FAN 1.0 EDFE handling in RCP.
1332417	Fixed invalid error code on LFN queue allocation failure.
1333119	Fixed usage of sl_wisun_set_trusted_certificate() in example applications.
1336806	Fixed wrong join state 4 transition when cleaning the neighbor table.
1325595	Fixed an issue causing a frame to stay blocked in the MAC indefinitely.

Fixed in release 2.1.0.0

ID #	Description
1301513	Fixed state transition that could skip the joins state 1 during a reconnection.
1324900	Fixed SO_NONBLOCK socket option not being handled correctly.
1310166	Fixed an issue causing the radio to stay in IDLE state until the next TX after an RX timeout. This was impacting a workaround that was set up to avoid staying blocked indefinitely in RX.
1307422	Fixed an issue where MPL parameter modifications are not taken into account on second and further connections.
1322100	Ignored previous RPL parents' information when resetting a router with join state 1 acceleration disabled (discovery.allow_skip=0).
1309738	Join metric IE (JM-IE) were not read from ULAD multicast frame.
1322546	Increased timeout for async frames and ensured async fragmentation is reset after a timeout on an abort.
1305036	Accepted downward frame containing an empty source routing header (SRH). Empty SRH can be sent when trying to reach a direct child. Silicon Labs stack is not sending empty SRH. It fixes an interoperability issue with competitors' border routers.
1290741	Fixed RPL global repair. It was mainly visible in certification tests.

Fixed in release 2.0.0.0

ID #	Description
1290487	Fixed an issue causing the radio to be stuck in RX. The cause has been identified as a race in the lower layers of the radio driver.
1287317	Fixed an invalid memory access in the timer and event lists.
1285497	Fixed multiple invalid counter and timer updates during state transitions and RPL parent updates. They were causing abusive router disconnections.
1275243	Fixed an invalid use of a link-local as a source address in the DAO. On rare occasions, routers were using a link-local source address in the outer IPv6 header.
1272406	Fixed the condition upon which the MAC reset the CCA failures counter. The CCA failures counter is increased after each CCA busy event. Once the counter reaches a value of 8, it increases the retry counter by one. The CCA failures counter was only reset after a successful transmission. It is now reset after every transmission.
1258384	Fixed an invalid PAN ID filter configuration that was preventing routers from connecting to a new border router after a PAN timeout.

1.4 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
1119464	**Packets sent when FSK FEC is enabled can infringe ARIB regulation.**	**Until this problem has been addressed correctly, the stack will refuse to start if both ARIB enforcement and FEC are enabled.**

ID #	Description	Workaround
1067978	Packets sent using a PHY with a bandwidth larger than the base PHY's bandwidth can infringe ARIB regulation.	Limit the communications when using mode switch with ARIB enforcement enabled.
1176014	FG25 asserts with a RAIL_ASSERT_FAILED_RTCC_SYNC_STALE_DATA error code when entering EM2.	Do not allow the power manager to go to EM2 by adding a requirement on EM1 in the application.
1328491	Multicast is unreliable when operating under heavy multicast load.	Limit the amount of simultaneous multicast traffic.

1.5 Deprecated Items

None

1.6 Removed Items

None

2 Wi-SUN Applications

2.1 New Items

Added in release 2.1.0.0

- Wi-SUN SoC CLI
  - Added new ping settings, including the packet length, the packet sequence number, and the interval between two ping requests.
- OTA DFU remote control over CoAP for all settings of sl_wisun_ota_dfu_config.h (file name, URI path, …)
- libcoap-3 support

2.2 Fixed Issues

Fixed in release 2.2.0.0

ID #	Description
1341012	Fixed memory leak in Wi-SUN SoC CoAP Meter.
1341040	Fixed potential memory leak in CoAP resource discovery module.
1338959	Fixed TFTP Client stuck in blocked state if the communication crashed.
1332965	Fixed OTA DFU “stop” command can't stop the file transfer.
1347781	Fixed resource discovery request initialization in CoAP Meter.

Fixed in release 2.1.0.0

ID #	Description
1289218	Fixed Wi-SUN - SoC (CoAP) Meter packet fragmentation issues.
1283362	Fixed iPerf FINACK packet parser issue.
1296544	Fixed inet_pton return value handling.
1296797	Fixed broken json in CoAP packet printer.
1294628	Fixed remote CoAP CLI buffer handling for Wi-SUN - SoC Network Measurement. Cleaning previous buffer content.
1276803	Wi-SUN - SoC Network Measurement : “iperf get json” returns with “Invalid resolved buffer” until the first test.
1304230	In certain cases, proper packing of data structures is only ensured when using IAR compiler. However, this discrepancy can lead to issues when combining binaries built with both GCC and IAR in the network. The issue relates to all applications except: • Wi-SUN - CLI example • Wi-SUN - LFN CLI example • Wi-SUN - RCP

2.3 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
1327376	The RCP UART driver is unstable under certain conditions that we have not able to identify for the moment. Those instabilities are causing CRC errors that the RCP or wsbrd is not able to recover from.	Use CPC instead of using the RCP light weight interface.

ID #	Description	Workaround
1067236	The border router RCP SPI interface is unstable when used with a throughput higher than 1 Mbytes/s	The use of border router RCP SPI interface is not recommended for the time being.
	Simplicity Studio – Network Analyzer: Wi-SUN Encrypted Packets are not supported Undecoded frames (only after Ack) according to PTI issues on Series 2	

2.4 Deprecated Items

Four applications will be phased out, and instead Wi-SUN – SoC Socket Application will be introduced combining together the four functionalities. This transition will bring in simplicity and also enhance overall flexibility.

- Wi-SUN - SoC UDP Client
- Wi-SUN - SoC UDP Server
- Wi-SUN - SoC TCP Client
- Wi-SUN - SoC TCP Server

The following applications are deprecated, but their “CoAP versions” will be kept. The ability to update to the “non-CoAP” version will remain an option by uninstalling the CoAP component from the examples.

- Wi-SUN – SoC Meter (Wi-SUN - SoC CoAP Meter is not depricated)
- Wi-SUN – SoC Collector (Wi-SUN - SoC CoAP Collector is not deprecated)

2.5 Removed Items

None.

3 Using This Release

This release contains the following:

Wi-SUN stack library

Wi-SUN sample applications

Wi-SUN border router pre-compiled demos

Documentation

If you are a first time user, see https://docs.silabs.com/wisun/latest/wisun-getting-started-overview/

3.1 Installation and Use

The Wi-SUN SDK is provided as part of the Simplicity SDK, the suite of Silicon Labs SDKs. To quickly get started with the Simplicity SDK, install 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.

Simplicity Studio installs the Simplicity SDK by default in:

- (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.

3.2 Security Information

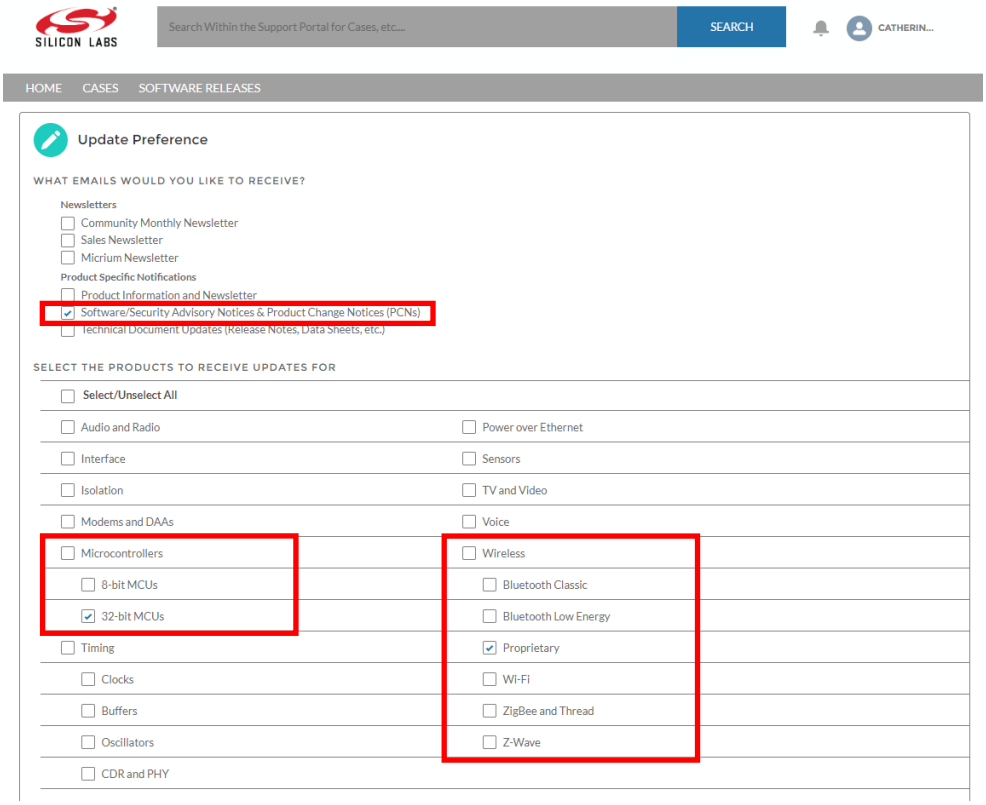
Secure Vault Integration

This version of the stack does not integrate Secure Vault Key Management.

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:



3.3 Support

Development Kit customers are eligible for training and technical 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

SILICON LABS

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

www.silabs.com