Z-Wave and Z-Wave Long Range 700/800 SDK 7.18.8.0 GA

Gecko SDK Suite 4.1
September 13, 2023

Z-Wave and Z-Wave Long Range 700/800 are designed to meet the demands of the future smart home, where increasing needs for more sensors and battery-operated devices require both long range and low power. Context-aware environments are the next evolution in the smart home market, and they require technologies that have been optimized specifically for these applications.

100% Interoperable: Every product in the Z-Wave ecosystem works with every other product, regardless of type, brand, manufacturer or version. No other smart home/IoT protocol can make this claim.

Best-In-Class Security: Z-Wave's Security 2 (S2) framework provides end-to-end encryption and the most advanced security for smart home devices and controllers. Homes with S2 Z-Wave devices are virtually un-hackable.

SmartStart Easy Installation: SmartStart radically simplifies the installation of smart devices by using QR code scans for uniform, trouble-free setup. Devices and systems can be pre-configured dramatically easing deployments.

Backwards-Compatible: Z-Wave certification mandates backward-compatibility. The first Z-Wave devices on the market, more than ten years old, still perform as intended in networks with the latest Z-Wave technologies.

The Z-Wave 700/800 SDK v7.18.8 GA release is intended for development of Z-Wave-certifiable, 700/800-based products entering volume production. This release contains underlying platform changes only having no impact on the Z-Wave functionality. For more information about the certification status of Z-Wave and Z-Wave Long Range 700/800 SDK v7.18.8 GA, see section 10 Product Life Cycle and Certification.

These release notes cover SDK version(s):
7.18.8.0 GA released, September 13, 2023 (underlying platform changes only)
7.18.7.0 OSR released August 28, 2023
7.18.6.0 GA released April 5, 2023
7.18.5.0 OSR released March 15, 2023 (early access part support only)
7.18.4.0 GA released January 18, 2023 (early access part support only)
7.18.3.0 GA released October 19, 2022 (early access part support only)
7.18.2.0 GA released September 28, 2022
7.18.1.0 GA released August 17, 2022
7.18.0.0 Pre-Certified GA released June 8, 2022

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 Z-Wave 700/800 SDK, see section 9 Using This Release.

KEY FEATURES

- Key fob controller application
- New Z-Wave 800 SDK containing preprogrammed radio boards for quick IOT demos and Z-Wave technology evaluations
- Z-Wave 800 S2 protocol uses secure vault for cryptographic key storage and hardware acceleration
- White paper about large network performance
- Serial API Controller and End Device-source code available enabling customization of I/O etc.
- Z-Wave Long Range supports now both 250ms and 1000ms wakeup beams
- Improved SmartStart inclusion time on Z-Wave Long Range
- Streamlined hardware dependencies in Z-Wave applications
- Z-Wave Region stored in MFG token
- Support of WSTK v2 mainboard BRD4002A
- GCC Compiler 10.3

Contents

1 Supported Radio Boards

This section describes the radio boards supported by the certified and pre-certified applications for the 700 and 800 Series, respectively.

Table 1-1. Supported Radio Boards.

Series	Radio Board	Description	Applications
800	BRD2603B	ZGM230SA: ZW-LR, SiP, 14 dBm & Secure Vault Mid.	Applications using BRD2603B SDK
800	BRD4204A	EFR32ZG23: ZW-LR, SoC & 14 dBm	Serial API using BRD4002A
800	BRD4204B	EFR32ZG23: ZW-LR, SoC & 14 dBm	Serial API using BRD4002A
800	BRD4204C	EFR32ZG23: ZW-LR, SoC, 14 dBm & Secure Vault High	Serial API using BRD4002A
800	BRD4204D	EFR32ZG23: ZW-LR, SoC, 14 dBm, Secure Vault High & external 32kHz crystal mounted	Serial API using BRD4002A
800	BRD4205A	ZGM230SA: ZW-LR, SiP, 14 dBm & Secure Vault Mid	Applications using BRD4002A/BRD8029A
800	BRD4205B	ZGM230SB: ZW-LR, SiP, 14 dBm & Secure Vault High.	Applications using BRD4002A/BRD8029A
800	BRD4210A	ZGM230S: ZW-LR, SoC, 14 dBm & Secure Vault High	Applications using BRD4002A/BRD8029A
700	BRD4200A	ZGM130S: SiP & 14 dBm	Applications using BRD4002A/BRD8029A
700	BRD4201A	EFR32ZG14: SoC & 14 dBm	Serial API using BRD4002A
700	BRD4202A	ZGM130S: SiP, 14 dBm & no SAW filters	Applications using BRD4002A/BRD8029A
700	BRD4206A	EFR32ZG14: ZW-LR, SoC & 14 dBm	Serial API using BRD4002A
700	BRD4207A	ZGM130S: ZW-LR, SiP & 14 dBm	Applications using BRD4002A/BRD8029A
700	BRD4208A	EFR32ZG14: ZW-LR, SoC & 20 dBm	Serial API using BRD4002A
700	BRD4209A	ZGM130S: ZW-LR, SoC & 20 dBm	Applications using BRD4002A/BRD8029A

The applications in the above table need a radio board in combination with BRD4002A – Wireless Starter Kit Mainboard (WSTK) and BRD8029A – Buttons and LEDs Expansion Board. Notice that BRD4002A is compatible with the old BRD4001A mainboard that is going to be deprecated. The Serial APIs in the above table only need a radio board and a BRD4002A – Wireless Starter Kit Mainboard (WSTK). Refer to *INS14278: How to Use Certified Apps* and *INS14816: How to Use Pre-Certified Apps*, for details.

ZW-LR indicates that the radio board supports both Z-Wave and Z-Wave Long Range. 14/20 dBm indicates the transmit power of the radio board. Secure Vault is an industry-leading suite of state-of-the-art security features that address escalating Internet of Things (IoT) threats.

Table 1-2. Radio Boards versus OPNs.

Series	Radio Board	OPN Description
800	BRD2603B	ZGM230SA27HGN3
800	BRD4204A	EFR32ZG23A010F512GM48
800	BRD4204B	EFR32ZG23A010F512GM48
800	BRD4204C	EFR32ZG23B010F512IM48
800	BRD4204D	EFR32ZG23B010F512IM48
800	BRD4205A	ZGM230SA27HNN0
800	BRD4205B	ZGM230SB27HGN2
800	BRD4210A	EFR32ZG23B020F512IM48
700	BRD4200A	ZGM130S037HGN2
700	BRD4201A	EFR32ZG14P231F256GM32
700	BRD4202A	ZGM130S037HGN2
700	BRD4206A	EFR32ZG14P231F256GM32
700	BRD4207A	ZGM130S037HGN2
700	BRD4208A	EFR32ZG14P731F256GM32
700	BRD4209A	EFR32ZG13P531F512GM48

The table above shows the Radio Boards and OPN relation. This table can be used to clarify the compatibility of the prebuilt binaries offered in the GSDK. The prebuilt binaries are built targeting boards and not OPNs. More OPNs are available than the ones listed above. For those OPNs the prebuilt binaries will not work. The desired application must be built targeting the specific OPN instead.

2   Z-Wave Protocol

Be aware that 800 products based on SDK v7.17.x do not support upgrade of Secure Element firmware over the air (OTA). However, a migration path exists to upgrade both main bootloader and Secure Element firmware to enable support of this feature. See *INS14895: Instruction for How to Use Tiny App* regarding the upgrade path. The 800-based SDK v7.18.x supports upgrade of Secure Element firmware over the air (OTA).

2.1   New Items

The Gecko Software Development Kit version 4.1 introduces a new underlying platform architecture based on components. The Z-Wave SDK 7.18.x now uses this component structure, and therefore the structure and build method of Z-Wave applications has changed compared to previous releases of Z-Wave SDK.

Release 7.18.1 GA contains the following new features:

- The NVM3 file system is reduced by 8 kB to free up space for application code and it only applies to 800-based radio boards.
- Fragmented beaming for Long Range and 3-channels (JP/KR region) now supports both 250 ms and 1000 ms Wake Up intervals.

Notice that the 8 kB reduction of the Z-Wave protocol NVM3 file system have an impact when making OTA firmware update on 800-based applications deployed on version 7.17.2 and earlier. To make an OTA firmware update from 7.17.2 to 7.18.1/2 requires that 7.18.1/2 is modified to keep the same NVM3 protocol size as 7.17.2. This can be configured by the define NVM3_DEFAULT_NVM_SIZE when building 7.18.1/2.

2.2   Improvements

None

2.3   Fixed Issues

Fixed in release 7.18.7 OSR

ID #	Description
1126896	Fixed potential communication issues.
1112282	Enhanced frame verifications.

Fixed in release 7.18.5 OSR GA

ID #	Description
1108515	End nodes could unintentionally reset during power down if the power down process is interrupted by network traffic.
1108516	Inclusion can occasionally fail when a lot of FLiRS devices are present in the network. In networks with 20+ FLiRS nodes it is recommended to set the ADD_NODE_OPTION_NO_FL_SEARCH when starting add mode on a controller. When inclusion is completed successfully a call to FUNC_ID_ZW_REQUEST_NODE_NEIGHBOR_UPDATE should be made to ensure that the routing table is updated correctly for the newly included node.
1108513	A TxQueue buffer overflow can in rare cases get the radio into a permanent receive state that can only be exited by transmitting a frame or forcing the radio to idle mode.

Fixed in release 7.18.4 GA

ID #	Description
1089436	When performing OTA firmware update to 7.18.3, do not re-generate qrcode for errors in qrcode-ext fields because the public and private keys are still valid.

Fixed in release 7.18.2 GA

ID #	Description
1029436	When objects exist in NVM3 the function zpal_nvm_enum_objects() will always return 1 erroneously. It must return the number of objects within the given borders.

ID #	Description
856681	700 Series FLiRS end device rarely could unintentionally reset. Can happen when the end device wakes up from EM2, starts a frame transmit, sleeps in EM1 (while transmission is in progress) and then wakes up again, by some event or by either a hardware or timer timeout, before transmission has finished.

Fixed in release 7.18.1 GA

ID #	Description
825895	800 series-based device cannot add a 100 series-based device.
830343	FUNC_ID_NVM_BACKUP_RESTORE is used by Z/IP Gateway in this order; OPEN, READ, and CLOSE. However, 800 returns no response on CLOSE while 700 works fine.
852477	It was not possible to configure I/O ports for UART on SerialAPI projects.
849375	Information about bootloader memory slot configuration missing from Z-Wave documentation.
845284	800 series protocol NVM file system reduced in size from 40k to 32k in end devices.
857518	700 series SensorPIR and Multilevel Sensor would sometimes fail SmartStart inclusion.
855000	Tx Power level can now set the MAX TX POWER to values higher than 12.7 dBm. The bug applies to Classic Z-Wave for 2/3-ch regions.
853880	SerielAPI command ZW_REMOVE_NODE_ID_FROM_NETWORK did not work.
843209	NOP power frame could use a reserved value for Tx power.
843208	Controllers would not send a Node Information frame when entering classic learn mode.
830343	SerialAPI FUNC_ID_NVM_BACKUP_RESTORE did not give a response on CLOSE operation for 800 series.

Fixed in release 7.18.0 Pre-Certified GA

ID #	Description
822972	Z-Wave Controller library is not available for both the ZG23A (BRD4204A and BRD4204B) and ZG23B (BRD4204C and BDR4204D).
833179	When excluding a Z-Wave Long Range device from the network the callback for FUNC_ID_ZW_REMOVE_NODE_FROM_NETWORK with REMOVE_NODE_STATUS_DONE – the nodeID is 0000 rather than the expected nodeID of the removed device. Additionally, note that FUNC_ID_SERIAL_API_GET_LR_NODES still has the node present. This only happens if another Node Info Frame (NIF) was received between inclusion and exclusion.
817671	Z-Wave 700/800 Listen Before Talk (LBT) threshold work for the default setting. However, it is not possible to change the threshold setting to other values.
738700	Controller cannot handle incoming notification command from a device during removal of a device.
839495	A node that is waiting for a routed_Ack frame will ignore it, if it the routed_ack frame was received before the silent_ack from the first hop if the routed frame. The result is additional transmitted frames creating an unnecessary overhead.
746113	Packet Trace Interface (PTI) is enabled on 800. This peripheral collects all received and transmitted packets and some meta data directly from the sequencer, thereby providing valuable information to help debug complex problems in a mesh network.
848606	Network Wide Exclusion (NWE) through repeaters doesn't work.
819943	Difference between 700 and 800 SmartStart timeout. Timeout expires about 4 seconds later on 800 compared to 700 that takes around 245 seconds.

2.4   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 Silicon Labs Release Notes page

ID #	Description	Workaround
1018947	SerialAPI: SERIAL_API_SETUP_CMD_MAX_LR_TX_PWR_GET does not reflect actual tx power if the power is set to an illegal value	None
753756	Network Wide Inclusion (NWI) of 500-based apps doesn't work through 700/800 repeaters.	NWI works at second attempt.
355095	In small networks Assign Return Routes will only generate direct range or one-hop routes even though multi-hop routes are possible.	None
361273	Transport Service is used when it is necessary to split a frame in two parts due to size. However, Transport Service does not forward RSSI information from the lower layers but only routing information. The RSSI value is the difference between LWR RSSI and background RSSI. As a consequence it is not possible to use RSSI for large frames handled by Transport Service in a network health calculation.	None

2.5   Deprecated Items

None

2.6   Removed Items

None

3   Z-Wave Plus V2 Application Framework

3.1   New Items

None

3.2   Improvements

For a detailed description of application development using the Z-Wave Plus V2 Framework, refer to *INS14259: Z-Wave Plus V2 Application Framework SDK7.*

A porting guide is also available for customers who want to migrate 800 hardware. The guide contains a detailed example of how to port a non-component/700-based Switch On/Off App (7.16.3) to a component/800-based Switch On/Off App (7.17.0). See *APL14836: Application Note for Porting Z-Wave Appl. SW from 700 to 800 hardware.*

Enabled support for excluding some mandatory command classes when building an application that is small enough for updating Secure Engine.

All direct hardware dependencies were removed from apps and ZAF. Use of application-related hardware is located in different hardware specific source files, i.e. <app>_hw.c. APIs for NVM, Bootloader, power management, etc. are now covered by the Z-Wave Platform Abstraction Layer (ZPAL). Applications can still use Silicon labs platform APIs if desired. The migration to ZPAL was done to enable future support for building the Z-Wave apps on more than one platform.

3.3   Fixed Issues

Fixed in release 10.18.1 Pre-Certified GA

ID #	Description
758507	The NVM3 file storing the result of the OTA upgrade was not done before reboot. The file migration would therefore fail in case it needed to be converted from the old format to the new FW format.
855489	CC Notification: Notification report was sent when notification state was set to OFF

3.4   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 the Silicon Labs Release Notes page

ID #	Description	Workaround
369430	All S2 multicast frames are sent using verified delivery S2_TXOPTION_VERIFY_DELIVERY whether or not a response is expected.	Change source code depending on the frame sent.
473723	True status doesn't report correctly if there are multiple instances like colors (in Color Switch CC), endpoints, etc.	None

3.5   Deprecated Items

None

3.6   Removed Items

None

4 Certified Applications

The certified applications based on v7.x.1+ will be formally certified by a certification house. However, the first release (v7.x.0) will only contain pre-certified applications based on a certification test using CTT v3. Refer to *INS14278: How to Use Certified Apps* for details.

4.1 Door Lock Key Pad

4.1.1 New Items

None

4.1.2 Improvements

None

4.1.3 Fixed Issues

None

4.1.4 Known Issues in the Current Release

None

4.1.5 Deprecated Items

None

4.1.6 Removed Items

None

4.2 LED Bulb

4.2.1 New Items

None

4.2.2 Improvements

None

4.2.3 Fixed Issues

None

4.2.4 Known Issues in the Current Release

None

4.2.5 Deprecated Items

None

4.2.6 Removed Items

None

4.3 Power Strip

4.3.1 New Items

None

4.3.2 Improvements

None

4.3.3 Fixed Issues

None

4.3.4 Known Issues in the Current Release

None

4.3.5 Deprecated Items

None

4.3.6 Removed Items

None

4.4 Sensor PIR

4.4.1 New Items

None

4.4.2 Improvements

None

4.4.3 Fixed Issues

Fixed in release 10.18.1 GA

ID #	Description
844498	When the Wake Up Interval times out, SensorPIR app in SDK 7.17.2 for both 700s and 800s reports the reset reason as 4 (ERESETREASON_EM4_EXT_INT), which should be 1 (ERESETREASON_EM4_WUT)

4.4.4 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 the Silicon Labs Release Notes page

ID #	Description	Workaround
758906	Wakeup current has increased on both 700 and 800 SoCs. Caused by an unintentional IR calibration performed at start-up.	None

4.4.5 Deprecated Items

None

4.4.6 Removed Items

None

4.5   Switch On/Off

4.5.1 New Items

None

4.5.2 Improvements

None

4.5.3 Fixed Issues

None

4.5.4 Known Issues in the Current Release

None

4.5.5 Deprecated Items

None

4.5.6 Removed Items

None

4.6 Wall Controller

4.6.1 New Items

None

4.6.2 Improvements

None

4.6.3 Fixed Issues

None

4.6.4 Known Issues in the Current Release

None

4.6.5 Deprecated Items

None

4.6.6 Removed Items

None

5 Pre-Certified Applications

The pre-certified applications will not be formally certified but certification tests have been performed based on CTT v3. Refer to *INS14816: How to Use Pre-Certified Apps* for details.

5.1 Multilevel Sensor

5.1.1 New Items

None

5.1.2 Improvements

None

5.1.3 Fixed Issues

Fixed in release 10.18.1 GA

ID #	Description
844930	Multilevel Sensor should send Basic Set to the nodes in Association Group 2 when button S2 is pressed for at least 1 second.

5.1.4 Known Issues in the Current Release

None

5.1.5 Deprecated Items

None

5.1.6 Removed Items

None

5.2 Key Fob Controller

This application is new as of 7.18.x. It offers an example of how to create a key fob that is able to include and control other Z-Wave nodes. One use case could be a kit consisting of a key fob and a battery-driven shade. As the key fob can add more devices to its network, it opens the possibility for adding additional shades.

5.2.1 New Items

None

5.2.2 Improvements

None

5.2.3 Fixed Issues

Fixed in release 10.18.0 Pre-Certified GA

ID #	Description
820623	The Portable Controller EM2 sleep current shows that it consumes more than 40 µA. Expects approximately 2 µA.
844319	Keyfob as a secondary controller cannot perform security probing of an associated device.

5.2.4 Known Issues in the Current Release

None

5.2.5 Deprecated Items

None

5.2.6 Removed Items

None

6 Serial API Applications

Beginning with version 7.16, when backing up and restoring a SerialAPI via the FUNC_ID_NVM_BACKUP_RESTORE, the SerialAPI will automatically upgrade the protocol non-volatile memory (NVM) to the latest version. Any backup made of a 7.16 or later SerialAPI can be restored to its original version or to a later version of the SerialAPI without any manual upgrade of the protocol NVM being necessary.

The serial interface is unchanged in version 8.

As of SDK version 7.18.x, Serial API is available as source code as well as binary. This opens the possibility for building customized versions of Serial API with different pin configuration or additional hardware utilization. A use case might be to use SPI instead of UART for serial communication.

No application using Serial API End Device is available in the GSDK.

6.1 Serial API Controller

Added in release 10.18.0 GA

Host application on Unify uses the Serial API Controller application.

6.1.1 New Items

None

6.1.2 Improvements

None

6.1.3 Fixed Issues

Fixed in release 10.18.2 GA

ID #	Description
1024915	SerialAPI implementation of the SetTxPower16Bit can now set the MAX TX POWER to values higher than 12.7 dBm. The bug applies to Classic Z-Wave for 2/3-ch regions.

6.1.4 Known Issues in the Current Release

None

6.1.5 Deprecated Items

None

6.1.6 Removed Items

None

7 Important Changes

7.1 Power Down Debug

Debugging in EM2 power down mode is enabled by default, as it was in the previous SDK version. To disable the EM2 debug capablility in Simplicity Studio go to application.slcp **SOFTWARE COMPONENTS** -> **Services** -> **Device Initialization** and open the Component Editor for **Device Init: EMU**. Then turn off the switch at **Allow debugger to remain connected in EM2**.

The macro APP_POWERDOWNDEBUG in config_app.h that was previously used to select if EM2 debug would be enabled/disabled has been deprecated.

7.2 ZAF_Init() and ZAF_Reset()

ZAF_Init() will initialize command classes that have registered an init function using REGISTER_CC_V4(). ZAF_Reset() will reset command classes that have registered a reset function using REGISTER_CC_V4(). See CC_PowerLevel.c for an example of this.

7.3 Command Class Z-Wave Plus Info

Command Class Z-Wave Plus Info was refactored to decouple it better from the application. This means that the CC_ZWavePlusInfo.h does not exist anymore. On the application side the static const SCCZWavePlusInfo CCZWavePlusInfo is no longer needed because the CC_ZWavePlusInfo_Init(&CCZWavePlusInfo); function is removed. All configuration is done under the hood by the command class and the values are still extracted from the config_app.h.

7.4 Command Class Version

Command Class Version was refactored to support excluding it from a build and thus creating a smaller application that can be used for updating Secure Element. This resulted in the following functions no longer being required in the application:

CC_Version_getNumberOfFirmwareTargets_handler()

This function was replaced by a configuration parameter for the ZAF component that can be set in Simplicity Studio.

handleGetFirmwareVersion() / CC_Version_GetFirmwareVersion_handler()

handleGetFirmwareVersion() was a compatibility macro of CC_Version_GetFirmwareVersion_handler(). The firmware version is now fetched directly inside the command class using ZPAL functions. An application must continue to set the version in config_app.h using APP_VERSION, APP_REVISION and APP_PATCH.

This function is defined weakly in CC_Version.c and can be overridden in case an application requires support for more than one firmware target. The Z-Wave example applications use one firmware target only. For more information about firmware targets, see the Command Class Firmware Update Meta Data specification.

CC_Version_GetHardwareVersion_handler()

This function was replaced by a configuration parameter for the ZAF component that can be set in Simplicity Studio.

7.5 Command Class Device Reset Locally

Command Class Device Reset Locally was refactored to support excluding it from a build and thus creating the option for a smaller application that can be used for updating Secure Element.

CC_DeviceResetLocally_notification_tx() no longer takes an AGI profile or a callback function as arguments. The profile was removed because the recommended profile is Lifeline. The callback is now declared externally in CC_DeviceResetLocally.h.

7.6 Command Class Power Level

CC_PowerLevel.h was removed as CC Power Level no longer has an API in the application. The initialization and reset of this command class is now done automatically by ZAF_Init() and ZAF_Reset().

7.7 MFG Token Z-Wave Region

Starting with 7.18.0, Z-Wave Region is saved in MFG Token `MFG_ZWAVE_COUNTRY_FREQ`. When the binary is flashed to the board for the first time, the value of Z-Wave region gets written to this token.

It is important to note that the value of `MFG_ZWAVE_COUNTRY_FREQ` does NOT get erased when a binary with a different region is flashed. Once written, this value is only changed by writing a new value, or `0xFF` (uninitialized). This applies to all applications.

Useful commands:
- Write REGION_US value : `commander flash --tokengroup znet --token MFG_ZWAVE_COUNTRY_FREQ:0x01`
- Read value: `commander tokendump --tokengroup znet --token MFG_ZWAVE_COUNTRY_FREQ`
- Delete value `commander flash --tokengroup znet --token MFG_ZWAVE_COUNTRY_FREQ:0xFF`

It is still possible to set the region using `ZW_REGION` value, but it is important to remember that `MFG_ZWAVE_COUNTRY_FREQ` must be `0xFF`, otherwise `ZW_REGION` will be ignored.

Please note that Serial API reads the region from the MFG token on first startup only. After that, region is stored in NVM and can be changed by using PC Controller, for example.

7.7.1 OTA Consideration

Beginning with 7.18.0, the value in the GBL file region used for firmware update is ignored if a valid value exists in `MFG_ZWAVE_COUNTRY_FREQ`. This allows the GBL file to have any region, and the region used by the app would still match the original value after OTA.

NOTE: For firmware update from versions up to and including 7.17 to version 7.18 or later, the region in the GBL file MUST match the value used by the app.

7.8 Z-Wave Long Range SmartStart Sequence

SmartStart frames order has been swapped for LR supporting devices, so that LR inclusion is prioritized. This change does not affect devices that do not support LR.

7.9 Z-Wave Debug Print

To enable debug output, build with `Z-Wave Debug Print` component. This component will include the correct iostream component needed. This was done to reduce the FLASH usage on debug builds.

8 Open Source Software

Z-Wave is using FreeRTOS as the underlying OS, and it is based on FreeRTOS Kernel V10.4.3.

9 Using This Release

This release contains the following

- Z-Wave Plus V2 Application Framework
- Z-Wave Certified Applications for a broad range of smart home applications
- Z-Wave Protocol and Serial API Applications

If you are a first-time user, Z-Wave documentation is installed with the SDK. See INS14280: Z-Wave Getting Started for End Devices, INS14278: How to Use Certified Apps in Z-Wave, and INS14281: Z-Wave Getting Started for Controller Devices for instructions.

This SDK depends on a Gecko Platform. The Gecko Platform code provides functionality that supports protocol plugins and APIs in the form of drivers and other lower layer features that interact directly with Silicon Labs chips and modules. Gecko Platform components include EMLIB, EMDRV, RAIL Library, NVM3, PSA, and mbedTLS. Gecko Platform release notes are available through Simplicity Studio's Launcher Perspective.

9.1 Installation and Use

Order a Z-Wave Wireless Starter kit. The kit offers the easiest and fastest way to start evaluation and development of your own Z-Wave mesh application. It provides a single world-wide development kit for both end devices and gateways with multiple radio boards, with which developers can create a mesh network and evaluate the Z-Wave module.

The Z-Wave and Z-Wave Long Range 700/800 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.

Simplicity Studio installs the GSDK by default in:

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

To implement a specific application, Silicon Labs recommends starting with one of the existing pre-certified apps with the desired Role Type.
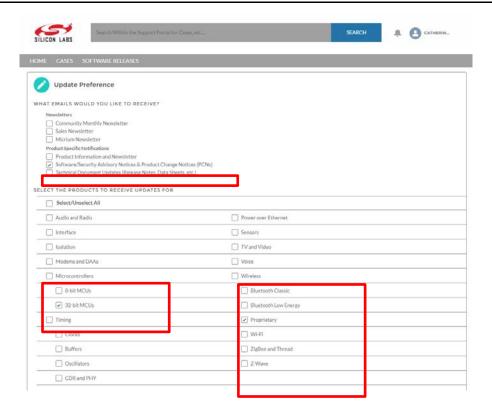
9.2 Security Information

Secure Vault Integration

This version of the stack are using secure vault interface for key management of asymmetric keys (ECC Curve 25519) and Symmetric keys (AES).

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.

9.3   Support

Development Kit customers are eligible for training and technical support.

See support resources and contact Silicon Laboratories support at http://www.silabs.com/support.

10 Product Life Cycle and Certification

Silicon Labs will add new features based on market requirements and continuously improve the Z-Wave Protocol to position the Z-Wave Ecosystem. The Z-Wave Protocol Life Cycle is a process to provide rapid innovation, new features and robust matured protocol release to Z-Wave Partners. The Z-Wave Protocol Life Cycle defines the maturation process of Z-Wave Protocol generations and consist of three phases divided in five Life Cycle stages. A change in the Z-Wave SDK utilized for a specific device does require recertification; however, the type of certification required, the amount of testing needed, and the associated fees depend on the scope of the change. Refer to Z-Wave Alliance home page https://z-wavealliance.org/ for details.

Table 10-1. Z-Wave SDK Life Cycle Status

Series	SDK Version	Release Date [DD-MMM-YYYY]
700/800	7.19.2 GA	08-MAR-2023
700/800	7.19.1 GA	01-FEB-2023
700/800	7.19.0 Pre-Certified GA	14-DEC-2022
700/800	7.18.8 GA	13-SEP-2023
700/800	7.18.7 OSR	28-AUG-2023
700/800	7.18.6 GA	05-APR-2023
700/800	7.18.5 OSR GA	15-MAR-2023
700/800	7.18.4 GA	18-JAN-2023
700/800	7.18.3 GA	19-OCT-2022
700/800	7.18.2 GA	28-SEP-2022
700/800	7.18.1 GA	17-AUG-2022
700/800	7.18.0 Pre-Certified GA	08-JUN-2022
700/800	7.17.2 GA	09-MAR-2022
700/800	7.17.1 Pre-Certified GA	28-JAN-2022
700/800	7.17.0 Pre-Certified GA	08-DEC-2021
700	7.16.3 GA	13-OCT-2021
700	7.16.2 GA	08-SEP-2021
700	7.16.1 GA	21-JUL-2021
700	7.16.0 Pre-Certified GA	16-JUN-2021
700	7.15.4 GA	07-APR-2021
700	7.15.2 Pre-Certified GA	27-JAN-2021
700	7.15.1 Pre-Certified GA	09-DEC-2020
700	7.14.3 GA	14-OCT-2020
700	7.14.2 GA	09-SEP2020
700	7.14.1 GA	29-JUL-2020
700	7.14.0 Beta	24-JUN-2020
700	7.13.11 GA	02-NOV-2022
700	7.13.10 GA	18-AUG-2021
700	7.13.9 GA	03-MAR-2021
700	7.12.2 GA	26-NOV-2019
700	7.12.1 GA	20-SEP-2019

11 Legal

11.1 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 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**

11.2 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®, Gecko OS, Gecko OS Studio, ISOmodem®, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, 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.

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

SILICON LABS

www.silabs.com