



Zigbee EmberZNet SDK 6.7.0.0 GA

Gecko SDK Suite 2.7

December 13, 2019

Silicon Labs is the vendor of choice for OEMs developing Zigbee networking into their products. The Silicon Labs Zigbee platform is the most integrated, complete, and feature-rich Zigbee solution available.

Silicon Labs EmberZNet SDK contains Silicon Labs' implementation of the Zigbee stack specification.

These release notes cover SDK version(s):

6.7.0.0 released December 13, 2019.



KEY FEATURES

- Adds sample application for Zigbee Alliance TIS/TRP RF testing
- New optimized network steering algorithm
- Limited token set option for Smart Energy CommsHub
- Expanded EZSP Frame Header to support additional frame IDs

Important Notice

Zigbee support for EM35x devices is planned for removal after the June 2020 release. More information on this move, including End of Service dates, can be found in the [Deprecated Items](#) section of this document.

Compatibility and Use Notices

If you are new to the EmberZNet SDK, see [Using This Release](#).

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 8.30.1.

- Using wine to build with the `IarBuild.exe` command line utility or IAR Embedded Workbench GUI on macOS or Linux could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.
- Customers on macOS or Linux are advised not to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

GCC (The GNU Compiler Collection) version 7.2.1, provided with Simplicity Studio.

- GCC is not supported with the EM35x, EM358x, and EM359x device families.

Contents

- 1 New Items2
 - 1.1 New Plugins2
 - 1.2 New APIs2
 - 1.3 New Sample Applications2
- 2 Improvements3
 - 2.1 Plugin Changes.....3
 - 2.2 API Changes.....3
 - 2.3 Sample App Changes4
 - 2.4 Documentation Changes.....4
- 3 Fixed Issues5
- 4 Known Issues in the Current Release.....7
- 5 Deprecated Items.....10
- 6 Removed Items.....11
- 7 Using This Release12
 - 7.1 Installation and Use.....12
 - 7.2 Support.....12
- 8 Legal13
 - 8.1 Disclaimer13
 - 8.2 Trademark Information.....13

1 New Items

1.1 New Plugins

None

1.2 New APIs

For additional documentation please refer to our online Zigbee reference material at <https://docs.silabs.com/zigbee/latest/>

Added in release 6.7.0.0

emberAfNcpGpSentHandler – allows xNCP applications to handle DGP Sent event processing.

emberGetNeighborFrameCounter – fetch the last received frame counter as found in the Network Auxiliary header for the specified neighbor or child. This API returns an EmberStatus depending on whether the Frame Counter is found in the neighbor or child table.

MAC Timestamps

MAC timestamps for received packets are now exposed on the Zigbee stack. We provide an API

emberGetLastHopTimestamp(uint32_t *lastHopTimestamp) to grab the value of the MAC timestamp for the latest received packet. The value returned is the MAC timer (ticks since up) at the moment when the Start Frame Delimiter (SFD) was received for the latest message.

This function may only be called from within:

- emberIncomingMessageHandler()
- emberNetworkFoundHandler()
- emIncomingRouteRecord()
- emberMacPassthroughMessageHandler()
- emberIncomingBootloadMessageHandler()

Please see the documentation for **emberGetLastHopTimestamp** for more details, including platform-specific information.

1.3 New Sample Applications

Added in release 6.7.0.0

StandardizedRfTesting

Sample application demonstrating RF testing through TIS (Total Isotropic Sensivity) / TRP (Total Radiated Power) interfaces. The current version implements a pre-standardization version of the specification (docs-19-01701-00).

2 Improvements

2.1 Plugin Changes

Changed in release 6.7.0.0

Network Steering

Two new options have been added to the **Network Steering** plugin to improve performance.

- The Optimize Scans option (enabled by default) collects channel scans together and then joins networks that have been saved in a cache, resulting in fewer scans issued by the device when joining a network. For example, all channels will be scanned on the current mask, then networks are joined one by one without issuing another scan.
- The Try All Keys At Once option (requires the Optimize Scans option to be enabled) will use the install code derived key, centralized key, and distributed key when joining each candidate network. This results in faster joining times but may suffer from joining promiscuous networks allowing joining using the centralized key (e.g. if a device wishes to join with an install code, it may end up joining a Home Automation network before it finds its desired network using its install code derived key). Users that wish to prioritize joining using install code keys should not check this option.

The default for `bdbSecondaryChannelSet` has been changed from `0x07FFF800` to `0x4E73000`, resulting in no overlap between the primary and secondary channel set. This is reflected in the **Network Steering** plugin with Optimize Scans enabled.

The Scan Duration Component option in the **Network Steering** plugin default has been changed from 5 to 4. This change results in spending around 260 milliseconds when scanning a channel, down from around 500 milliseconds per channel with value 5. Users should ensure that beaconing devices do not jitter for longer periods than this value when sending a beacon.

OTA Bootload Cluster Storage

The OTA Cluster Platform bootloader plugin has been updated with the ability to flash all images embedded within the .ota image. Each image is consecutively flashed onto the device with a device reset in-between flashes.

SE Token Libraries

Created an optional optimized token map for ZigBee Smart Energy Commshub devices (em356, em357, efr32xg1x). Removes unrelated tokens from stack libraries (including Green Power, Multi-Network, and ZLL). This feature can be enabled on select devices by choosing the appropriate option in the "Other Options" tab of AppBuilder in studio.

Zigbee PRO Stack Library

All ZigBee applications will now be built with the default security features enabled. The **Security Library Core** plugin has been merged into the main stack libraries (**ZigBee PRO Stack Library**, **ZigBee PRO Leaf Library**, etc) and the **Security Library Core Stub** has been removed.

The `emberSetLogicalAndRadioChannel()` API has moved from **ZigBee Light Link Library** to **ZigBee PRO Stack Library**.

HAL Library

Most applications running on EFR32 devices now take advantage of the new Simple MPU component, which disallows running from all RAM except the area specifically set aside for RAM functions. Due to a known incompatibility, the Simple MPU component is not used to initialize the MPU when running DMP or other Micrium-based applications.

Micrium RTOS

Stack size of the statistic task of the **Micrium RTOS** plugin was increased from 128 bytes to 256 bytes.

2.2 API Changes

Changed in release 6.7.0.0

EZSP Protocol Version 8 and Secure EZSP Protocol Version 2

Both EZSP and Secure EZSP have adopted a new frame format with the following changes: (1) the fields of "Frame Control" and "Frame ID" are now two bytes; (2) no longer use "Legacy Frame ID"; (3) consume two bits of "Frame Control" to indicate the frame format version which is version 1 now.

The new frame format of EZSP protocol version 8 is described in the "UG100: EZSP Reference Guide" and the new frame format of Secure EZSP protocol version 2 is described in the "AN1125: Creating and Using a Secure EZSP Host-to-NCP Interface". More specific EZSP examples are shown in the "AN711: SPI Host Interfacing Guide for Zigbee".

EZSP_SET_GPIO_RADIO_POWER_MASK has been moved into the "EzspValueId" section and renamed to EZSP_VALUE_GPIO_RADIO_POWER_MASK. Use the EZSP_SET_VALUE command to achieve the equivalent function.

EZSP_SET_CTUNE and EZSP_GET_CTUNE have been moved into the "EzspConfigId" section and renamed to EZSP_CONFIG_CTUNE_VALUE. Use the EZSP_SET_CONFIGURATION_VALUE and EZSP_GET_CONFIGURATION_VALUE commands to achieve the equivalent functions.

Beacon Caching

The beacon caching code now records the most prioritized beacon per network. See emberGetFirstBeacon and emberGetNextBeacon for APIs on how to traverse the beacon cache. The most prioritized beacon is subject to parent classification, RSSI, LQI, and other factors.

2.3 Sample App Changes

Note: Sample applications in releases beginning with June 2020 will no longer target 256kB parts, including em35x, EFR32MG1, and EFR32MG14. This is due to new developments in the Zigbee specification that have led to increased flash requirements beyond the 256 kB provided by these parts.

2.4 Documentation Changes

None

3 Fixed Issues

Fixed in release 6.7.0.0

| ID # | Description |
|--------|---|
| 239480 | Fixed a code flow issue causing network leave announcements to be sent before the leave response. The network leave announcement will now wait until the ZDO Leave response goes out (or until APS retries run out, if enabled). |
| 330132 | NCP applications can now be augmented with custom token headers. |
| 331528 | There was an issue with the name of the Manufacturing Library CLI event to check that sending is complete. The metadata has been fixed to correctly generate the event code and the handler function has been renamed to match. The correct event is now set active in the mfglibRxHandler. |
| 348357 | As the ApsFrameCounter currently is a shared variable for multi networks, resetting it every time when the node is joining a new network may cause the TC of the joined network to drop the APS encrypted packets. For a solution, we will check whether or not the current joining network is the only active network, if no, don't reset the ApsFrameCounter. |
| 359622 | If the OTA incoming message is an image page request, then don't send a duplicate response when handling the incoming message, because the response for an image page request is also handled by the "sendBlockRequest()" function. |
| 371180 | Fixed bug where including the Battery Monitor plugin could disable other features of the chip inadvertently. |
| 385119 | Fixed "option install-code" CLI command to stop adding derived key into the link key table for Z3. |
| 385412 | After receiving a device announce, the trust center will not send out an immediate route request with the default HIGH RAM concentrator type, anymore. Instead, an internal message is passed to the source route library to schedule a route request with correct concentrator type. The new request is scheduled according the min/max timing period between MTORRs, which itself could be configured through the Concentrator Support plugin. |
| 385812 | The existing em4 plugin callbacks have been completely renamed. Now we have 4 callbacks, 2 of which needs to be implemented by the application if it wishes to use the em4 mode. These 2 callbacks are emberAfPluginEm4EnterCallback() and emberAfPluginEm4ExitCallback(). |
| 387743 | Zigbee leaf stack no longer consumes any memory creating/storing a routing table. |
| 390692 | A device that receives a unicast ZCL Groups cluster AddGroupIfIdentifying command, and which is not identifying, now returns ZCL status SUCCESS in the Default Response message. Previously it returned status FAILURE, which is wrong since the device is behaving correctly (not identifying, thus does not add group) per ZCL specification. Note that this is observed only for a received unicast command; no response is sent for a command received via multicast/broadcast. |
| 398189 | The level-control plugin's implementation of the emberAfOnOffClusterLevelControlEffectCallback() function is refactored to eliminate a circumstance where an "if" conditional statement tests a variable for a value that was earlier assigned to that variable. Coverity flags this as an always-true condition. |
| 400188 | Plugins enabled/disabled from Appbuilder can now enable/disable hardware configurator modules/peripherals given that an Appbuilder plugin is associated with a hardware configurator. |
| 403200 | The application image upload window has been updated to remove all pre-built Ember (em*) binaries for HA/Se/Sensor/Sink. |
| 408103 | An issue with debug output timing on EFR32xG21 has been fixed. This results in improved speed and/or reliability in a number of areas, including sleep and NCP communications, when the Debug plugin is enabled. |
| 424410 | Fixed the issue of emberSerialReadAvailable returning values greater than the buffer size. |
| 426464 | CommsHub will now send a response when receiving an invalid enum in a EndDeviceTimeOut Request. |
| 429083 | Fixed EZSP host makefile compatibility issue with Windows 10/Cygwin environments. |
| 430883 | Fixed an issue where Dual Band Coordinators could cause the network retry queue to overflow when forwarding a broadcast fails on one interface, but succeeds on another. We now submit broadcast packet on second MAC only if it is succeeded on native MAC. |
| 430956 | Fixed an issue where the original channel's used duty cycle could be reset during a channel scan. |
| 431858 | Incorrect interrupt names were corrected in diagnostic code for EFR32xG21 and xGM21 parts. |
| 431860 | Fixed a potential out of bounds memory access in token code. |
| 434267 | Fixed bug on EFR32 where checking for USART TX being idle returned true before transmission was complete, which had resulted in some NCP bootloading issues. The affected functions were halInternalUartTxIdle and COM_InternalTxIdle. |
| 437423 | emberGetChildData was not properly masking an internal flag in the timeout before returning the data. This resulted in the stack diagnostics plugin printing incorrect timeout values. The issue has been corrected. |

| ID # | Description |
|--------|--|
| 440341 | Fixed a GCC linker issue. The linker will error out if there is a possibility of overlap of data_init section. |
| 440898 | Fixed issue where NVM3 library was not included properly in GCC projects for EFM32 in Simplicity Studio. |
| 442654 | Fixed a race condition that could result to network retry queue overflow. The race condition could occur on a router or coordinator device, causing the NWK Retry Queue to lock up and overflow, such that no more NWK or APS layer frames would be transmitted. This could occur when all of the following conditions were met: (1) An APS broadcast frame to addressed to 0xFFFF is pending on the parent for relaying to at least one sleepy child on next poll. (2) Parent queues this broadcast data as a MAC unicast for transmission to its sleepy child after receiving a Data Request and begins transmission of the data. (3) A second broadcast to 0xFFFF is queued for relaying to sleepy children before the in-flight transmission of the first broadcast completes its handoff to the end device. |
| 444155 | We have observed a minor issue in password-protected-cli. In case of a successful password entry after a few wrong attempts, number of wrong password entry attempts did not reset in the tokens. Therefore, the number of wrong password attempts carried over after the next reset. Depending on the wrong password entry threshold, this could introduce addition of an unwanted lockout period in between resets. To fix this, now we reset the token value, after a successful password entry attempt. |
| 444215 | There was an issue when setting a default value for a 64-bit attribute value (with ZigBee data type: BITMAP64) in the appbuilder. |
| 444454 | An issue has been fixed where emberGetNextBeacon would fail to return a valid, next beacon. This would cause network steering to occasionally fail to join a network when the Optimize Scans option was ticked. |
| 444531 | A timeout issue with em35x SPI NCPs has been resolved by moving the seeding of the PRNG with strong rand numbers up in the initialization process before initialization of the serial connection with the host. Additionally, the default wait time in host apps has been increased from 300ms to 350ms (WAIT_SECTION_TIMEOUT_MS). |
| 444929 | Fixed a race condition that could result to network retry queue overflow. The race condition could occur on a router or coordinator device, causing the NWK Retry Queue to lock up and overflow, such that no more NWK or APS layer frames would be transmitted. This could occur when all of the following conditions were met: (1) An APS broadcast frame to addressed to 0xFFFF is pending on the parent for relaying to at least one sleepy child on next poll. (2) Parent queues this broadcast data as a MAC unicast for transmission to its sleepy child after receiving a Data Request and begins transmission of the data. (3) A second broadcast to 0xFFFF is queued for relaying to sleepy children before the in-flight transmission of the first broadcast completes its handoff to the end device. |
| 445334 | Network steering did not handle a busy channel status properly, and therefore it would have failed joining any channel, if one channel was busy. The issue is now fixed. |
| 445625 | The Device Version for endpoints in the following dynamic multiprotocol sample applications has been corrected from 2 to 1: DynamicMultiprotocolLight, DynamicMultiprotocolLightSed, DynamicMultiprotocolZigbeeZllLightBle. |
| 445764 | Fixed a problem with energy scans on sub-GHz channel pages which could result in a packet buffer overflow, corrupting other packet data used by the stack. |
| 447448 | Fixed emlib issue where the core clock frequency was used instead of the hf clock frequency when calculating the frequency of HFCLKLE which can be used in the LFB clock tree. This bug could cause wrong baudrate calculations for LEUART when using HFCLKLE in systems where the HF clock and core clock are different frequency. |
| 447797 | EMLIB IADC: The definition of `iadcNegInputGnd` has been modified to set PINNEG to 1. This prevents a polarity error when performing IADC conversions between supply pins and ground. |
| 449148 | Fixed issue where EFR32xG21 NCP apps would fail to compile due to missing GPIO definitions. |
| 452265 | Restored the use of USART0_RX_IRQHandler on EFR32xG2 devices This will allow use of wake up from idle plugin and software flow control on EFR32xG21. |

4 Known Issues in the Current Release

Issues in bold were added since the previous release.

| ID # | Description | Workaround |
|--------|---|---|
| 60858 | Sleepy broadcast payload is sometimes corrupted when relaying to child | No known workaround |
| 60975 | EZSP_VALUE_TOKEN_STACK_NODE_DATA, EZSP_VALUE_UART_SYNC_CALLBACKS, EZSP_VALUE_MAXIMUM_INCOMING_TRANSFER_SIZE, and EZSP_VALUE_MAXIMUM_OUTGOING_TRANSFER_SIZE ValueIDs are writable but not readable. | No known workaround |
| 70851 | System-timer implementation precludes runtime adjustment to GPIO wake handling | No known workaround |
| 82569 | RAM corruption (in Packet Buffers) could occur if MAC Filter Match List Size is non-zero and a list of the maximum size is provided to ezspSetValue for EZSP_VALUE_MAC_FILTER_LIST. | No known workaround |
| 106307 | Nodetest calChannel command does not wake the radio to work properly. | No known workaround |
| 119939 | ZDO IEEE Request's APS ACK proxied by parent incorrectly includes long source address. | No known workaround |
| 135649 | Multi-networking can cause APS frame counter confusion between networks. | Use emberAfSecurityInitCallback to add EMBER_NO_FRAME_COUNTER_RESET to EmberInitialSecurityBitmask. |
| 229938 | ZLL devices are sending ZLL device information frames with the 0x0104 HA/Z3 common profile ID instead of 0xC05E ZLL profile ID. | No known workaround |
| 235222 | Virtual UART (VUART) works on Ethernet but does not work over USB on WSTK. | No known workaround |
| 251287 | To achieve the lowest current during sleep on EFR32xG12, EFR32xG13, and EFR32xG14 parts, you must turn on voltage scaling. However, the radio will not operate with voltage scaling turned on, so to turn it on you must also make sure to disable it after each wake-up. Furthermore, some resets will not turn off voltage scaling, so please ensure that it is disabled before attempting to turn on the radio. Note that there is a ramp when turning voltage scaling on or off, so enabling this feature may increase the time it takes to go to sleep or wake up. | No known workaround |
| 261670 | Harden the ZLL touchlink process to mitigate malicious attacks | No known workaround |
| 266341 | Z3 Light sample app has two endpoints that support similar cluster commands, so duplicate responses may be generated for certain commands. | No known workaround |
| 271644 | A device that performs a classic join to a legacy ZLL gateway may eventually leave the network on its own initiative. | No known workaround |
| 278063 | Smart Energy Tunneling plugins have conflicting treatment/usage of address table index | No known workaround |
| 281231 | Enabling Serial 3 or USB functionality on EM358x and EM359x may cause memory management faults and other errors. As EM358x and EM359x USB support has been deprecated, please ensure that Serial 3 and USB functionality are disabled. | No known workaround |
| 281832 | Green Power Common plugin incorrectly formats groupList and groupListCount parameters of GP Pairing Configuration frame. | No known workaround |
| 289569 | Network-creator plugin power level picklist doesn't offer full range of supported values for EFR32 | No known workaround |
| 295498 | UART reception sometimes drops bytes under heavy load in Zigbee+BLE DMP use case | Use hardware flow control or lower the baud rate. |
| 301024 | Currently the Dynamic Multi-Protocol Sample applications do not build with GCC, they require the IAR compiler due to dependencies on both the BLE and Micrium stacks. | No known workaround |

| ID # | Description | Workaround |
|---------------|---|--|
| 312291 | The halCommonGetIntxxMillisecondTick functions on linux hosts currently use the gettimeofday function, which is not guaranteed to be monotonic. If the system time changes, it can cause issues with stack timing. The workaround is to modify these functions to use clock_gettime with the CLOCK_MONOTONIC source instead. | No known workaround |
| 331438 | Service discovery may time out too quickly in busy networks. | Define EMBER_AF_DISCOVERY_TIMEOUT_QS to customize the timeout period. |
| 333146 | The coexistence plugin behavior has changed to be required for all applications. By default coexistence functionality should be stubbed out when not configured via board header (EM35x) or not enabled via HWCONF (EFR32). Appbuilder may automatically enable the HWCONF coexistence module without warning which can trigger an error for dynamic multiprotocol applications, which do not yet support coexistence. | No known workaround |
| 338151 | Initializing NCP with a low packet buffer count value may cause corrupt packets. | No known workaround |
| 345167 | Sleepy end devices occasionally not sending APS ACK for received APS unicasts polled from parent. | No known workaround |
| 356937 | Read/write attribute CLI commands do not support manufacturer-specific ZCL attributes. Some implementations may allow local CLI debug access to display or modify these attributes. | Access the attributes from a remote device in the network via ZCL global Read/WriteAttributes commands. |
| 362133 | The default chip RSSI offset on the EFR32 chips is incorrect. For accuracy, we recommend measuring this on your hardware and then applying it with the rref RAIL_SetRssiOffset() API. In the future, more reasonable defaults will be provided, however, the board dependent component may still need to be measured for custom hardware. | Use RAIL_SetRssiOffset() |
| 363162 | There is a bug in emberAfAddAddressTableEntry which could allow for duplicate entries in the address table | Under Investigation |
| 398694 | Disabling endpoint 2 (Touchlink) in the Z3Light sample app causes high emberRunTask execution time. | Under Investigation |
| 401119 | The Image Builder utility may flag a harmless warning when creating an OTA image from a .gbl file. The warning is "Bad length for tag. Size X < Min size Y," where X and Y are variable. The warning is inaccurate and will be removed in a future release. | Under Investigation |
| 410372 | There is a buffer allocation bug in the NCP framework which could result in all packet buffers being re-initialized under exceptional high load conditions. | Avoid using the 0xFF option to maximize packet buffers on the NCP |
| 426066 | OTA client could reset when it starts to apply the new image if all debug print is disabled. | No known workaround |
| 426200 | Enabling a large number of endpoints (~40) and setting a large reporting table entries (close to 255) causes emberAfPluginReportingLoadReportingConfigDefaults routine to take ~14 seconds to run which results in a watchdog reset. | To get around the issue, disable or reset the watchdog in the loop. |
| 437502 | hallNit() is called twice in EmberZNet SPI NCP Host Application. | No known workaround |
| 437704 | The OccupiedCoolingSetpoint attribute should be optional by default in Appbuilder; workaround is modification of ha.xml file in SDK. | Modify the "optional" flag from false to true in the OCCUPIED_COOLING_SETPOINT attribute in app/zcl/ha.xml |
| 437817 | After EmberZNet 6.6.0, joining and insecurely rejoining children are now subject to a new, temporary timeout EMBER_SHORT_CHILD_TIMEOUT. This defaults to 2 minutes if undefined and should be able to be overridden. However, it doesn't function as expected since defining and changing the value of EMBER_SHORT_CHILD_TIMEOUT doesn't take any effect. The short temporary timeout for the joining device will always stay as 2 minutes. | No known workaround |
| 438993 | Certain IRQ Pin/Port settings do not allow wake on interrupt in some EM3xx parts. | No known workaround |

| ID # | Description | Workaround |
|--------|--|--|
| 442664 | In a dense network with many devices joining simultaneously, a packet buffer assert is sometimes seen when collecting beacons during network steering. | No known workaround |
| 444721 | There is an incompatibility between the prebuild/postbuild scripts, wine 4.0, and IAR Build. As a result, this release does not contain any prebuilt EBL or GBL files. | Please use wine 1.8.5 to trigger IAR build processes in a unix environment. |
| 451029 | The Z3Light is configured as a Router Zigbee Device Type instead of Coordinator/Router. As a result, whenever modifications to the ZCL configuration are made in an application, the resulting update will disable coordinator specific plugins such as Network Creator. | Change the Zigbee Device Type to "Coordinator or Router". Re-enable the Network Creator plugin if disabled. |
| 451867 | When using the network steering plugin with both 'Optimize scans' and 'Try all keys at once' options enabled in the plugin UI, the node will fail to join any networks at all if an install code token has not been flashed to the device. The device will exit the join process with an output of "Error: NWK Steering could not setup security: 0xB7." | To avoid this issue, flash an install code token to the device. |
| 452682 | Dynamic Multi-Protocol applications will not run properly on EFR32xG21 with the default clock configuration. | In the HWCONF, in the CMU peripheral, select HFRCO as the HF Clock Source and "80 MHz from HFXO" as the PLL Configuration. |

5 Deprecated Items

Deprecated in release 6.7.0.0

Zigbee support for EM35x devices - Planned for removal after June 2020

Reason: New developments in the Zigbee specification have led to increased flash requirements beyond the 256 kB provided by most parts, and introduced new security requirements that EM35x devices cannot adequately support. Software support for these parts will be maintained for the full lifetime of the products in accordance with [Silicon Labs longevity commitments](#), but no new features will be released. Customers are advised to start any new Zigbee developments using EFR32 series parts.

End-of-Service (EoS) Date: June 2020. The GSDK release at this time will be the last release to support EM35x devices.

Maintenance Period: Critical bug fixes and security patches may be made available on currently supported GSDK releases for the lifetime of the part, at least through September 2024.

Network Steering Configured Key API

The "plugin network-steering pre-configured-key-set" CLI command and emAfPluginNetworkSteeringSetConfiguredKey API will be removed in a future software release. Users that wish to use a configured key during network steering should instead add the desired key value into the transient key table before starting network steering. The EUI used for this key should be the wildcard EUI (FFFFFFFFFFFFFFFF).

6 Removed Items

Removed in release 6.7.0.0

In previous releases, source routing (add/remove table) and concentrator code (scheduling many-to-one route requests, handling route error) previously existed as 2 different copies, one host side plugin version and one NCP version. The host side source routing table and MTORR scheduling have now been removed.

To enable source routing, include both the concentrator plugin (configuration part) and the source routing library (the core table management and MTORR scheduling functionality).

As a side effect most of the APIs/callbacks allowing the app to interfere with the source routing has been deprecated. Specifically, the ones that depended on the host-side source-route table, including:

- emberAppendCachedSourceRouteFromHost()
- ezspSetSourceRoute()
- emberFindSourceRoute()
- emberAfEzspSetSourceRoute()
- emberAfEzspSetSourceRouteCommandCallback()
- setConcentrator/ezspsetconcentrator()
- emberAfPluginEzspZigbeeProSetConcentratorCommandCallback()
- emberAfPluginConcentratorPrintHostSourceRouteTable() and print-host-table CLI command.

7 Using This Release

This release contains the following:

- Zigbee stack v6.7.0.0
- Zigbee Application Framework v6.7.0.0
- Zigbee Sample Applications

For more information about Zigbee and the EmberZNet SDK see [UG103.02: Zigbee Fundamentals](#).

If you are a first-time user, see [QSG106: Getting Started with EmberZNet PRO](#), for instructions on installing and configuring your development environment, building and flashing a sample application, and documentation references pointing to next steps.

7.1 Installation and Use

Stack installation instruction are covered in [QSG106: Getting Started with EmberZNet PRO](#).

Use the EmberZNet SDK with the Silicon Labs Simplicity Studio V4 development platform. Simplicity Studio ensures that most software and tool compatibilities are managed correctly. Install software and board firmware updates promptly when you are notified.

Documentation specific to the SDK version is installed with the SDK. Additional information can often be found in the [knowledge base articles \(KBAs\)](#). API references and other information about this and earlier releases is available on <https://docs.silabs.com/>.

7.2 Support

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

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

8 Legal

8.1 Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications.

Application examples described herein are for illustrative purposes only.

Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

8.2 Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISModem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri, Z-Wave and others are trademarks or registered trademarks of Silicon Labs.

ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings.

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