

EmberZNet Stack Release Notes

Product: Release Notes for EmberZNet PRO 5.7.2 GA

Release Date: April 29, 2016

1 Overview

This release contains the GA release of EmberZNet PRO 5.7.2.

It contains the following:

- Hardware Abstraction Layer (HAL) software provided as source code
- ZigBee PRO Networking stack
- Ember ZCL Application Framework (AFV2) including Sample Applications
- Application, Serial-uart-only and Serial-uart-ota Bootloaders
- Full set of Documentation including API guides and Application Notes

An installer for the Simplicity Studio development environment is available on the support portal.

It is our policy to certify EmberZNet PRO stack releases as ZigBee Certified Platforms. Note that ZigBee Certification may not be completed at the time of release, so please check with Support before relying on this release as the basis for ZigBee product certification.

1.1 Software Version

Stack Software: Version 5.7 build 444. The version number is defined as SOFTWARE_VERSION in the source code in the "stack/config/config.h".

This stack software release is intended to be used with the following versions of the other components:

- Simplicity Studio 3.3
- ISA3 Utilities Version 4.8.0.0 build 104, used with both EM3x and Mighty Gecko (EFR32MG) development environments
- Simplicity Commander 0.14.1 for interacting with the WSTK for Mighty Gecko platforms
- WSTK Firmware version 0v14p2b458
- IAR Embedded Workbench for ARM 7.30.1. If you are receiving this release as a product upgrade, the installer for IAR Embedded Workbench for ARM 7.30.1 is available at <ftp://files.iar.com/pub/silabs/EWARM-7301.exe>.
- ECC Library 1.0.1.20 (Available upon request via Silicon Labs support)

1.2 What's New

1.2.1 What was changed in EmberZNet 5.7.2.0:

- Improved the upgrade process from SIMEEv1 to SIMEEv2.
- Added EM358x Occupancy Sensor reference application.
- General bug fixes.

1.2.2 What was changed in EmberZNet 5.7.1.0:

- Added NCP support for testing and setting CTUNE on EFR32 family.
- Added documentation on how to support 64 end devices (please use xNCP)
- Updated trust center support to ignore insecure rejoins. In order to prevent this at the end device, please refer to the following Silicon Labs knowledge base article (KBA) found [here](#)
- General bug fixes.

1.2.3 What was changed in EmberZNet 5.7.0.0:

- Support for Mighty Gecko SOC and NCP Platforms.
- ZigBee PRO revision 21 (r21) support.
- Support for ZigBee 3.0 Features
- Antenna diversity support at the stack and HAL level: If enabled via Board Header file, antenna selection is changed automatically by stack after a MAC TX failure is detected. See dev0680_0881.h board header for example.
- Simplicity Studio 3.3 Support (replacing Ember Desktop as configuration tool). If you are upgrading from an earlier version of Simplicity Studio, please ensure that you remove that version before installing Simplicity Studio 3.3.
- Support for standard data types (such as "uint8_t" in place of "int8u")
- Includes Home Automation Reference app plugins previously introduced in 5.4.4.0.
- Improved support for EZSP over USB.

1.3 Support

Development Kit customers are eligible for training and technical support. You can use the Silicon Laboratories web site <http://www.silabs.com/zigbee> 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>

2 Using This Release

2.1 First Time Users

See QSG106, Getting Started with EmberZNet PRO, for step-by-step instructions on installing software, configuring the Simplicity Studio development environment, and building and flashing an example application.

- **EM35x Development Kit:** Refer to the Quick Start Guide included in your Development Kit for detailed step-by-step instructions about initially setting up your kit.
- **EFR32MG Mesh Networking Kit:** see QSG106, Getting Started with EmberZNet PRO, for instructions.

2.2 Accessing Documentation

- To view the full set of documentation, click on:

Start → All Programs → Silicon Labs → EmberZNet 5.7.2 → Documentation

Or open the "documentation/index.htm" file from the root directory of the installed Ember stack. The documentation includes HTML pages describing the Ember Stack API and the Ember HAL API, along with a Development Kit Users Guide, an Application Developers Guide, the chip data sheet, and various Application Notes.

- To view information on the ISA3 and associated utilities, click on:

Start → All Programs → Ember → ISA3 Utilities → Documentation

Or open the "docs/index.html" file from the root directory of the installed ISA3 Utilities. This includes explanations of how to use the ISA3 Command line utilities, how to program certificates and install codes, as well as general documentation about the operation and use of the ISA3.

- To view documentation for IAR Embedded Workbench, click on:

Start → All Programs → IAR Systems → IAR Embedded Workbench for ARM → Release Notes

2.3 The File Layout

When the EmberZNet PRO stack software is installed, the following directories are present:

- **app:** contains sample application utilities and source code.
 - **app/framework:** contains source files for Application Framework Version 2. This code is configured using the AppBuilder GUI Tool to build applications that implement ZigBee application profiles such as ZigBee Smart Energy (SE) and Home Automation (HA) profiles. This directory also includes a sample-apps folder containing InSight Configurator (isc) files to build
 - **app/nodetest:** contains pre-built hex images of the nodetest image packaged with a bootloader. Additional NodeTest binaries can be found in the build/nodetest... subdirectories for specific chip variants.
 - **app/util:** contains useful utility libraries for application development.
- **build:** contains the pre-built stack libraries. Note that when based non-application-framework based sample applications are built (such sink, sensor, sleepy-sensor, mobile-sensor), their resulting binaries are in a subdirectory under this directory.
- **em35x-ezsp-images:** contains network-coprocessor binaries and host image samples for EFR32MG and EM3xx EZSP applications
- **documentation:** contains documentation files. Open "documentation/index.htm" to see a list of all documentation files.
- **hal:** contains the HAL source code
- **stack:** contains header files necessary to build EmberZNet applications.
- **tool:** contains the AppBuilder configuration settings, bootloader images and project files and bootloader Java application.
 - **tool/appbuilder:** contains project templates, xml files and properties files for Application Builder tool.
 - **tool/image-builder:** contains image-builder upgrade file used by ZigBee OTA Bootload cluster.
 - **tool/bootloader-em{Chip Designation}:** contains the pre-built bootloader images and project files for building bootloaders with custom modifications.
 - **tool/usb-host-drivers:** contains drivers to allow customers to use USB to communicate with an EM358x product.

3 Known/Fixed Issues

3.1 Fixed Issues

3.1.1 Fixed Issues in EmberZNet 5.7.2

- Issue 180366: Incorporate latest Great Britain Companion Specification (GBCS) code/fixes into 5.7 branch.
- Issue 184213: USB can receive but will not respond after resuming from suspend.

3.1.2 Fixed Issues in EmberZNet 5.7.1

- Issue 174102: Add EZSP policy to ignore Trust Center rejoins (insecure rejoins) rather than making those devices leave
- Issue 92829: EM35x NCPs only support 32 end device children (although documentation implies 64).
- Issue 178061: USB CDC driver in EM358x HAL doesn't enumerate properly on POSIX-compliant hosts

3.1.3 Fixed Issues in EmberZNet 5.7.0

- Issue 92829: EM35x NCPs only support 32 end device children (although documentation implies 64).
- Issue 113243: Parent router isn't checking against child IDs when assigning IDs to new nodes

3.2 Known Issues

- Issue 184269: Race condition between radio receiver code and flash write/erase code could result in stack being stuck in a state where packets were no longer properly received until reboot.
- Issue 184141: Some tokens may be incorrectly set back to their default values during SimEEPROM driver startup when an upgrade is performed from SimEEPROM v1 data to SimEEPROM v2 data.
- Issue 173569: `calculateTimer()` should use `MILLISECOND_TICKS_PER_SECOND` for increased accuracy.
- Issue 162190: Fixed issue in packet reception that could cause misprocessing of non-ACK-requesting packets after an ACK with frame-pending bit set was sent.
- Issue 161027: There has been an issue reported where setting and getting the stack profile from a non-primary network does not work correctly. This issue does not affect single-network applications.
- Issue 135649: Multi-networking can cause APS frame counter confusion between networks. Workaround: Use `emberAfSecurityInitCallback` to add `EMBER_NO_FRAME_COUNTER_RESET` to `EmberInitialSecurityBitmask`.
- Issue 90870: Sleepy End Device sends Network Leave (Announce) but not Leave Response.
- Issue 122276: Sleepy-sensor NCP devices draw too much current after performing a rejoin.
- Issue 121984: Turning off NCP concentrator support does not disable all stack concentrator logic.
- Issue 119939: ZDO IEEE Request's APS ACK proxied by parent incorrectly includes long source address.
- Issue 119037: Packet-buffer.c Assert at line 352 occurs during rapid packet transmission and Partner-link-key-exchange.

- Issue 103833: Second energy scan request caught in first energy scan request knocks node out of network.
- Issue 92638: Assert when sending reports to non-responding device.
- Issue 82600: Setting a non-zero MAC Filter Table Size but no MAC Filter Table Entries causes NCP resets when joining a network
- Issue 61008: Scanning state machine (stack level or form-and-join util) can get stuck in "scanning" state indefinitely
- Issue 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.
- Issue 60970: TC link key should be used for Transport Key to rejoining devices even if decision is Send Key In Clear
- Issue 60944: EZSP-SPI NCP may become unresponsive if callbacks are received during ECC operations.
- Issue 60868: Extra retries seen on ZDO requests; ZDO response going out before APS ACK.
- Issue 60858: Sleepy broadcast payload is sometimes corrupted when relaying to child
- Issue 60774: MTORR reception by concentrator neighbor sometimes causes assert in route-discovery.c (emHandleRouteCommand).
- Issue 60757: Indirect Transaction Expiry route error not being sent when TX failure count is reached on parent router.
- Issue 90325: If you program an image onto an EM35x device using JTAG, the device must be power cycled after programing in order to for the device to enter deep sleep. If Serial Wire is used as the programming interface than a power cycle is not required after programming.
- Issue 62126: Packet stream and non-Packet debug stream should be better synchronized by ISA3 in Ember Desktop event capture.
- Issue 159525: Fast data transfer from host to EM3588 USB without acknowledgement from EM3588 sometimes causes EM3588 firmware to crash.
- Issue 106307: Nodetest calChannel command does not wake the radio to work properly.
- Issue 87700: Sometimes the MAC ACKs associated with a sleepy device's transmissions get a false CRC failure due to timing issues of PTI (packet trace interface) and the software when the radio is shutting down as the MAC ACK arrives, resulting in wrong interpretation of CRC while in fact it is correct on the chip.
- Issue 82812: Timer Registers naming inconsistent in regs.h
- Issue 71020: emberSetRadioPower no longer defined in bootloader PHY context, use emSetPhyRadioPower() as a workaround in the bootloader.
- Issue 70851: System-timer implementation precludes runtime adjustment to GPIO wake handling
- Issue 70799: Overlay of RESETINFO on CSTACK causes IAR stack overflow warning

4 Deprecated APIs and Functionality in this Release

- N/a

5 Application Framework

Please refer to the Application Framework release notes for information.

6 Intended Behavior

Here are some commonly reported issues that can be misleading behavior but function as intended:

- Broadcasts with empty payload are not heard through the loopback by the sending node
 - **Problem:** When sending a broadcast with no payload, the sending node does not get an `emberIncomingMessageHandler()` call for the broadcast loopback. Other devices in the network will receive the broadcast as expected.
 - **Workaround:** add a 1 byte payload to a broadcast that the sender expects to hear through the broadcast loopback.
 - **Issue Reference:** 59946
- When a node hears one of its own messages, the `lastHopLqi` and `lastHopRssi` parameters should have invalid values.
 - **Problem:** the last read values for LQI and RSSI are used when a device hears a message from itself
 - **Workaround:** a device should not use LQI and RSSI values when the source address is itself.
 - **Issue Reference:** 59620
- Parent could allow a device rejoining with an in-use ID to cause a node on the network to change its ID
 - **Problem:** If a child attempts a rejoin using a `shortID` that is already assigned to another device, the parent does not allow the rejoin and sends an ID conflict message causing the joined device using the same ID to change its ID.
 - **Workaround:** none, intended behavior.
 - **Issue Reference:** 62472
- A route discovery is not initiated if a source routed unicast is sent with the `EMBER_APS_OPTION_FORCE_ROUTE_DISCOVERY_APS` option
 - **Problem:** the source route option takes precedence over the route discovery option. This could be setup to return an error, but the designed behavior is to allow the source route option (the route being used is a many-to-one route) to take precedence over the route discovery option.
 - **Workaround:** none, intended behavior
 - **Issue Reference:** 62551

- Route Records are not sent to Trust Center from a router after it reboots
 - **Problem:** Route table data (such as the concentrator flag associated with route destinations that are concentrators) are stored in RAM, which is cleared upon reboot. When the router reboots, it has no Many-to-one Route [MTOR] to the Trust Center [TC], so it has no reason to send Route Records when communicating to the TC. Furthermore, routing to the TC becomes almost impossible because no known route exists; the exception to this is when the router in question is a neighbor of the TC, in which case a one-hop route is inferred by the router once it finds the destination in its neighbor table. However, even in the inferred, one-hop route case, there is no actual route entry and thus no concentrator status to be preserved yet, so no Route Record arrives for the TC to process.
 - **Workaround:** While it is possible for the TC (assuming its running our EmberZNet PRO stack) to infer a similar one-hop route back to the router in the neighboring case and thus route its replies, this won't work for routers that are non-neighbors of the TC. In order to properly correct the situation for all routers, the TC needs to be doing Many-to-one Route Requests [MTORR] periodically to advertise its status as a concentrator. Once the MTORR is received by the router, a route table entry for the TC will be created, and Route Records will resume when unicasting to the TC.
 - **Issue Reference:** 60750