

EmberZNet SDK 6.5.0 GA

Silicon Laboratories, Inc.

December 14, 2018

1 Release Highlights

1.1 Version 6.5.0 GA:

- CTUNE frequency calibration support using Simplicity Studio Commander
- Improved client and server Touchlink support for Zigbee 3.0
- Alpha automatic HW RX Diversity for 2.4GHz 15.4 PHY
- TRNG support for GB Metering on EFR32MG12
- Support for IAR stack protection feature for GB Metering
- Additional xNCP part support for EFR32MG13
- Various bug fixes

2 Using This Release

This release contains the following

- Gecko SDK 2.5.0
- Zigbee stack EmberZNet Pro v6.5.0
- Zigbee Application Framework v6.5.0
- Zigbee Sample Applications

For more information about the Silicon Labs EmberZNet Pro stack see UG103.02 - Zigbee Fundamentals. If you are a first time user, see QSG106: Getting Started with EmberZNet Pro.

2.1 Compatible Software:

The Zigbee stack should be used in conjunction 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.

This version of the SDK is compatible with the following tool chains.

- IAR Embedded Workbench for ARM (IAR-EWARM) version 8.30.1. Follow instructions in QSG106: Getting Started with EmberZNet Pro, to download from the support portal, and obtain a 30 day trial license.
- GCC (The GNU Compiler Collection) version 7.2.1 is provided with Simplicity Studio.

IAR builds in Simplicity Studio are fully supported. However, an issue has been identified when using wine to build with the IarBuild.exe command line utility or IAR Embedded Workbench GUI on macOS or Linux. This could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.

It is not recommended for customers on macOS or Linux to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

2.2 Support

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

3 Added Items

3.1 Version 6.5.0 GA:

3.1.1 New Application Plugins:

- MbedTLS Plugin

mbed TLS support is updated in order to support multiple flavors of mbed TLS configuration. The mbedtls plugin, which was previously only for multiprotocol, is split up into the mbed TLS Common and the mbed TLS Multiprotocol Configuration plugins. Also, the mbed TLS TRNG Configuration plugin is added in order to support using the TRNG hardware on parts that support the feature.

- Packet Handoff

The Packet Handoff plugin hooks directly into the Zigbee Stack and provides a mechanism to pass packets from the stack into application code through the PacketFilter API. This plugin copies packets from stack buffers into flat byte-arrays to be processed, so the plugin allows filtering of which types of packets will be handed off for processing.

- IAR Stack Protection

Stack protection may now be enabled for IAR projects by enabling the corresponding plugin and option under the "Other options" tab in Studio.

3.1.2 New Stack APIs:

- **emberSendPanIdUpdate(EmberPanId newPan)** Triggers sending an update PAN ID message with the new PanID. Returns true if sending an update was successful, false otherwise.

3.1.3 New Hardware-related Support:

- Added support for a new IEEE802.15.4 PHY that supports receive antenna diversity for customers with an external antenna switch and multiple antennas on their boards. Refer to AN1181 for further information.
- Added support for a new IEEE802.15.4 PHY that performs better in the presence of WiFi interference which can be turned on at build time. This will have slightly worse sensitivity in isolation so it is off by default. Refer to the HAL_COEX_PHY_ENABLED option in the Coex plugin in Simplicity Studio when building an application.

4 Changed Items

4.1 Version 6.5.0 GA:

- **Command Handler**

- The command handler API has been modified and replaced with a more general PacketFilter API.
- emberIncomingCommandHandler/emberOutgoingCommandHandler have been replaced with emberAfIncomingPacketFilterCallback/emberAfOutgoingPacketFilterCallback.
- EmberZigbeeCommandType has been replaced to EmberZigbeePacketType to cover a broader range of packet types.
- New EmberPacketAction return type - informs the stack what actions should be taken on the given packet.
- Previously the old Command Handler API operated on stack buffers with an index corresponding to the Command ID. The new API in conjunction with the Packet Handoff Plugin copies the packet contents into a flat array, and passes a pointer to the size of the packet allowing the size and contents of the array to be modified.

- **PAN ID Conflict Handling**

- PAN ID conflicts no longer trigger update PAN ID messages by default, but rather call emberPanIdConflictHandler().
- Use emberSendPanIdUpdate API to emulate previous behavior.
- emberPanIdConflictHandler will be called if the number of observed PANID conflicts exceeds EMBER_PAN_ID_CONFLICT_REPORT_THRESHOLD within a period of 1 minute.

- **ZLL Commissioning**

The ZLL Commissioning plugin has now been broken into 3 separate plugins, comprising of dedicated client and server side plugins and a common code plugin (ZLL Commissioning Common) in order to make it easier for a Touchlink-enabled application to support either client or server sides exclusively, or to support both. A fourth plugin, ZLL Commissioning Network, has been added, whose functionality is only used for legacy ZLL 1.0 implementations.

- **Functionality Changes**

- * The 'reset-to-factory-new' processing has been enhanced for both local (CLI) and remote (Interpan ZCL) reset commands to initialize policy and channel masks in addition to the existing functionality of leaving the network, clearing groups and scenes, and performing the factory-new plugin initialization.
- * The profile ID in the Endpoint Information command now depends on the endpoint when the Zigbee 3.0 Support option is set, rather than being hard-coded to the ZLL profile ID. This is now as per the ZCL specification. The form-network-for-router-initiator is now done unconditionally, since it is mandatory for Zigbee 3.0 compliance. Apart from this case, the Zigbee 3.0 Support option setting in the ZLL Commissioning Common plugin is only used directly in the Touchlink Scan Request and Response messages as the 'Profile Interop' bit in the 'Touchlink Information' field, and to determine the profile ID in the Scan Response, and should normally be set on, unless there is some special legacy compatibility reason to set it off.
- * Updates to the 'Rx On at Startup' mechanism to allow it to be used correctly by a dual-mode device (i.e. a device which supports touchlinking as both client/initiator and server/target), and in particular, to refuse a request to reduce the Rx On duration.

- **Plugin Changes**

- * The various options for the old ZLL Commissioning plugin have been redistributed among the new post-split plugins, except that the link initiator option has been removed completely, since it is now implied/mandatory for a touchlink client.

- * A 'secondary-channel' CLI command has been added.
 - * A new Get Radio Power callback has been added to the Network Creator plugin, in order that network creates initiated by the touchlinking client will use the power defined in the respective ZLL Commissioning Common plugin option, rather than the power defined in the Network Creator plugin option. (The callback is currently handled by the ZLL Commissioning Client plugin.)
 - * Note that due to the requirements of the Zigbee Alliance Test Tool, all Touchlink CLI commands are still prefaced 'plugin zll-commissioning', rather than 'plugin zll-commissioning-client', etc., but only the commands supported by the included plugins will be actively supported. So, for example, the command 'plugin zll-commissioning link' will do nothing unless the ZLL Commissioning Client plugin has been included in the build.
- Zigbee Alliance Test Tool Compatibility
 - * Certain legacy network code has been moved into a fourth plugin, with a view to deprecating and possibly removing it in future. However, it is currently still required by certain CLI commands used by the Zigbee Alliance Test Tool, so is retained for the time being. It now has its own stack status callback, with the main callback now going into the common code plugin.
 - A Simplicity Studio upgrade rule will by default include all four post-split ZLL Commissioning plugins to ensure functionality when upgrading to EmberZNet 6.5.0. However, it is recommended that these changes be reviewed manually in order to take advantage of the new optimizations. For example, the ZLL Commissioning Network plugin will be enabled if neither the Network Creator nor the Network Steering plugin is enabled, but this plugin is used for legacy ZLL 1.0 implementations. The provided sample applications with Touchlinking, such as Z3Light and Z3Switch, have been updated accordingly. Any application .isc file that was consuming ZLL Commissioning callbacks will be updated to consume the newer callbacks. Application callback files, which are C source files, are not modified but are fixed with a macro mapping the old callback names to the new callback names. It is suggested that customers change the older callback function names to the new ones to avoid any potential ambiguity.
- The Parent Information and Parent Additional Information tokens are now written to flash immediately when their respective values change. This differs from past releases (6.4.0 and prior) where a node that joins a network would wait 30 seconds before writing to the parent tokens. The Parent Information token is written to whenever the parent short address or EUI changes, whereas the Parent Additional Information token is written to whenever the parent keepalive support value is changed.
 - A change has been made to the Concentrator plugin whereby an Update Device message sent by a parent router will either add, modify, or delete a source route to the child depending on the contents of the Update Device message. For example, if a child joins or rejoins a parent, the concentrator will add or update the source route to that child. If the child has left the network, the concentrator will remove the appropriate source route. In relation to these changes, the API emberChangeSourceRouteHandler has been changed to include a fourth parameter which indicates if the child has left the network.
 - The default source the watchdog timer has been change from the ULFRCO to the LFRCO for EFR platforms. Accordingly, the typical timeout period has dropped approximately 2.3 percent. Previously, the device would preemptively reset after 1.536 seconds on average. It now does so after 1.5 seconds.
 - End device timeout configuration has been upgraded to use the standard Zigbee R21 values. Prior stacks used `EMBER_END_DEVICE_POLL_TIMEOUT` || `EMBER_END_DEVICE_POLL_TIMEOUT_SHIFT` to calculate a value in seconds. Now, `EMBER_END_DEVICE_POLL_TIMEOUT` is a value for the enumeration in the R21 spec. The prior default was approximately 5 minutes. The default has been changed to the 256 Minute default from R21. AppBuilder will automatically convert changed settings to the new format. However if the macros were manually overridden, this will not be detected. If the manual override causes an out of range value, a compile error will result until it is corrected.

5 Deprecated Items

5.1 Version 6.5.0 GA:

- The `EMBER_PAN_ID_CONFLICT_REPORT_THRESHOLD` macro is currently used to establish the number of observed PAN ID conflicts within a 1 minute which will trigger `emberPanIdConflictHandler` (previously PAN ID update). This threshold and its associated logic for counting the number of PANID conflicts will be deprecated in an upcoming release, resulting in `emberPanIdConflictHandler` being called on every detected PANID conflict.
- In the next release, we will eliminate the support for host-side source route table. This means all the source routing information will be only stored on the NCP side. There will be no additional source route table on the host to store more entries. In terms of source code, the `source-route-host.c` source file in the Concentrator plugin will be removed in future releases. The Source Route Table Size option in the Concentrator plugin will also be deprecated.

6 Removed Items

6.1 Version 6.5.0 GA:

- **Pre-built NCP images**

Pre-built NCP images for all platforms (found in the `ncp-images/` directory) have been removed. Customers are recommended to utilize the XNCP Framework to customize and build their own NCP images. Some pre-built applications based on XNCP Appbuilder configurations are included with the SDK.

Applications with ECC support will no longer come pre-built with the SDK. Sample apps can be configured to use Zigbee ECC and CBKE libraries if provided an ECC-163k1/283k1 library implementation.

Below is a list of pre-built images removed this release. Next to each image is the corresponding XNCP Appbuilder configuration to use as a starting point when building an application that matches the image.

- `ncp-spi-use-with-ezsp-spi-btl-NCP_REV` -> `ncp-spi`
- `ncp-uart-rts-cts-use-with-serial-uart-btl-NCP_REV` -> `ncp-uart-hw`
- `ncp-uart-xon-xoff-use-with-serial-uart-btl-NCP_REV` -> `ncp-uart-sw`
- `ncp-spi-use-with-ezsp-spi-btl-ecc-NCP_REV` -> `ncp-spi + ECC`
- `ncp-uart-use-with-serial-btl-ecc-NCP_REV.s37` -> `ncp-uart-hw + ECC`
- `ncp-uart-fifo-use-with-serial-btl-ecc-NCP_REV` -> `ncp-uart-sw + ECC`
- `em260-uart-fifo-ecc-tokens` -> `ncp-uart-sw + ECC`

7 Fixed Issues

7.1 Version 6.5.0 GA:

- 185606 AppBuilder now supports overlapping manufacturer codes for Zigbee Cluster Library.
- 192041 An issue was fixed where end devices would issue Route Request messages. The stack libraries for end devices now no longer issues Route Requests. Additionally, routers and coordinators that receive a Network Status: Router Error message will remove routing table entries whose destination match the destination-in-error of the Route Error message.
- 249836 An edge case was fixed on EFR32 that could cause resets of the watchdog timer to be lost if sleep was entered before they completed.

- 267722 The transient link key processing has been updated to allow more simultaneous joiners. Previously, the transient keys were stored in a single buffer, and the 255 byte buffer size limit restricted it to 7 joiners. Now, the only restriction is the number of available buffers.
- 282979 Service discovery now supports as many as four simultaneous issued discovery requests (previously limited to one request outstanding at a time).
- 330387 An issue regarding EM3xx stack corruption after waking up from sleep was fixed.
- 331428 An issue has been fixed where `EMBER_TABLE_FULL` was always being returned when calling `emberAddOrUpdateKeyTableEntry` when using NVM3.
- 336656 SubGHz BRD4163a binaries (notably `nodetest`) were mistakenly removed from the SDK. They have now been restored.
- 336696 When enabling Minimum Wake-Time Idle-Sleep plugin option, if the device has not been awake for long enough it will remain active, not sleeping or idling, until the end of the minimum wake-time duration
- 340003 An issue triggering a watchdog reset when adding duplicate transient link keys was fixed in EmberZNet 6.4.0 (336177)
- 342334 Several Coverity issues have been resolved.
- 342464 An issue has been fixed where the EFR32 serial driver state got out of sync with the actual contents of the buffer. Previously, partial or old data was being returned.
- 342842 An issue has been fixed where the Update TC Link Key plugin would send a Node Descriptor, Request Key, or Verify Key message and hang until a response is received. The Update TC Link Key plugin now attempts retries when a Base Device Behavior-specified timeout is reached (5 seconds). The Update TC Link Key plugin now calls the Zigbee Key Establishment handler in the event where the Trust Center fails to respond completely.
- 343998 Fixed an issue with a failing Network Key Update in Unicast mode.
- 344501 A note has been added to the Network Steering plugin APIs warning that `emberAfPluginNetworkSteeringStart` should not be called from a stack status callback. The Network Steering code acts when its own stack status callback is called, and thus its start API should not be called in the same context.
- 345199 Corrected the handling of Level Control cluster Execute-If-Off behavior based on Options attribute setting and override fields in certain cluster commands.
- 345466 A misconfiguration of `mbedtls` which caused some abnormally long bluetooth operations has been rectified for dynamic multiprotocol.
- 346994 A bug preventing endpoints in Host-NCP mode from being consistently enabled and disabled has been fixed.
- 347418 Several bugs with the 'Rx On at Startup' mechanism have been fixed, including broken 'Rx On' expiry on router targets (Rx On never expires), incorrect handling of 'Rx On' period extensions, and incorrect handling of an interruption to an 'Rx On' period due to a reset. Note that pending a host-side implementation of `emberZllCancelRxOnWhenIdle()`, `emberZllSetRxOnWhenIdle()` with `duration == 0` will perform an Rx On operation cancel, and will now be accepted even if `rxAlwaysOn` is set.
- 349608 The EZSP-SPI flavor of the Gecko bootloader generally used in SPI NCP systems now supports upgrading its own second stage. See UG266 for further information on the Gecko bootloader. Note that this operation generally overwrites part of the application running on top of the bootloader. In such a scenario, the system host must be prepared to communicate with the (newly installed) bootloader directly and upload an application image to restore normal functionality.

- 350133 Serial port hardware flow control does not work on macOS Mojave. The CRTSCTS setting (a terminal control option for hardware flow control) is not set by an attempt to enable it on Mojave, which was not the case with previous versions of macOS. As a workaround, software flow control is enabled if hardware flow control fails to be enabled.
- 354325 emberTrustCenterJoinHandler is no longer incorrectly called when the TC receives a Leave command with the request bit cleared and the rejoin bit set.
- 355092 Fixes a regression where a touchlink network update between an initiator and a target on a centralized network can cause the target to lose its security setup. Also fixes a non-compliance bug, where a touchlink initiator will update its logical channel if a target on the same network is on a different channel, but has the same network update id; it should only do so if the id in the message is greater than its own. This bug is also fixed for a touchlink target on receipt of a network update request from an initiator on the same network.
- 356233 An issue was fixed that was causing applications to attempt to set the priority for fixed-priority interrupts.
- 357966 An issue has been fixed where the variable emAllowTcRejoinsUsingWellKnownKeyTimeoutSec was undefined when compiling the Network Creator Security plugin for an SoC application.
- 360472 Bug fixed to account for EZSP legacy frame ID and extended frame ID fields, to present the correct frame-identifying information in the EZSP trace console output.

8 Open Issues

8.1 EmberZNet Open Issues:

- 60757 Indirect Transaction Expiry route error not being sent when TX failure count is reached on parent router.
- 60774 MTORR reception by concentrator neighbor sometimes causes assert in route-discovery.c (emHandleRouteCommand).
- 60858 Sleepy broadcast payload is sometimes corrupted when relaying to child
- 60868 Extra retries seen on ZDO requests; ZDO response going out before APS ACK.
- 60944 EZSP-SPI NCP may become unresponsive if callbacks are received during ECC operations.
- 60970 TC link key should be used for Transport Key to rejoining devices even if decision is Send Key In Clear
- 60975 EZSP_VALUE_TOKEN_STACK_NODE_DATA, EZSP_VALUE_UART_SYNCH_CALLBACKS, EZSP_VALUE_MAXIMUM_INCOMING_TRANSFER_SIZE, and EZSP_VALUE_MAXIMUM_OUTGOING_TRANSFER_SIZE ValueIDs are writable but not readable.
- 61008 Scanning state machine (stack level or form-and-join util) can get stuck in "scanning" state indefinitely
- 66508 Framework should avoid sending unicast loopback messages with APS security since stack doesn't support this
- 66785 Messaging Client plugin should differentiate between Cancel Msg command and timed out / replaced message
- 66786 "zcl ota server reload" doesn't properly reload image info when using OTA Simple Storage plugin
- 66944 Duplicate Key Confirm Response message can lock up KE plugin state machine

- 70799 Overlay of RESETINFO on CSTACK causes IAR stack overflow warning
- 82600 Setting a non-zero MAC Filter Table Size but no MAC Filter Table Entries causes NCP resets when joining a network
- 86948 Fragmented messages can be passed to the application with old data from the rxFragmentedPacket buffer
- 92147 ZLL Scan Response Should Be Sent at Power 0 rather than last-used power level
- 103833 Second energy scan request caught in first energy scan request knocks node out of network.
- 106307 Nodetest calChannel command does not wake the radio to work properly.
- 119037 Packet-buffer.c Assert at line 352 occurs during rapid packet transmission and Partner-link-key-exchange.
- 119828 ota-client.c does not use the server EUI64 in Partner Link Key Exchange.
- 119939 ZDO IEEE Request's APS ACK proxied by parent incorrectly includes long source address.
- 121984 Turning off NCP concentrator support does not disable all stack concentrator logic.
- 126087 Sleepy end device would return NO_LOCAL_RESOURCES when a coordinator initiates key establishment with it in Multi-networking.
- 135649 Multi-networking can cause APS frame counter confusion between networks. Workaround: Use emberAfSecurityInitCallback to add EMBER_NO_FRAME_COUNTER_RESET to EmberInitialSecurityBitmask.
- 158598 OTA Client plugin's Bootloading Message Timeout doesn't account for Short Poll Interval being longer than the default and may spuriously time out messages before the sleepy client has had a chance to poll for the response.
- 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.
- 180028 OTA Client Plugin should send ZCL Default Response to messages with wrong Mfg Code
- 201417 Adding GPIO Sensor Interface plugin to an EM358x project results in error: "identifier "GPIO_SENSOR_IRQ" is undefined".
- 213424 Problem with ZLL Devices responding to a multicast addScene when they should not.
- 229938 ZLL devices are sending ZLL device information frames with the 0x0104 HA/Z3 common profile ID instead of 0xC05E ZLL profile ID.
- 235222 Virtual UART (VUART) works on Ethernet but does not work over USB on WSTK.
- 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.
- 261670 Harden the ZLL touchlink process to mitigate malicious attacks
- 266341 Z3 Light sample app has two endpoints that support similar cluster commands, so duplicate responses may be generated for certain commands.

- 271644 A device that performs a classic join to a legacy ZLL gateway may eventually leave the network on its own initiative.
- 274414 AFv2 blank app project template for SOC and Host should use Z3 defaults, not ZHA defaults
- 278063 Smart Energy Tunneling plugins have conflicting treatment/usage of address table index
- 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.
- 281832 Green Power Common plugin incorrectly formats groupList and groupListCount parameters of GP Pairing Configuration frame.
- 289569 network-creator plugin power level picklist doesn't offer full range of supported values for EFR32
- 295498 UART reception sometimes drops bytes under heavy load in Zigbee+BLE DMP use case
- 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.
- 303943 ezspGetStandaloneBootloaderVersionPlatMicroPhy only reports the major and minor version, excluding the build number, when used with the Gecko Bootloader
- 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.
- 315661 Seeding of the random number generator on EM3xx can take ~400 milliseconds.
- 316212 The UART XMODEM standalone gecko bootloader with the legacy ebl parser can fail to return to the bootloader menu if the XMODEM transfer is interrupted unexpectedly (such as due to a power cycle). If the ebl file format is a requirement, then a recovery pin must be implemented to mitigate the risk.
- 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.
- 337431 A bug keeping a sleepy end device awake during the short poll interval has been resolved
- 338151 Initializing NCP with a low packet buffer count value may cause corrupt packets.
- 341629 Z3 end devices can't join the network via ZHA key if joining through parent with only 1 free child table entry. (Entry created during install-code-based key joining will persist after join failure, causing target parent to be at capacity until old entry times out.)
- 345167 Sleepy end devices occasionally not sending APS ACK for received APS unicasts polled from parent.

- 355239 CREATOR codes for tokens are normally only required for applications that use Simulated EEPROM version 1 or 2 plugins or the Simulated EEPROM version 2 to NVM3 upgrade plugin. However, certain token APIs included in this release requires a CREATOR code to be defined for all tokens. Workaround: All tokens must have a CREATOR code defined, even when not using the Simulated EEPROM version 1 or 2 plugins or the Simulated EEPROM version 2 to NVM3 upgrade plugin.
- 345367 A bug has been identified which results in periods of awokeness in sleepy end devices when many APS ACK required messages are sent in quick succession.
- 346541 Attribute reports may mix attribute values with incorrect endpoints when reporting to a destination that is offline
- 362149 The size of NVM3 cache is not automatically adjusted based on the number of objects used by the application. An improperly sized cache may cause performance degradation and other undesirable effects. If you encounter problems with the default value, we recommend you change the cache size option in the NVM3 plugin such that it can hold entries for all NVM3 objects.

9 Intended Behavior

10 Documentation Changes

11 History