

s132_nrf52_5.0.0 release notes

Introduction to the s132_nrf52 release notes

About the document

The release notes describe the changes in the s132_nrf52 v5 from version to version.

The release notes are intended to list all relevant changes in a given version. They are kept brief to make it easy to get an overview of the changes. More details regarding changes and new features can be found in the s132_nrf52 migration document (normally available for major releases only).

Issue numbers in parentheses are for internal use and should be disregarded by the customer.

Copyright (c) Nordic Semiconductor ASA. All rights reserved.

s132_nrf52_5.0.0-2.alpha

The main new features of this alpha version, compared to the 5.0.0-1.alpha version, are application control of the Data Length Update and PHY Update Procedures, SoftDevice configuration API extensions, support for Network Privacy Mode, support for multiple peripheral connections, support for up to 20 connections in total, and configuration of individual links including per link ATT_MTU configuration.

Notes:

- This release has changed the Application Programmer Interface (API) from the 5.0.0-1.alpha release. This requires applications to be recompiled.
- The memory requirements of the SoftDevice have changed.

SoftDevice properties

- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **132 kB** (0x21000 bytes).
 - RAM: **5.12 kB** (0x1478 bytes). This is the minimum required memory. Actual requirements are dependent upon the configuration chosen at `sd_ble_enable()` time.
 - Call stack: The SoftDevice uses a call stack combined with the application. The worst-case stack usage for the SoftDevice is **1.48 kB** (0x5EC bytes). Application writers should ensure that enough stack space is reserved to cover both worst-case SoftDevice call stack usage combined with worst-case application call stack usage.

New functionality

- SoftDevice
 - The SoftDevice now supports sleep clock accuracy values less than 20 ppm as a peripheral (DRGN-8158).
 - The RC oscillator accuracy can now be set to any of the defined `NRF_CLOCK_LF_ACCURACY` values, and there is no default anymore. In other words, the `nrf_clock_lf_cfg_t::accuracy` parameter now has the same functionality when used with the RCOSC clock source as with the XTAL clock source (DRGN-8666).
- BLE
 - Support for 20 links in total with freely selectable role (Central/Peripheral) for each link (DRGN-7102, DRGN-7152, DRGN-7848).
 - The BLE bandwidth configuration and application packet concept has been replaced with per link configurable:
 - Event length (DRGN-7858)
 - Write without response queue size (DRGN-7488, DRGN-7858)
 - Handle Value Notification queue size (DRGN-7487, DRGN-7858)
 - The GPIO pin to toggle can now be the same for PA and LNA (DRGN-8354).
- LL
 - The SoftDevice can be configured to disable and enable slave latency (DRGN-8305). This allows the application to override the slave latency set by the master.
 - The SoftDevice can be configured to not disconnect if the peer initiates parallel version and feature exchange procedures (DRGN-8306).
 - Support for Network Privacy Mode (DRGN-8658)
- GAP
 - The event length (i.e. the time set aside on every connection interval) can now be configured per link by the application (DRGN-7858).
 - The application is given control of the Data Length Update Procedure. The application can initiate the Data Length Update Procedure and has to respond when initiated by the peer (DRGN-8297).
 - The application is given control of the PHY Update Procedure. The application can initiate the PHY Update Procedure and has to respond when the procedure is initiated by the peer (DRGN-8473).
 - GAP option `BLE_GAP_OPT_PREFERRED_PHYS_SET` to set the default PHY preferences for the SoftDevice is removed (DRGN-8473).
- GATT
 - The maximum ATT_MTU can now be configured per link by the application (DRGN-7858).
- GATTC
 - The application packet concept has been replaced with a dedicated transmission queue for Write without responses. Also, the `BLE_EVT_TX_COMPLETE` event has been replaced with `BLE_GATTC_EVT_WRITE_CMD_TX_COMPLETE`. Write without response queue size can now be configured per link by the application (DRGN-7488, DRGN-7858).
- GATTS
 - The application packet concept has been replaced with a dedicated transmission queue for Handle Value Notifications. Also, the `BLE_EVT_TX_COMPLETE` event has been replaced with `BLE_GATTS_EVT_HVN_TX_COMPLETE`. Handle Value

Notification queue size can now be configured per link by the application (DRGN-7487, DRGN-7858).

Using 2 Mbps

The SoftDevice provides a new SV call `sd_ble_gap_phy_update()` and two new events, `BLE_GAP_EVT_PHY_UPDATE_REQUEST` and `BLE_GAP_EVT_PHY_UPDATE`, to support initiating or responding to a PHY Update procedure and to be notified about incoming peer initiated PHY Update procedures and link PHY updates. Upon receiving a `BLE_GAP_EVT_PHY_UPDATE_REQUEST`, the application needs to respond with an `sd_ble_gap_phy_update()` SV call. For more information, see API documentation.

This alpha version of the SoftDevice supports connection establishment using the 1 Mbps PHY and then changing either the transmitting PHY or the receiving PHY (asymmetric link configuration), or both (symmetric link configuration) to use the 2 Mbps PHY. The PHYs can be changed using the abovementioned SV call.

Link Layer encryption and long data packet payload (up to 251 octets) are supported on both 1 Mbps and 2 Mbps PHYs.

Changes

- SoftDevice
 - The `sd_power_ramon_set()`, `sd_power_ramon_clr()`, and `sd_power_ramon_get()` SoftDevice APIs have been replaced with `sd_power_ram_power_set()`, `sd_power_ram_power_clr()`, and `sd_power_ram_power_get()` (DRGN-8117). Therefore, the application now has access to the registers `RAM[x].POWER` instead of the deprecated `RAMON/RAMONB`.
 - SWI3 is no longer reserved for use by the SoftDevice and is available for the application (DRGN-8367).
 - Interrupt priority 5 is now available to the application (DRGN-8853).
- BLE
 - More pointers have been defined as `const` in the BLE API allowing the application to put more data into flash instead of RAM if desired (DRGN-6133).
 - Configuration parameters passed to `sd_ble_enable()` have been moved to the SoftDevice configuration API (DRGN-8107).
- Documentation
 - The Message Sequence Charts (MSCs) for LL Data Length Update Procedure have been corrected, extended, and improved (DRGN-8722).
 - Improved documentation for `sd_ble_gap_adv_start()` (DRGN-8799)

Bug fixes

- SoftDevice
 - Fixed an issue where `sd_ble_enable()` may corrupt up to 8 bytes above the returned `app_ram_base` when the SoftDevice is configured with 0 Peripheral roles and 0 Central roles (DRGN-8802).
 - The `sd_power_pof_threshold_set` API has been fixed to support all the new levels that were introduced in nRF52 (DRGN-8348).
 - Fixed an issue where the SoftDevice could trigger a BusFault when forwarding a HardFault to the application (DRGN-8604).
 - Fixed an issue where scanning or advertising with timeout greater than 256 seconds and having two host protocol timers running at the same time might lead to delayed timeouts (DRGN-7804).
 - `sd_softdevice_enable()` now returns an error code if called with `fault_handler` set to `NULL` or to an invalid function pointer. If the application returns from the `fault_handler` function, the SoftDevice will do an `NVIC_SystemReset()` (DRGN-7122).
 - It is no longer required to clear `INTENSET` for `TIMER0` before the timeslot ends if the application uses `TIMER0` inside a timeslot scheduled with the Radio Timeslot API (DRGN-7776).
 - The `SVCALL` macro can now be used with the GCC C++ compiler as well (DRGN-8028).
- BLE
 - Several Doxygen documentation errors have been corrected (DRGN-7386, DRGN-7853, DRGN-8136).
- LL
 - Fixed an issue where using more than eight links and receiving a lot of data concurrently could lead to undefined behavior (DRGN-8433).
 - Fixed an issue where the SoftDevice could assert if scan parameters are updated after the scanner has accepted a new LE connection (DRGN-8635).
 - Fixed an issue where using encryption on multiple master links at the same time could cause an assert (DRGN-8532).
 - Fixed an issue where the SoftDevice would only be able to send two packets per connection event after a Data Length Update Procedure to a LL Data Channel PDU payload size of more than 34 bytes (DRGN-8392).
 - Fixed an issue where a connection parameter update from a short connection interval to a longer connection interval when using long ATT MTUs could lead to reduced bandwidth (DRGN-8427).
 - Fixed an issue where the controller completed a procedure when it received an `LL_UNKNOWN_RSP` without checking if it was the expected procedure that returned the error opcode (DRGN-7999).

- The SoftDevice no longer rejects `LL_LENGTH_REQ` and `LL_LENGTH_RSP` with parameters which are out of range according to Bluetooth 4.2 specification (DRGN-7872).
- Fixed an issue where bit errors in the length field of an encrypted packet caused the packet to be interpreted as longer than was sent by the peer (DRGN-7898). This issue could have manifested in the following ways:
 - SoftDevice memory buffer corruption which could lead to an assert or incorrect behavior.
 - SoftDevice may send a packet with an incorrect MIC field leading to a disconnect from the peer.
- The SoftDevice no longer accepts `LL_PHY_REQ` and `LL_PHY_RSP` with empty TX and/or RX PHY fields (DRGN-7950).
- GAP
 - Fixed an issue where the `BLE_GAP_DATA_LENGTH_AUTO` value for `p_dl_params->max_tx_octets` and `p_dl_params->max_rx_octets` in `sd_ble_gap_data_length_update()` might not work as expected on connections using a configuration with configured event length of 2, 3, or 4 (DRGN-8779).
 - Fixed an issue where the `conn_handle` parameter in the event `BLE_GAP_EVT_DATA_LENGTH_UPDATE_REQUEST` was not populated correctly (DRGN-8749).
 - Fixed an issue where the SoftDevice would assert when `sd_ble_gap_device_identities_set()` was called while advertiser is running (DRGN-8634).
 - Two missing Advertising Data Types have been added: `BLE_GAP_AD_TYPE_LESC_CONFIRMATION_VALUE` (0x22) and `BLE_GAP_AD_TYPE_LESC_RANDOM_VALUE` (0x23) (DRGN-8101).
 - `sd_ble_gap_connect()` now always stops the scanner (DRGN-7679).
 - Fixed an issue where `sd_ble_gap_conn_param_update()` called in peripheral role may in some cases return `NRF_ERROR_BUSY` for 30 seconds after the previous procedure initiated by that call was completed (DRGN-8577).
- GATT
 - It is no longer possible to issue a write command if the write command queue size is set to 0 on the config API (DRGN-8353).
- GATTS
 - It is no longer possible to issue an HVN if the HVN queue size is set to 0 on the config API (DRGN-8353).
- Documentation
 - Fixed documentation for `sd_ble_gap_addr_set()` and `sd_ble_gap_privacy_set()` (DRGN-8624).
 - Fixed documentation for `sd_ble_adv_start()` (DRGN-8624).
 - Fixed documentation for `sd_ble_gap_privacy_get()` (DRGN-8896).

Limitations

- SoftDevice
 - If Radio Notifications are enabled, flash write and flash erase operations initiated through the SoftDevice API will be notified to the application as Radio Events (FORT-809).
 - Synthesized low frequency clock source is not tested or intended for use with the BLE stack.
 - Applications must not modify the `SEVONPEND` flag in the `SCR` register when running in priority levels higher than 6 (priority level numerical values lower than 6) as this can lead to undefined behavior.
- GATTS
 - To conform to the Bluetooth specification, there shall not be a secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906).

Known Issues

- SoftDevice
 - If Connection Event Length Extension is enabled, the Radio Notification may be suppressed between connection events (DRGN-7687).
 - Calling `sd_ble_gap_sec_params_reply()`, `sd_ble_user_mem_reply()`, or `sd_ble_gatts_rw_authorize_reply()` more than 6 times without pulling events in between may in some cases lead to link disconnect (DRGN-8627).
- GATTS
 - When `BLE_EVT_USER_MEM_REQUEST` event is pulled by the application, incoming packet processing may in some cases be delayed until the application replies with the `sd_ble_user_mem_reply()` call (DRGN-8595).
 - The value of the attribute in `BLE_GATTS_EVT_RW_AUTHORIZE_REQUEST` event corresponding to the first Prepare Write Request on a link with heavy traffic may get corrupted if the application delays the pulling of SoftDevice events (DRGN-8595).

s132_nrf52_5.0.0-1.alpha

The s132 is a SoftDevice for the nRF52832 chip.

These release notes list the changes and differences from **s132_nrf52_3.0.0**.

Notes:

- This is a major release which has changed the Application Programming Interface (API), requiring applications to be recompiled.
- The memory requirements of the SoftDevice have changed.

SoftDevice properties

- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **128 kB** (0x20000 bytes).
 - RAM: **6.43 kB** (0x19C0 bytes) (minimum required memory - actual requirements are dependent upon the configuration chosen at `sd_ble_enable()` time).
 - Call stack: The SoftDevice uses a call stack combined with the application. The worst case stack usage for the SoftDevice is **1.54 kb** (0x624 bytes) (s132_nrf52_3.0.0 has 0x600 bytes of worst case stack usage). Application writers should ensure that enough stack space is reserved to cover both worst case SoftDevice call stack usage combined with worst case application call stack usage.

New functionality

- LL
 - Support for transmitting and receiving on the 2 Mbps PHY has been added (DRGN-7552).

Using 2 Mbps

The SoftDevice provides a new GAP option `BLE_GAP_OPT_PREFERRED_PHYS_SET`, a new SV call `sd_ble_gap_phy_request()`, and a new event, `BLE_GAP_EVT_PHY_UPDATE` to support the new PHY. Please read the API documentation for more details about these.

This alpha version of the SoftDevice supports connection establishment using the 1 Mbps PHY and then changing either the transmitting PHY or the receiving PHY (asymmetric link configuration), or both (symmetric link configuration) to use the 2 Mbps PHY. The PHYs can be changed using the above mentioned SV call.

Link Layer encryption and long data packet payload (up to 251 octets) are supported on both 1 Mbps and 2 Mbps PHYs.

Bug fixes

There are no bug fixes in this release.

Limitations

- SoftDevice
 - If Radio Notifications are enabled, flash write and flash erase operations initiated through the SoftDevice API will be notified to the application as Radio Events (FORT-809).
 - Synthesized low frequency clock source is not tested or intended for use with the BLE stack.
 - Applications must not modify the `SEVONPEND` flag in the `SCR` register when running in priority level 1 as this can lead to undefined behavior.
 - If the application uses `TIMER0` inside a timeslot (scheduled with the Radio Timeslot API), `INTENSET` for `TIMER0` must be cleared before the timeslot ends (DRGN-7776).
- LL
 - The peripheral role has priority over the central role when it comes to keeping the links alive.
 - For 2 Mbps, see the section "Using 2 Mbps" above.
- GAP
 - A broadcaster **and** a scanner cannot both be active if there are 8 connections established (DRGN-6543).
- GATTS
 - To conform to the Bluetooth specification there shall not be a secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906, DRGN-2260).

Known Issues

- If `sd_softdevice_enable()` is called with `fault_handler` set to `NULL`, an invalid function pointer, or a pointer to a returning function, the behavior will be undefined (DRGN-7122).
- If Connection Event Length Extension is enabled, the Radio Notification may be suppressed between connection events (DRGN-7687).
- When `sd_ble_gap_connect()` returns an error code, the scanner may be stopped (DRGN-7679). To ensure the scanner is in a known state, `sd_ble_gap_scan_stop()` should be used to stop the scanner when `sd_ble_gap_connect()` returns an error code.

