s122_nrf52 release notes

Introduction to the s122_nrf52 release notes

About the document

These release notes describe the changes in the s122_nrf52 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 may be found in the s122_nrf52 migration document (normally available for major releases only).

This document may be updated for an already released version of SoftDevice. The changes will be tagged with "Update X", where X is a number incremented each time the document has been revised.

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

Contents

s122_nrf52_8.1.1

This is a production release that contains bug fixes to the 8.1.0 release.

Notes:

- The release notes list changes since s122_nrf52_8.1.0.

- This SoftDevice is binary compatible with the s122_nrf52_8.1.0, and memory requirements have not changed. Applications are therefore not required to be recompiled.

- The format of the release notes has changed. The release notes for previous releases, included in this document, have also been updated to the new format.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52811, nRF52820 and nRF52833.

- This SoftDevice contains the Master Boot Record (MBR) version 2.5.0 (DRGN-11287).

    - This MBR version is compatible with previous MBR versions.

- The combined MBR and SoftDevice memory requirements for this version are as follows:

    - Flash: **112.0 kB** (0x1C000 bytes).

    - RAM: **4.7 kB** (0x12E0 bytes). This is the minimum required memory. The actual requirements depend on 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 **1.5 kB** (0x600 bytes). Application writers should ensure that enough stack space is reserved to cover the worst-case SoftDevice call stack usage combined with the worst-case application call stack usage.

- The Firmware ID of this SoftDevice is 0x0112.

Bug fixes

- SoftDevice

    - Fixed an issue where accessing the restricted peripherals outside their interrupt handlers in a timeslot caused a HardFault (DRGN-14620).

    - Fixed an issue where the QoS Channel Survey would only listen on one channel (DRGN-14751).

    - Updated links to Bluetooth.com in the API documentation (DRGN-14697).

- LL

    - Fixed an issue in central role which could cause a disconnect if there were scheduling conflicts while doing a control procedure with an instant (DRGN-11222).

Limitations

- SoftDevice

    - Synthesized low frequency clock source is not tested or intended for use with the Bluetooth LE 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.

- If the scanner is configured with a scan window larger than 16 seconds, the scanner will truncate the scan window to 16 seconds (DRGN-10305).

- The SoftDevice may generate several events when connected, based on peer actions, meaning without previous action from the application. The `BLE_GAP_EVT_PHY_UPDATE_REQUEST` event, for instance, is generated when a connected peer sends a Phy Update Request, even when an application does not include logic to change PHY. There are several such events that may require action from an application if they are received. For more information, see the `sd_ble_enable()` API in SoftDevice.

- GATT

  - To conform to the Bluetooth Core Specification v5.2, there shall be no secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906).

Known Issues

- SoftDevice

  - The `BLE_GAP_EVT_SEC_INFO_REQUEST` event will not report the identity address of the peer to the application. A workaround is to do a mapping of the connection handle to the peer's identity address (DRGN-10340).

- GATTC

  - The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event if `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()` is called when a Primary Service Discovery by Service UUID is already ongoing. When the application has called `sd_ble_gattc_primary_services_discover()`, it should wait for the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event before calling `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()` (DRGN-11300).

- LL

  - If the application adds an all zeroes IRK with the `sd_ble_gap_device_identities_set()`, it will be treated as a valid entry in the device identity list. An all zeroes IRK is invalid and must not be added (DRGN-9083).

s122_nrf52_8.1.0

The main new feature of this version compared to the s122_nrf52_8.0.0 version is the efficient discovery of 128-bit UUIDs.

Notes:

- The release notes list changes since s122_nrf52_8.0.0.

- This SoftDevice is binary compatible with the s122_nrf52_8.0.0, and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice Properties

- This SoftDevice variant is compatible with nRF52811, nRF52820 and nRF52833.

- This SoftDevice contains the Master Boot Record (MBR) version 2.5.0 (DRGN-11287).

  - This MBR version is compatible with previous MBR versions.

- The combined MBR and SoftDevice memory requirements for this version are as follows:

  - Flash: **112 kB** (0x1C000 bytes)

  - RAM: **4.8 kB** (0x12E0 bytes). This is the minimum required memory. The actual requirements depend on 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 **1.5 kB** (0x600 bytes). Application writers should ensure that enough stack space is reserved to cover the worst-case SoftDevice call stack usage combined with the worst-case application call stack usage.

- The Firmware ID of this SoftDevice is 0x00F6.

New Features

- GATTC

  - 128-bit UUIDs can be discovered more efficiently by enabling the `BLE_GATTC_OPT_UUID_DISC` option. This option enables the automatic insertion of discovered 128-bit UUIDs to the Vendor Specific UUID table (DRGN-9653).

Changes

- SoftDevice

  - `sd_ble_gap_tx_power_set()` is extended to support a bigger range of radio output power levels. The highest possible radio output power level is now +8dBm (DRGN-14449).

Bug Fixes

- SoftDevice

  - Fixed an issue where for idle periods shorter than 9 ms, there was an additional power consumption of 0.5 to 1 mA. For example that meant an additional average current consumption of about 600 μA for 7.5 ms interval connections (DRGN-14154/DRGN-14152) .

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 (DRGN-5197/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.

  - If the scanner is configured with a scan window larger than 16 seconds, the scanner will truncate the scan window to 16 seconds (DRGN-10305).

  - The SoftDevice may generate several events when connected, based on peer actions, meaning without previous action from the application. The BLE_GAP_EVT_PHY_UPDATE_REQUEST event, for instance, is generated when a connected peer sends a Phy Update Request, even when an application does not include logic to change PHY. There are several such events that may require action from an application if they are received. For more information, see the sd_ble_enable() API in SoftDevice.

- GATT

  - To conform to the Bluetooth Core Specification v 5.2, there shall be no secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906).

Known Issues

- SoftDevice

  - **Update 1:** When running on nRF52811 which does not have the USB peripheral, using `sd_power_usb*` APIs can lead to undefined behavior (DRGN-12720).

  - **Update 1:** When running on nRF52811 which does not support TX power larger than 4 dBm, using the `sd_ble_gap_tx_power_set()` API to set a TX power larger than 4 dBm can lead to undefined behavior (DRGN-14827).

  - **Update 1:** When running on nRF52811, the `sd_protected_register_write()` API is not supported and will always return `NRF_ERROR_INVALID_ADDR` (DRGN-14688).

- GATTC

  - The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event if `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()` is called when a Primary Service Discovery by Service UUID is already ongoing (DRGN-11300). When the application has called `sd_ble_gattc_primary_services_discover()`, it should wait for the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event before calling `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()`.

- LL

  - If the application adds an all zeroes IRK with the `sd_ble_gap_device_identities_set()`, it will be treated as a valid entry in the device identity list. An all zeroes IRK is invalid and must not be added (DRGN-9083).

s122_nrf52_8.0.0

The main new features of this version compared to the s122_nrf52_8.0.0-10.alpha version are synchronous scheduling of central links, scanner, initiator, and QoS channel survey.

SoftDevice Properties

- This SoftDevice variant is compatible with nRF52820 and nRF52833.

- This SoftDevice contains the Master Boot Record (MBR) version 2.5.0 (DRGN-11287).

    - This MBR version is compatible with previous MBR versions.

- The combined MBR and SoftDevice memory requirements for this version are as follows:

    - Flash: **112 kB** (0x1C000 bytes)

    - RAM: **4.8 kB** (0x12e0 bytes). This is the minimum required memory. The actual requirements depend on 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 **1.5 kB** (0x600 bytes). Application writers should ensure that enough stack space is reserved to cover the worst-case SoftDevice call stack usage combined with the worst-case application call stack usage.

- The Firmware ID of this SoftDevice is 0x00EA.

New Features

- SoftDevice

    - An API to turn off event length validation. The SoftDevice still allocates the required time for the active PHY, which may cause scheduling conflicts. For example, this allows connecting on 1 Mbps PHY while allocating an event length for 2 Mbps PHY when the application intends to use 2 Mbps PHY (DRGN-13902/DRGN-13908).

Changes

- SoftDevice

    - Bluetooth Core Specification v5.2 qualified (DRGN-11264).

    - The VersNr field in the LL_VERSION_IND packet now contains the value 0x0B to indicate compatibility with Bluetooth Core Specification v5.2 (DRGN-13606).

    - References to SoC Errata are added to the documentation of all the events and APIs which report RSSI and should be observed if using RSSI measurements (DRGN-13544).

    - Reduced the overhead for an earliest possible Radio Timeslot (DRGN-10440).

    - Central links, scanner, initiator and QoS channel survey are now scheduled synchronously. Previously, this was the case only for central links and the scanner (DRGN-13866).

- LL

    - Unsolicited PING_RSP packets are ignored instead of closing the connection (DRGN-12792).

    - If the Host sets 1 Mbps PHY in the TX_PHYs or RX_PHYs parameters and the event length is not sufficient, the BTLE_CONNECTION_LIMIT_EXCEEDED error code is returned. Previously, it was not possible to set an insufficient event length for 1 Mbps PHY (DRGN-14039).

Bug Fixes

- SoftDevice

  - Fixed an issue where the SoftDevice could slightly misplace events when resolving scheduling conflicts between Timeslot events and higher priority radio events (DRGN-13571).

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 (DRGN-5197/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.

  - If the scanner is configured with a scan window larger than 16 seconds, the scanner will truncate the scan window to 16 seconds (DRGN-10305).

  - The SoftDevice may generate several events when connected, based on peer actions, meaning without previous action from the application. The `BLE_GAP_EVT_PHY_UPDATE_REQUEST` event, for instance, is generated when a connected peer sends a Phy Update Request, even when an application does not include logic to change PHY. There are several such events that may require action from an application if they are received. For more information, see the sd_ble_enable() API in SoftDevice.

- GATT

  - To conform to the Bluetooth Core Specification v 5.2, there shall be no secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906).

Known Issues

- SoftDevice

  - For idle periods shorter than 9 ms, there will be an additional power consumption of 0.5 to 1 mA. For example this means an additional average current consumption of about 600 µA for 7.5 ms interval connections (DRGN-14154/DRGN-14152) .

- GATTC

  - The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event if `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()` is called when a Primary Service Discovery by Service UUID is already ongoing (DRGN-11300). When the application has called `sd_ble_gattc_primary_services_discover()`, it should wait for the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event before calling `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()`.

- LL

    - If the application adds an all zeroes IRK with the
      `sd_ble_gap_device_identities_set()`, it will be treated as a valid entry in the device
      identity list. An all zeroes IRK is invalid and must not be added (DRGN-9083).

s122_nrf52_8.0.0-10.alpha

The S122 is a size-optimized central only BLE SoftDevice. The S122 API is a compatible subset of the S132 SoftDevice API. For features that are common to S122 and S132, the API is the same. To show the API compatibility, the S122 follows the same version numbering as S132. See the section "Changes" below for features that are not available in the S122 compared to the S132.

SoftDevice properties

- This SoftDevice variant is targeted towards nRF52820.
- This SoftDevice contains the Master Boot Record (MBR) version 2.5.0 (DRGN-11287).
    - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
    - Flash: **108 kB** (0x1B000 bytes)
    - RAM: **4.8 kB** (0x12A8 bytes). This is the minimum required memory. The actual requirements depend on 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 **1.75 kB** (0x700 bytes). Application writers should ensure that enough stack space is reserved to cover the worst-case SoftDevice call stack usage combined with the worst-case application call stack usage.
- The Firmware ID of this SoftDevice is 0xFFFE.

New functionality

This release is the first version of the S122.

Changes

Compared to the s132_nrf52_7.0.1, the following features have been added:

- SoftDevice:
    - `sd_power_usb*` APIs and events.
- GAP:
    - The security manager timer is now reset on the reception of each keypress notification (DRGN-13374).

Compared to the s132_nrf52_7.0.1, the following features have been removed:

- SoftDevice:
    - MWU is not supported (DRGN-13658).
    - Cache is not supported (DRGN-13451).
- GAP:
    - Peripheral Role is no longer supported and the following functions, structures, defines and events have been removed (DRGN-13474):
        - `sd_ble_gap_adv_addr_get()`
        - `sd_ble_gap_adv_set_configure()`

- `sd_ble_gap_adv_start()`

- `sd_ble_gap_adv_stop()`

- `sd_ble_gap_ppcp_set()`

- `sd_ble_gap_ppcp_get()`

- `sd_ble_gap_sec_info_reply()`

- `BLE_GAP_EVT_SEC_INFO_REQUEST`

- `BLE_GAP_EVT_SCAN_REQ_REPORT`

- `BLE_GAP_EVT_ADV_SET_TERMINATED`

- `BLE_GAP_OPT_LOCAL_CONN_LATENCY`

- `BLE_GAP_OPT_SLAVE_LATENCY_DISABLE`

- LE Data Length Extension is no longer supported and the following functions, structures, defines and events have been removed (DRGN-13451):

  - `sd_ble_gap_data_length_update()`

  - `BLE_GAP_EVT_DATA_LENGTH_UPDATE_REQUEST`, `BLE_GAP_EVT_DATA_LENGTH_UPDATE`

  - `BLE_GAP_DATA_LENGTH_AUTO`

  - `ble_gap_data_length_params_t`, `ble_gap_data_length_limitation_t`

  - `ble_gap_evt_data_length_update_request_t`, `ble_gap_evt_data_length_update_t`

- L2CAP:

  - L2CAP Connection Oriented Channels is no longer supported and the header file *ble_l2cap.h* with its functions, structures, defines and events have been removed (DRGN-13451):

    - `sd_ble_l2cap_ch_setup()`, `sd_ble_l2cap_ch_release()`, `sd_ble_l2cap_ch_rx()`, `sd_ble_l2cap_ch_tx()`, `sd_ble_l2cap_ch_flow_control()`

    - `ble_l2cap_ch_rx_params_t`, `ble_l2cap_ch_setup_params_t`, `ble_l2cap_ch_tx_params_t`, `ble_l2cap_conn_cfg_t`

    - `BLE_L2CAP_EVT_CH_SETUP_REQUEST`, `BLE_L2CAP_EVT_CH_SETUP_REFUSED`, `BLE_L2CAP_EVT_CH_SETUP`, `BLE_L2CAP_EVT_CH_RELEASED`, `BLE_L2CAP_EVT_CH_SDU_BUF_RELEASED`, `BLE_L2CAP_EVT_CH_CREDIT`, `BLE_L2CAP_EVT_CH_RX`, `BLE_L2CAP_EVT_CH_TX`

    - `ble_l2cap_evt_t`, `ble_l2cap_evt_ch_tx_t`, `ble_l2cap_evt_ch_rx_t`, `ble_l2cap_evt_ch_credit_t`, `ble_l2cap_evt_ch_sdu_buf_released_t`, `ble_l2cap_evt_ch_setup_request_t`, `ble_l2cap_evt_ch_setup_refused_t`, `ble_l2cap_evt_ch_setup_t`

Bug fixes

- SoftDevice

  - Fixed an issue where attempting to enable or disable privacy during passive scanning could cause the SoftDevice to assert. (DRGN-12165).

- Fixed an issue where `sd_ble_gap_device_name_set()` would return `NRF_ERROR_INTERNAL` if the allocated space for the device name is too small. The new behavior is that `NRF_ERROR_DATA_SIZE` is returned instead (DRGN-10195).
- Fixed an issue where SoftDevice could very rarely assert if LFCLKSRC is RCOSC and a scanner with long scan window was used (DRGN-12529).

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 (DRGN-5197/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.

  - If the scanner is configured with a scan window larger than 16 seconds, the scanner will truncate the scan window to 16 seconds (DRGN-10305).

- GATT

  - To conform to the Bluetooth Core Specification v5.2, there shall be no secondary service that is not referenced somehow by a primary service. The SoftDevice does not enforce this (DRGN-906).

Known Issues

- GATTC

  - The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event if `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()` is called when a Primary Service Discovery by Service UUID is already ongoing (DRGN-11300). When the application has called `sd_ble_gattc_primary_services_discover()`, it should wait for the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event before calling `sd_ble_gattc_primary_services_discover()` or `sd_ble_gattc_read()`.

- LL

  - If the application adds an all zeroes IRK with the `sd_ble_gap_device_identities_set()`, it will be treated as a valid entry in the device identity list. An all zeroes IRK is invalid and must not be added (DRGN-9083)