

s112_nrf52 release notes

Introduction to the s112_nrf52 release notes

About the document

These release notes describe the changes in the s112_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 s112_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.

Copyright © Nordic Semiconductor ASA. All rights reserved.

Contents

Introduction to the s112_nrf52 release notes	1
About the document	1
s112_nrf52_7.3.0	4
SoftDevice properties	4
New Features	4
Bug Fixes	4
Limitations	5
Known Issues	5
s112_nrf52_7.2.1	7
SoftDevice properties	7
New Features	7
Bug Fixes	7
Limitations	7
Known Issues	8
s112_nrf52_7.2.0	10
SoftDevice properties	10
New Features	10
Changes	10
Bug Fixes	11
Limitations	11
Known Issues	11
s112_nrf52_7.0.1	13
SoftDevice properties	13
Changes	13
Bug fixes	13
Limitations	14
Known Issues	14
s112_nrf52_7.0.0 (Deprecated)	16
SoftDevice properties	16
s112_nrf52_6.1.1	19
SoftDevice Properties	19
Changes	19
Bug Fixes	19
Limitations	19
Known Issues	20
s112_nrf52_6.1.0	21

SoftDevice Properties	21
New Functionality	21
Changes	21
Bug Fixes	22
Limitations	23
Known Issues	23
s112_nrf52_6.0.0	24
SoftDevice properties	24
New functionality	24
Changes	24
Bug fixes	25
Limitations	25
Known Issues	26
s112_nrf52810_5.1.0	27
SoftDevice properties	27
Device Compatibility	27
New functionality	27
Changes	27
Bug fixes	28
Limitations	29
Known Issues	29
s112_nrf52810_5.1.0-2.alpha	30
SoftDevice properties	30
Device Compatibility	30
New functionality	30
Changes	30
Bug fixes	31
Limitations	31
Known Issues	32

s112_nrf52_7.3.0

The main new feature of this version compared to the s112_nrf52_7.2.1 version is the possibility for the application to configure the slave to wake up every connection event if it detects that the master is not receiving its packets.

Notes:

- The release notes list changes since s112_nrf52_7.2.1.
- This SoftDevice is binary compatible with the s112_nrf52_7.2.1, and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52805, nRF52810, nRF52811, nRF52820, and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.7 kB** (0xEB8 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 **1.8 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 0x0126.

New Features

- GAP
 - A new option named `BLE_GAP_SLAVE_LATENCY_WAIT_FOR_ACK` is available to the application. This can be used to configure the slave to wake up every connection event if it has not received an ACK from the master for at least the length of the slave latency events. This configuration may increase the power consumption in environments with a lot of radio activity (DRGN-15555).

Bug Fixes

- LL
 - Fixed an issue where the last advertising event within the configured advertising duration may end with sending a packet without listening for a `CONNECT_IND` or `SCAN_REQ` packet (DRGN-15484).
- GAP
 - Fixed an issue where the application might not receive a `BLE_GAP_EVT_DISCONNECTED` event if the application does not continuously pull events from the SoftDevice (DRGN-15619).

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).
 - 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.
 - 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

- MBR
 - When copying the Bootloader on the nRF52805 or nRF52811 using the `SD_MBR_COMMAND_COPY_BL` MBR command, the MBR will not write-protect itself. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself (DRGN-11287).
- SoftDevice
 - When running on nRF52820 using `sd_protected_register_write` API can lead to undefined behavior (DRGN-12447).
 - The SoftDevice will generate a resolvable address for the TargetA field in directed advertisements if the target device address is in the device identity list with a non-zero IRK, even if privacy is not enabled and the local device address is set to a public address. A workaround is to remove the device address from the device identity list (DRGN-10659).
 - A memory access fault (`NRF_FAULT_ID_APP_MEMACC`) can occur in `sd_nvic_critical_region_exit()` if a high priority SoftDevice interrupt occurs during a critical section, for example due to radio traffic (DRGN-10613). It can be fixed by editing `__NRF_NVIC_SD_IRQS_1` in `nrf_nvic.h` so that it becomes:

```
#define __NRF_NVIC_SD_IRQS_1 ((uint32_t)(1U << (MWU_IRQn - 32)))
```
- GAP
 - If the Peer Preferred Connection Parameters Characteristic (PPCP) contains "No specific values indication (0xFFFF)", application should not perform a peripheral initiated connection parameter update using PPCP. Otherwise invalid values will be used in

L2CAP_CONNECTION_PARAMETER_UPDATE_REQ. A workaround is to always specify the connection parameters in the `sd_ble_gap_conn_param_update` call (DRGN-15111).

- 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).
- `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
- `ble_gap_cfg_role_count_t::adv_set_count` configuration parameter is not functional. The application should set it to `BLE_GAP_ADV_SET_COUNT_DEFAULT` when configuring the role count (DRGN-14113).

- 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).

s112_nrf52_7.2.1

The main new features of this version compared to the s112_nrf52_7.2.0 version are:

- The application can configure CCCD permission for the service change characteristic.
- Fixing an issue where the high duty cycle advertiser did not connect successfully.

Notes:

- The release notes list changes since s112_nrf52_7.2.0.
- This SoftDevice is binary compatible with the s112_nrf52_7.2.0 and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52805, nRF52810, nRF52811, nRF52820 and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.7 kB** (0xEB8 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 0x0119.

New Features

- GATTS
 - The application can configure CCCD permission for the Service Changed characteristic, using `BLE_GATTS_CFG_SERVICE_CHANGED_CCCD_PERM` (DRGN-15034).

Bug Fixes

- LL
 - Fixed an issue where the high duty cycle advertiser did not always connect successfully if the RC was used as the source for the LFCLK, and the peer didn't initiate the connection fast enough (DRGN-15003).

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).
- 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.
- 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

- MBR
 - When copying the Bootloader on the nRF52805 or nRF52811 using the `SD_MBR_COMMAND_COPY_BL` MBR command, the MBR will not write-protect itself. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself (DRGN-11287).
- SoftDevice
 - **Update 1:** When running on ~~nRF52811~~ nRF52820, using `sd_protected_register_write` API can lead to undefined behavior (DRGN-12447). The previous version of these release notes stated an incorrect board.
 - **Update 1:** ~~When running on nRF52811, using `sd_power_usb_*` APIs can lead to undefined behavior (DRGN-12720).~~ This is now documented in the API.
 - The SoftDevice will generate a resolvable address for the TargetA field in directed advertisements if the target device address is in the device identity list with a non-zero IRK, even if privacy is not enabled and the local device address is set to a public address. A workaround is to remove the device address from the device identity list (DRGN-10659).
 - A memory access fault (`NRF_FAULT_ID_APP_MEMACC`) can occur in `sd_nvic_critical_region_exit()` if a high priority SoftDevice interrupt occurs during a critical section, for example due to radio traffic (DRGN-10613). It can be fixed by editing `__NRF_NVIC_SD_IRQS_1` in `nrf_nvic.h` so that it becomes:

```
#define __NRF_NVIC_SD_IRQS_1 ((uint32_t)(1U << (MWU_IRQn - 32)))
```

- GAP
 - If the Peer Preferred Connection Parameters Characteristic (PPCP) contains "No specific values indication (0xFFFF)", application should not perform a peripheral initiated connection parameter update using PPCP. Otherwise invalid values will be used in `L2CAP_CONNECTION_PARAMETER_UPDATE_REQ`. A workaround is to always specify the connection parameters in the `sd_ble_gap_conn_param_update` call (DRGN-15111).

- 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).
 - `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
 - `ble_gap_cfg_role_count_t::adv_set_count` configuration parameter is not functional. The application should set it to `BLE_GAP_ADV_SET_COUNT_DEFAULT` when configuring the role count (DRGN-14113).
- GATTC
 - The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATTC_EVT_PRIM_SRVC_DISC_RSP` event if the `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).

s112_nrf52_7.2.0

The main new features of this version compared to the s112_nrf52_7.0.1 version is the efficient discovery of 128-bit UUIDs, and access to the USB power registers. This release makes 8dBm tx power available on nRF52820.

Notes:

- The release notes list changes since s112_nrf52_7.0.1.
- This SoftDevice is binary compatible with the s112_nrf52_7.0.1, and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52805, nRF52810, nRF52811, nRF52820 and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.7 kB** (0xEB8 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 0x0103.

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).
- SoC
 - API for reading USB power register and USB power events have been added to this variant of the SoftDevice. See the API documentation for `sd_power_usb_*` (DRGN-14468).

Changes

- SoftDevice
 - For nRF52820 the `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).
- LL

- The slave accepts an `LL_REJECT_IND` as a valid response to an `LL_PHY_UPDATE_REQ` for aborting a self-initiated PHY update procedure. This change was added to improve the interoperability with devices not conforming to the Bluetooth Specification when aborting the PHY update procedure (DRGN-14193).

Bug Fixes

- GAP
 - Fixed an issue where the peripheral raised a `BLE_GAP_EVT_CONN_PARAM_UPDATE` event delayed by 30 s. This was happening when the connection parameter update resulted in the already active parameters for the link (DRGN-9865).

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).
 - 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.
 - 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

- MBR
 - When copying the Bootloader on the nRF52805 or nRF52811 using the `SD_MBR_COMMAND_COPY_BL` MBR command, the MBR will not write-protect itself. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself (DRGN-11287).
- SoftDevice
 - When running on nRF52820, using `sd_protected_register_write` API can lead to undefined behavior (DRGN-12447).
 - The SoftDevice will generate a resolvable address for the TargetA field in directed advertisements if the target device address is in the device identity list with a non-zero IRK,

even if privacy is not enabled and the local device address is set to a public address. A workaround is to remove the device address from the device identity list (DRGN-10659).

- A memory access fault (NRF_FAULT_ID_APP_MEMACC) can occur in `sd_nvic_critical_region_exit()` if a high priority SoftDevice interrupt occurs during a critical section, for example due to radio traffic (DRGN-10613). It can be fixed by editing `__NRF_NVIC_SD_IRQS_1` in `nrf_nvic.h` so that it becomes:

```
#define __NRF_NVIC_SD_IRQS_1 ((uint32_t)(1U << (MWU_IRQn - 32)))
```

- GAP

- 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).
- `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
- `ble_gap_cfg_role_count_t::adv_set_count` configuration parameter is not functional. The application should set it to `BLE_GAP_ADV_SET_COUNT_DEFAULT` when configuring the role count (DRGN-14113).

- GATT

- The `ble_gattc_service_t::uuid` field is incorrectly populated in the `BLE_GATT_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_GATT_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).

s112_nrf52_7.0.1

This is a production release that contains minor, backward compatible, changes to the 7.0.0 release.

For some combinations of SoftDevice and nRF52 IC, only Bluetooth Core Specification v5.1 qualified listings are available with corresponding QDIDs from v7.0.1.

Updating to this version from v7.0.0 is recommended.

Notes:

- The release notes list changes since the s112_nrf52_7.0.0 release.
- This SoftDevice is binary compatible to the s112_nrf52_7.0.0 and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52805, nRF52810, nRF52811, nRF52820 and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes)
 - RAM: **3.7 kB** (0xeb8 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 0x00CD.

Changes

- SoftDevice
 - Bluetooth Core Specification v5.1 compatibility (DRGN-12400).
 - The VersNr field in the LL_VERSION_IND packet now contains the value 0x0A to indicate Bluetooth Core Specification v 5.1 compatibility (DRGN-12466).
 - References to Errata are added to the documentation of all the events and APIs which report RSSI and should be observed if using RSSI measurements.
- LL
 - Bluetooth Core Specification Erratum #10818 is incorporated, allow HCI ACL data packets with 0-length payload, but do not transmit anything until receiving the next non-zero continuation fragment (DRGN-11430).

Bug fixes

- SoftDevice

- Fixed an issue where the time scheduled for a flash write or flash page erase using `sd_flash_write` or `sd_flash_page_erase` APIs on nRF52811 will be longer than required and the same as for nRF52832 (DRGN-12539).

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.
- GATT
 - To conform to the Bluetooth Core Specification v 5.0, 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

- MBR
 - When copying the Bootloader on the nRF52805 or nRF52811 using the `SD_MBR_COMMAND_COPY_BL` MBR command, the MBR will not write-protect itself. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself (DRGN-11287).
- SoftDevice
 - The `BLE_GAP_EVT_SEC_INFO_REQUEST` event will not report the identity address of the peer to the application. This issue was also present in previous releases. A workaround is to do a mapping of the connection handle to the peer's identity address (DRGN-10340).
 - `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
 - The SoftDevice will generate a resolvable address for the `TargetA` field in directed advertisements if the target device address is in the device identity list with a non-zero IRK, even if privacy is not enabled and the local device address is set to a public address. This issue was present also in previous releases. A workaround is to set the IRK to zero or to remove the device address from the device identity list (DRGN-10659).
 - The SoftDevice may generate several events, when connected, based on peer actions, i.e. without prior action from the application. The `BLE_GAP_EVT_PHY_UPDATE_REQUEST` event, for instance, will be 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 details check `sd_ble_enable()` API in SoftDevice.
 - The following issue was previously mistakenly removed from the release notes:
A memory access fault (`NRF_FAULT_ID_APP_MEMACC`) can occur in `sd_nvic_critical_region_exit()` if a high priority SoftDevice interrupt occurs during a critical section, for example due to radio traffic (DRGN-10613). This issue was present also in

previous releases. It can be fixed by editing `__NRF_NVIC_SD_IRQS_1` in `nrf_nvic.h` so that it becomes:

```
#define __NRF_NVIC_SD_IRQS_1 ((uint32_t)(1U << (MWU_IRQn - 32)))
```

- 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).

s112_nrf52_7.0.0 (Deprecated)

The main new feature of this version compared to the s112_nrf52_6.1.1 is the ability to configure the inclusion of the Central Address Resolution (CAR) and Peripheral Preferred Connection Parameters (PPCP) characteristics.

Notes:

- This release has changed the API. This requires applications to be recompiled.
- The memory requirements of the s112 SoftDevice have changed.
- The release notes list changes since s112_nrf52_6.1.1.

SoftDevice properties

- This SoftDevice variant is compatible with nRF52805, nRF52810, nRF52811 and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes)
 - RAM: **3.7 kB** (0xeb8 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 0x00C4.

New functionality

- GAP
 - API to obtain the next connection event counter (DRGN-10913).
 - API for inclusion configuration of the CAR and PPCP characteristics (DRGN-10874).

Changes

- SoftDevice
 - Removed macros defining PPI channels and groups available to the application (DRGN-10382).
- GAP
 - The API for configuring improved advertiser role scheduling is removed. The SoftDevice now uses the improved scheduling configuration by default (DRGN-10754).

Bug fixes

- SoftDevice

- Fixed an issue where utilizing the MWU on nRF52832 would lead to undefined behavior (DRGN-10917).
- Fixed an issue where the application would be blocked when requesting an earliest possible Radio Timeslot (DRGN-10402).
- LL
 - Fixed an issue where the slave might disconnect if many packets were lost and there was an ongoing Connection Parameter Update (DRGN-11147).

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.
- GATT
 - To conform to the Bluetooth Core Specification v 5.0, 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

- MBR
 - When copying the Bootloader on the nRF52805 or nRF52811 using the `SD_MBR_COMMAND_COPY_BL` MBR command, the MBR will not write-protect itself. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself (DRGN-11287).
- SoftDevice
 - The time scheduled for a flash write or flash page erase using `sd_flash_write` or `sd_flash_page_erase` APIs on nRF52805 or nRF52811 will be longer than required and the same as for nRF52832.
 - The `BLE_GAP_EVT_SEC_INFO_REQUEST` event will not report the identity address of the peer to the application. This issue was also present in previous releases. A workaround is to do a mapping of the connection handle to the peer's identity address (DRGN-10340).
 - `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
 - A memory access fault (`NRF_FAULT_ID_APP_MEMACC`) can occur in `sd_nvic_critical_region_exit()` if a high priority SoftDevice interrupt occurs during a critical section, for example due to radio traffic (DRGN-10613). This issue was present also in previous releases. It can be fixed by editing `__NRF_NVIC_SD_IRQS_1` in `nrf_nvic.h` so that it becomes:

```
#define __NRF_NVIC_SD_IRQS_1 ((uint32_t)(1U << (MWU_IRQn - 32)))
```

- The SoftDevice will generate a resolvable address for the TargetA field in directed advertisements if the target device address is in the device identity list with a non-zero IRK, even if privacy is not enabled and the local device address is set to a public address. This issue was present also in previous releases. A workaround is to set the IRK to zero or to remove the device address from the device identity list (DRGN-10659).
- 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()`.

s112_nrf52_6.1.1

This is a production release that contains minor but important changes to the 6.1.0 release.

Notes:

- The release notes list changes since the s112_nrf52_6.1.0 release.
- This SoftDevice is binary compatible to the s112_nrf52_6.1.0 and memory requirements have not changed. Applications are therefore not required to be recompiled.

SoftDevice Properties

- This SoftDevice is production tested for nRF52810 and nRF52832.
- This SoftDevice variant is production tested for nRF52811.
- This SoftDevice contains the Master Boot Record (MBR) version 2.4.1 (DRGN-10680).
 - This MBR version is compatible with previous MBR versions.
- The combined MBR and SoftDevice memory requirements for this version are the same as for the s112_nrf52_6.1.0:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.86 kB** (0xF70 bytes). This is the minimum required memory. The actual requirements depend on the configuration chosen at `sd_ble_enable()` time.
 - The Firmware ID of this SoftDevice is 0x00B8.

Changes

- SoftDevice
 - The MBR 2.4.1 is a minor backward compatible configuration update of the MBR for this release. There were no bugs resolved in this update, only minor build configuration option changes (DRGN-10680).
 - Applications can improve the radio utilization for multiprotocol applications by enabling the improved advertiser role scheduling configuration through the BLE Option API. The time reserved for an advertising event will then be decreased by up to 1.3 ms (DRGN-10398).

Bug Fixes

- SoftDevice
 - Fixed an issue where NRF_TIMER0 may not be reset at the start of a radio timeslot (DRGN-10650).

Limitations

- MBR
 - When copying the Bootloader on the nRF52811 using the MBR command, `SD_MBR_COMMAND_COPY_BL`, flash is not protected using the BPROT peripheral. This does not change the behavior of the MBR or DFU process as the MBR cannot be configured to overwrite itself. (DRGN-11287).

- 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.
- GATT
 - To conform to the Bluetooth Core Specification v 5.0, 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
 - `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
 - If the application requests an earliest possible Radio Timeslot and the timeslot is blocked, the SoftDevice will repeat the same request until it times out, thereby blocking the main context and the lower application interrupt priority levels. A workaround is to increase the timeout of the Radio Timeslot request to make it able to fit after the event that is blocking the request (DRGN-10402).
 - The application should not enable MWU if the SoftDevice is run on nRF52832 as this leads to undefined behavior (DRGN-10917).
 - The SoftDevice will generate a resolvable address for the TargetA field in directed advertisements if the target device address is in the device identity list with a non-zero IRK, even if privacy is not enabled and the local device address is set to a public address. This can make devices certified for Bluetooth versions older than 4.2 ignore the advertising packets. This issue is present in SoftDevice versions 3.0.0 and later. A workaround is to set the IRK to zero or to remove the device address from the device identity list (DRGN-10659).

s112_nrf52_6.1.0

The s112_nrf52_6.1.0 is the first S112 SoftDevice to support both nRF52810 and nRF52832.

Notes:

- The release notes list changes since the s112_nrf52_6.0.0 release.
- This SoftDevice is binary compatible to the s112_nrf52_6.0.0. Applications are therefore not required to be recompiled and memory requirements have not changed.

SoftDevice Properties

- This SoftDevice is production tested for nRF52810 and nRF52832.
- This SoftDevice contains the Master Boot Record (MBR) version 2.2.2 (DRGN-9537).
- The combined MBR and SoftDevice memory requirements for this version are the same as for the s112_nrf52_6.0.0:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.86 kB** (0xF70 bytes). This is the minimum required memory. The actual requirements depend on the configuration chosen at `sd_ble_enable()` time.
- The Firmware ID of this SoftDevice is 0x00B0.

New Functionality

- SoftDevice
 - The SoftDevice variant, flash usage, reserved PPIs, and reserved interrupt priorities are now available at compile time to the application through new APIs (DRGN-9627).
 - An API is added to enable the application to remove an unused UUID entry from the UUID table (DRGN-10389).
- GAP
 - With the new `sd_ble_gap_adv_addr_get()` API, the application can now get the Bluetooth device address that is being used by the advertiser (DRGN-10470).

Changes

- SoftDevice
 - On nRF52810, the time reserved by the SoftDevice is reduced by 297 Åµs when performing a flash word write, and by 4.7 ms when performing a flash page erase. This increases the probability of successfully scheduled flash operations (DRGN-9048).
 - Improved documentation for the `NRF_ERROR_INVALID_STATE` error code (DRGN-9693).
 - When the SoftDevice is acting as a peripheral, and the RC oscillator is used as the LFCLK source, the configured RC calibration period can now be increased. By default, the SoftDevice will now increase the receive window if two consecutive packets are missed and will then perform RC calibration if necessary (DRGN-9852).
 - SoftDevice 112_nrf52_6.0.0 accepted an advertising interval larger than `BLE_GAP_ADV_INTERVAL_MAX` as an experimental feature. However, this configuration

could make the SoftDevice assert. Now, the SoftDevice will return `NRF_ERROR_INVALID_PARAM` if the application configures an advertising interval larger than `BLE_GAP_ADV_INTERVAL_MAX` (DRGN-10322).

- Radio utilization for multi-protocol applications are improved significantly as the time allocated for a normal Radio Timeslot request session has decreased by up to 1 ms (DRGN-10405).
- GATT
 - `sd_ble_gatts_rw_authorize_reply()` now allows sending the `0xFC` (Write Request Rejected) profile error code which was introduced in the Bluetooth Core Specification Supplement CSSv7 (DRGN-10373).
- LL
 - Instead of disconnecting, the SoftDevice will now respond with `LL_UNKNOWN_RSP` when receiving control procedure PDUs with invalid lengths (DRGN-9997).

Bug Fixes

- SoftDevice
 - Fixed an issue where a HardFault could generate a new HardFault if the application called a NULL pointer (DRGN-9607).
 - Fixed an issue where the SoftDevice HardFault handler could hang if the application wrote to protected memory (DRGN-9694).
 - Fixed an issue where the HFXO would sometimes not be released properly after RC calibration. This is in addition to the bug fix for a similar condition resolved in `s112_nrf52_6.0.0` (DRGN-9920, DRGN-10166).
 - Fixed an issue where the PA/LNA GPIOs could be triggered too late. Furthermore, the PA pin is now set active 23 μ s before RADIO TX start, instead of 5 μ s before RADIO TX start. The LNA pin is set active 5 μ s before RADIO RX start, as before (DRGN-9928).
 - Fixed documentation for `SD_EVT_IRQHandler` and `RADIO_NOTIFICATION_IRQHandler`, where the default interrupt priority was documented incorrectly (DRGN-10174).
 - Fixed an issue where LFRC oscillator calibration could fail (DRGN-10255).
 - Fixed an issue that could make the SoftDevice assert when scheduling events close together (DRGN-10316).
- GAP
 - Fixed an issue where the advertiser would not update its address type if `sd_ble_gap_addr_set()` or `sd_ble_gap_privacy_set()` was called after `sd_ble_gap_adv_set_configure()` and before `sd_ble_gap_adv_start()` (DRGN-10025).
 - Fixed an issue that could cause an assert when an advertiser configured with invalid parameters connected to a peer (DRGN-10355).
 - Fixed an issue that could cause an assert when the advertiser was stopped (DRGN-10364).
- LL
 - Fixed an issue that could cause links to disconnect (DRGN-9844).
 - Fixed an issue where the slave might not listen during the entire connection parameter update (DRGN-10086).

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.
- GATT
 - To conform to the Bluetooth Core Specification v 5.0, 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
 - `sd_ble_gap_device_name_set()` may return `NRF_ERROR_INTERNAL` instead of `NRF_ERROR_NO_MEM` if the allocated space for the device name is too small. A workaround is to allocate large enough space for the device name before calling `sd_ble_gap_device_name_set()` (DRGN-10195).
 - If the application requests an earliest possible Radio Timeslot and the timeslot is blocked, the SoftDevice will repeat the same request until it times out, thereby blocking the main context and the lower application interrupt priority levels. A possible workaround is to increase the timeout of the Radio Timeslot request to make it able to fit after the event that is blocking the request (DRGN-10402).

s112_nrf52_6.0.0

The main new feature of s112_nrf52_6.0.0 compared to s112_nrf52810_5.1.0 is channel information in the Received Signal Strength Indication (RSSI) reports.

Notes:

- This release has changed the Application Programmer Interface (API). This requires applications to be recompiled.
- The memory requirements of the S112 SoftDevice have changed.
- The release notes list changes since s112_nrf52810_5.1.0.
- The SoftDevice release naming convention has changed: Instead of specifying the platform supported by the SoftDevice in the release name, the release notes will have this information.

SoftDevice properties

- This SoftDevice variant is production tested for nRF52810. It can be used on nRF52832 for development purposes.
- This version of the SoftDevice contains the Master Boot Record (MBR) version 2.2.2 (DRGN-9537).
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100.0 kB** (0x19000 bytes).
 - RAM: **3.86 kB** (0xF70 bytes). This is the minimum required memory. The actual requirements depend on the configuration chosen at `sd_ble_enable()` time.
- The Firmware ID of this SoftDevice is 0x00A7.

New functionality

- SoftDevice
 - The SoftDevice API for advertising and scanning is updated and prepared to support future features. For more information, see the migration document (DRGN-9712).
- GAP
 - Channel number for RSSI measurement is now available in advertising reports (DRGN-9473).
 - Channel number for RSSI measurement is now available for connections (DRGN-9667).
 - The SoftDevice now supports the configuration of TX power per link and per role (DRGN-6659).

Changes

- SoftDevice
 - The SoftDevice now returns `NRF_ERROR_BUSY` from flash API functions until the event generated after a previous flash operation has been pulled (DRGN-9565).
 - The SoftDevice now has an additional API for write-protecting memory. This can now be achieved by accessing the BPROT peripheral configuration registers through `sd_protected_register_write()` (DRGN-9337).
 - A message sequence chart for Unexpected Security Packet Reception has been added to Peripheral Security Procedures in the API documentation (DRGN-9479).

- GATT
 - The SoftDevice will now return `NRF_ERROR_TIMEOUT` instead of `NRF_ERROR_BUSY` from GATT API functions if a GATT procedure is blocked due to a previous procedure timeout (DRGN-9545).
 - Clarified API documentation: The length field in the parameter struct passed to `sd_ble_gatts_hvx()` may be written to by the SoftDevice (DRGN-9620).
- LL
 - The documentation of the PHY Update procedure is improved (DRGN-9678).
 - Bluetooth Core Specification Erratum #7408 is incorporated, meaning that it is now accepted to receive an `LL_UNKNOWN_RSP` during encryption procedure (DRGN-8414).

Bug fixes

- SoftDevice
 - Fixed an issue where `sd_ble_gap_rssi_get()` could sometimes return `NRF_ERROR_SUCCESS` with an invalid RSSI (DRGN-9746).
 - Fixed an issue where the HFXO would sometimes not be released properly after RC calibration (DRGN-9920).
- GATT
 - Fixed an issue where the SoftDevice could drop a write request if it was received at the same time as a write command (DRGN-9709).
- LL
 - Fixed an issue where the slave could disconnect with status code `BLE_HCI_DIFFERENT_TRANSACTION_COLLISION` if master sent an `LL_UNKNOWN_RSP` after a PHY procedure collision (DRGN-9870).
 - Fixed an issue where the SoftDevice might advertise with the `RxAdd` bit set to 1 for undirected advertisements. According to the Bluetooth Core Specification v 5.0, the `RxAdd` bit is reserved for future use for these PDU types (DRGN-9739).
 - Fixed an issue where the SoftDevice could assert if the identity list was used while advertising (DRGN-9723).

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 Core Specification v 5.0, 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
 - If the application calls a NULL pointer, there will be a HardFault inside the SoftDevice HardFault handler (DRGN-9607).
 - If `sd_ble_gap_addr_set()` or `sd_ble_gap_privacy_set()` is called after `sd_ble_gap_adv_set_configure()` and before `sd_ble_gap_adv_start()`, the advertiser will not update its address type (DRGN-10025).

s112_nrf52810_5.1.0

This release is the first production version of the S112. The S112 is a size-optimized peripheral-only BLE SoftDevice for Nordic Semiconductor's nRF52810 chip. The S112 API is a compatible subset of the S132 SoftDevice API. For features that are common to S112 and S132, the API is the same. To show the API compatibility, the S112 follows the same version numbering as S132.

The main change relative to the s112_nrf52810_5.1.0-2.alpha is that the observer role functionality is removed from the SoftDevice. The SoftDevice also has some minor updates and bug fixes.

Notes:

- The priority of SoftDevice interrupts SD_EVT_IRQn (SWI2_IRQn) and RADIO_NOTIFICATION_IRQn (SWI3_IRQn) is 6. This is different from S132 4.x and previous SoftDevices (DRGN-9245).

SoftDevice properties

- The S112 SoftDevice Specification document will be available at <http://infocenter.nordicsemi.com/> .
- This version of the SoftDevice contains the Master Boot Record (MBR) version 2.2.2 (DRGN-9537).
- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **96 kB** (0x18000 bytes).
 - RAM: **3.68 kB** (0xeb8 bytes). This is the minimum required memory. The actual requirements depend on the configuration chosen at `sd_ble_enable()` time.

Device Compatibility

The SoftDevice is built, production tested, and qualified for use with nRF52810.

The SoftDevice is not production tested and qualified for use with nRF52832. However, the SoftDevice can be used for development purposes on the nRF52832 if minor performance and stability issues are accepted during development.

New functionality

This release has no new features compared to the s112_nrf52810_5.1.0-2.alpha.

Changes

Relative to the s112_nrf52810_5.1.0-2.alpha:

- GAP

- Observer role is no longer supported, and the following functions, structures, defines, and events have been removed (DRGN-9298):

- BLE_GAP_EVT_ADV_REPORT
- BLE_GAP_TIMEOUT_SRC_SCAN
- BLE_GAP_SCAN_INTERVAL_MIN, BLE_GAP_SCAN_INTERVAL_MAX
- BLE_GAP_SCAN_WINDOW_MIN, BLE_GAP_SCAN_WINDOW_MAX
- BLE_GAP_SCAN_TIMEOUT_MIN, BLE_GAP_SCAN_TIMEOUT_MAX
- ble_gap_scan_params_t
- ble_gap_evt_adv_report_t
- sd_ble_gap_scan_start()
- sd_ble_gap_scan_stop()

- LL

- The SoftDevice now sends LL_REJECT_EXT_IND instead of LL_REJECT_IND if the peer has indicated support for LL_REJECT_EXT_IND (DRGN-9539).

Relative to the s132_nrf52_5.0.0:

- SoftDevice

- sd_flash_protect() will now return NRF_ERROR_NOT_SUPPORTED if a non-zero value is given for a BPROT configuration register that is not available on the platform (DRGN-9336). (This was also part of the s112_nrf52810_5.1.0-2.alpha release.)
- References to EGU* have been removed from nrf_soc.h and nrf_nvic.h as the SoftDevice is using SWI and not EGU to generate interrupts (DRGN-9257). (This was also part of the s112_nrf52810_5.1.0-2.alpha release.)

See the release notes for the s112_nrf52810_5.1.0-2.alpha for the rest of the changes.

Bug fixes

Compared to the s112_nrf52810_5.1.0-2.alpha, the following bugs have been fixed:

- SoftDevice

- Fixed an issue where sd_ble_gatts_attr_get() and sd_ble_gatts_value_get() could return undocumented BLE_ERROR_INVALID_ATTR_HANDLE error code in a situation where they should have returned NRF_ERROR_NOT_FOUND (DRGN-9216).
- Fixed an issue where the API documentation was referencing removed APIs (DRGN-9484).
- Fixed an issue where some include directives were missing in ble_gatts.h (DRGN-9467).

- GATT

- Fixed an issue where the SoftDevice could assert if ATT packets longer than the LL packet size were sent and received at the same time (DRGN-9328).

- LL

- Fixed an issue where the SoftDevice could assert while doing radio or flash activity (DRGN-9463).
- Fixed an issue where the SoftDevice could send LL_FEATURE_RSP with incorrect FeatureSet (DRGN-9551).

- Fixed an issue where the slave disconnect event reason code was set to `HCI_LOCAL_HOST_TERMINATED_CONNECTION` instead of `HCI_STATUS_CODE_PIN_OR_KEY_MISSING`. This occurred if the LTK (Long Term Key) was missing during the re-encryption of the link (DRGN-9190). (This was also fixed in the s112_nrf52810_5.1.0-2.alpha 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 levels higher than 6 (priority level numerical values lower than 6) as this can lead to undefined behavior.
- GATTS
 - To conform to the Bluetooth Core Specification v 5.0, 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

- GAP
 - The SoftDevice can assert if the whitelist and identity list is set at the same time with matching addresses. A workaround for this issue is to clear the whitelist before setting the identity list (DRGN-9535).

s112_nrf52810_5.1.0-2.alpha

The S112 is a size-optimized peripheral only BLE SoftDevice for Nordic Semiconductor's nRF52810 chip. The S112 API is a compatible subset of the S132 SoftDevice API. For features that are common to S112 and S132, the API is the same. To show the API compatibility, the S112 follows the same version numbering as S132. See the section "Changes" below for features that are not available in the S112 compared to the S132.

SoftDevice properties

- The combined MBR and SoftDevice memory requirements for this version are as follows:
 - Flash: **100 kB** (0x19000 bytes)
 - RAM: **3.65 kB** (0xe98 bytes)

Device Compatibility

This SoftDevice is built and tested for nRF52810.

For development purposes this SoftDevice can be run on the nRF52832, but some of the Errata workarounds for that device are not present in this version of the alpha SoftDevice. This may result in minor performance and stability issues on nRF52832.

New functionality

This release is the first version of the S112. It is based upon the s132_nrf52_5.0.0, and has no new functionality compared to that version.

Changes

Compared to the s132_nrf52_5.0.0, the following features have been removed:

- SoftDevice:
 - MWU is not supported, as the nrf52810 does not have MWU (DRGN-9341).
 - Cache is not supported, as the nrf52810 does not have Cache (DRGN-9256).
- GAP:
 - Observer role is still present in this alpha version of the SoftDevice, but might be removed for the production version.
 - Central Role is no longer supported and the following functions, structures, defines and events have been removed (DRGN-9145):
 - `sd_ble_gap_connect()`, `sd_ble_gap_connect_cancel()`
 - `sd_ble_gap_encrypt()`
 - `BLE_GAP_EVT_CONN_PARAM_UPDATE_REQUEST`
 - `ble_gap_evt_conn_param_update_request_t`
 - `BLE_GAP_OPT_COMPAT_MODE_1`
 - `ble_gap_opt_compat_mode_1_t`
 - `BLE_GAP_ROLE_CENTRAL`, `BLE_GAP_ROLE_COUNT_CENTRAL_DEFAULT`, `BLE_GAP_ROLE_COUNT_CENTRAL_SEC_DEFAULT`

- LE Data Length Extension is no longer supported and the following structures, defines and events have been removed (DRGN-9242):

- `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 it's functions, structures, defines and events has been removed (DRGN-923 8):

- `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

Compared to the s132_nrf52_5.0.0, the following bugs have been fixed:

- **SoftDevice**
 - Fixed an issue where Radio Notification could be suppressed between connection events when Connection Event Length Extension was enabled (DRGN-7687).
- **GAP**
 - Fixed an issue where the SoftDevice could assert if the white list and identity list were set at the same time with matching addresses (DRGN-9535).

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 Core Specification v 5.0, 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

No known issues.