ANT_S313_nrf52_7.0.1 release notes

Contents

ANT_S313_nrf52_7.0.1

The ANT_S313_nrf52_7.0.1 SoftDevice is the first production release of this variant. It is based upon the ANT_S212_nrf52_7.0.1 (ANT) SoftDevice and S113 v7.0.1 (BLE) SoftDevice combined. The S313 contains all features present in the ANT_S312_nRF52_7.0.1 SoftDevice. In addition, it includes the LE Data Packet Length Extensions, Connection-Oriented Channels in LE Credit Based Flow Control Mode, and the ability to trigger a task, for example a GPIOTE task, at the start of a connection event. The S313 API is a compatible superset of the S312 SoftDevice API and a compatible subset of the S332 and S340 SoftDevice APIs. For features that are common to S312, S313, S332, and S340, the Application Programming Interface (API) is the same. To show the API compatibility, the S313 follows the same version numbering as S313, S332, and S340. For features that are available in the S313 compared to the S312 v 6.1.1, see section "New functionality" below.

Notes

- The release notes list changes since S312_nrf52810_6.1.1.

SoftDevice Properties

- This SoftDevice variant is compatible with nRF52810, nRF52811, nRF52832, nRF52833, nRF52840
- This version of the 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 is as follows:
    - Flash: **152 kB** (0x26000 bytes)
    - RAM: **5.5 kB** (0x1580 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 **1.75 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 0xD0

New Functionality

- **LL**
    - Data length extension feature support (DRGN-7245)
- **L2CAP**
    - Connection-Oriented Channels in LE Credit Based Flow Control Mode (DRGN-8572)
- **GAP**
    - API to obtain the next connection event counter (DRGN-10913).
    - API for triggering a task when the SoftDevice is about to start a connection event (DRGN-10914).
    - API for inclusion configuration of the CAR and PPCP characteristics (DRGN-10874).

Changes

- **ANT**
  - PA/LNA support extended to GPIOs on Port 1 for supported devices
  - Maximum Tx Output Power of 8 dBm enabled for supported devices.
- **SoftDevice**
  - Bluetooth Core Specification v5.1 qualified (DRGN-12400).
  - The VersNr field in the LL_VERSION_IND packet now contains the value 0x0A to indicate Bluetooth Core Specification v5.1 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.
  - Removed macros defining PPI channels and groups available to the application (DRGN-10382).
- **LL**
  - Bluetooth Core Specification Erratum #10818 is incorporated. The SoftDevice now allows HCI ACL data packets with 0-length payload, but does not transmit anything until receiving the next non- zero continuation fragment (DRGN-11430).
  - Bluetooth Core Specification Erratum #10750 is incorporated. The `BLE_GAP_EVT_DATA_LENGTH_UPDATE` event will now be raised to the application when switching to and from Coded PHY. On-air behavior has not changed (DRGN-11435).
- **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

- **ANT**
  - Fixed an issue where Slave Shared Channels would not send an uplink reply in specific cases where the shared address matched.
  - Fixed an issue where Timestamp of a received message would be invalid on initial channel acquisition.
- **SoftDevice**
  - Fixed an issue where utilizing the MWU on nRF52832 would lead to undefined behavior (DRGN-10917).
  - 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).
  - 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

- **ANT**
    - The low frequency RC oscillator clock source (NRF_CLOCK_LF_SRC_RC) is not tested or intended for use with the ANT stack.
- **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 ANT or 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.
    - Flash write operations may exceed the timeout provided when performed with certain protocol operations (e.g. ANT Continuous Scan).
- **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 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 nRF52810, nRF52811 or nRF52832 using `sd_power_usb*` APIs can lead to undefined behavior (DRGN-12720).
    - When running on nRF52833 or nRF52840, using `sd_flash_protect` or `sd_protected_register_write` APIs can lead to undefined behavior (DRGN-12447).
    - When running on nRF52811, using `sd_protected_register_write` API can lead to undefined behavior (DRGN-12447).
    - 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 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.
- 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 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).