

# s210\_nrf51422 release notes

- [Introduction to the s210\\_nrf51422 release notes](#)
- [s210\\_nrf51422\\_5.0.0](#)
  - [New functionality](#)
  - [Bug fixes](#)
  - [Changes](#)
  - [Limitations](#)
  - [Known Issues](#)
- [s210\\_nrf51422\\_4.0.1](#)
  - [New functionality](#)
  - [Bug fixes](#)
  - [Changes](#)
  - [Limitations](#)
  - [Known Issues](#)
- [s210\\_nrf51422\\_4.0.0](#)
  - [New functionality](#)
  - [Bug fixes](#)
  - [Changes](#)
  - [Limitations](#)
  - [Known Issues](#)
- [s210\\_nRF51422\\_3.0.0](#)
  - [New functionality](#)
  - [Bugfixes](#)
  - [Changes](#)
  - [Limitations](#)
  - [Known issues](#)

## Introduction to the s210\_nrf51422 release notes

These release notes describes the changes in the s210\_nrf51422 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 the overview. More details regarding changes and new features may be found in the s210\_nrf51422 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.

# s210\_nrf51422\_5.0.0

- The SoftDevice Specification corresponding to this release is the S210 SoftDevice Specification v3.0.0
- This version of the SoftDevice contains the Master Boot Record (MBR) version 1.0.0
- The combined MBR and SoftDevice memory requirements for this version are as follows:
  - Flash: **52kB** (0xD000 bytes)
  - RAM: **2.25kB** (0x0900 bytes)

## New functionality

- **ANT**
  - **ANT Stack Enable Configuration.** Addition of a new API, `sd_ant_enable()`, allows the application to configure the following ANT stack settings:
    - Total number of ANT channels
    - Total number of encrypted channels
    - Total size of TX burst transfer buffer
  - Increased max supported ANT channels to 15 from 8.
  - Increased max supported encrypted channels to 15 from 1. Must be less than or equal to total number of ANT channels.

## Bug fixes

- **SoftDevice**
  - Fixed an issue where the SoftDevice current consumption could remain high (1mA) after disabling the SoftDevice when running on RC LFCLOCK (DRGN-5472, NRFFOETT-968).
- **ANT**
  - Error handling added on attempts to send transmissions on background scanning channel.
  - Addressed bug in fast channel initiation with pending close channel

## Changes

- **SoftDevice**
  - The DC/DC converter usage has been simplified by deprecating the AUTO option. The DC/DC converter is now supported when running the SoftDevice on nRF51 series IC revision 3 (DRGN-2420, DRGN-4622). (For IC revision information, see the nRF51 Series Compatibility Matrix, downloadable from the Nordic Semiconductor web page.)

- 6 previously reserved PPI channels have been freed and may be used by the application (DRGN-5082).
- A set of new macros have been introduced to access the SoftDevice info structure directly from hex or bin SoftDevice images (DRGN-4642).
- **ANT**
  - ANT Version string changed to AXX5.00B00
  - **Calling `sd_softdevice_enable()` by default creates 1 ANT channel (with encryption support) and 64 byte transmit burst buffer.**
  - If additional channels and/or functionality is required, `sd_ant_enable()` must be used to specify the desired configuration as well as supplied with required application RAM to be used by the ANT stack.

## 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)
  - On nRF51 series IC revision 2 and earlier, DC/DC converter operation controlled by the SoftDevice may interfere with radio function. As a result, `nrf_power_dcdc_mode` should not be modified by the application. The mode must not be set to `NRF_POWER_DCDC_MODE_AUTOMATIC` or `NRF_POWER_DCDC_MODE_ON` at any time (DRGN-2420).

## Known Issues

- **SoftDevice**
  - If Radio Notifications are configured while the SoftDevice is not in an idle state, the SoftDevice may in some situations assert (DRGN-5556, NRFFOETT-986). The workaround is to only configure Radio Notifications when the SoftDevice is in an idle state with no protocol stack or other SoftDevice activity in progress, as described in the API and in the S210 SoftDevice Specification v2.0.

# s210\_nrf51422\_4.0.1

This release is a bug fix release that adds an API header file that was missing from s210\_nrf51422\_4.0.0. The SoftDevice hex file is unchanged from the previous version.

This Softdevice is described in the S210 nRF51422 SoftDevice Specification v.2.0

This version of the SoftDevice contains the MBR version 1.0.0

## New functionality

- No new functionality.

## Bug fixes

- **SoftDevice**
  - nrf\_mbr.h added to the API header files.

## Changes

- No changes.

## 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.

## Known Issues

- No known issues.

# s210\_nrf51422\_4.0.0

This release adds several new features, among them support for over-the-air Device Firmware Update (DFU) through ANTFS and support for running other protocol stacks concurrently with the SoftDevice ANT protocol stack.

This is a major release which has changed the Application Programmer Interface (API), requiring applications to be recompiled.

This Softdevice is described in the S210 nRF51422 SoftDevice Specification v.2.0

This version of the SoftDevice contains the MBR version 1.0.0

## New functionality

- **SoftDevice**
  - The SoftDevice now supports concurrent multiprotocol operation using the Concurrent Multiprotocol Timeslot API. This enables the application to run a separate radio protocol or reserve timeslots from application space concurrently with the SoftDevice ANT protocol stack.
  - The SoftDevice now contains a Master Boot Record (MBR), which enables Device Firmware Update (DFU) of the SoftDevice itself (in addition to the application and bootloader) over the air. The MBR API enables copying and comparing regions in flash memory, and interrupt forwarding.
  - ROSC calibration can now be configured to be temperature dependent.
  - The Flash API is now available also when the SoftDevice is disabled.
  - The SoftDevice can now be configured to forward interrupts to one of several applications using the new `sd_softdevice_vector_table_set()` API call.
- **ANT**
  - RSSI proximity can now be configured and used in ANT RX scanning channel.
  - Added support to allow wildcard channel ID uplink transmissions on an ANT RX scanning channel.
  - Asynchronous TX channels are now capable of running asynchronously in the presence of other running ANT channels.
  - Channels opened with channel fast initiation option now start as soon as possible in the presence of other running ANT channels.
  - Improved RX scanning channel coexistence with application flash write and application radio timeslot scheduling.

## Bug fixes

- Fixed issue where radio override settings in FICR may not get applied for NRF\_1MBIT mode when running ANT.
- Fixed issue where specifying optional frequency hopping (FH) field in ANT advanced burst transfer parent configuration never puts the transfer parent in FH mode.
- Added missing radio coexistence configuration capability in ANT capabilities message.

## Changes

- **SoftDevice**

- Code and RAM region changes. Softdevice now includes Master Boot Record (MBR) region as described in the New functionality section.

	PREVIOUS	NEW
	s210_nrf51422_3.0.0 (ANT AXX3.02B00)	s210_nrf51422_4.0.0 (ANT AXX4.00B00)
	48 kB	52 kB
SoftDevice Reserved Code	(0x00000000 – 0x0000BFFF)	(4 kB assigned to MBR) (0x00000000 – 0x0000CFFF)
	2.25 kB	2.25 kB
SoftDevice Reserved RAM	(0x20000000 – 0x200008FF)	(4 bytes assigned to MBR) (0x20000000 – 0x200008FF)
SoftDevice Callstack	1 kB(*)	1 kB(*)
Application Start Code	0x0000C000	0x0000D000
Application Start RAM	0x20000900	0x20000900

- (\*) maximum estimation based on simulation.
- The FWID is no longer stored in the UICR. Updated versions of the tools (nRFgo Studio, nrfjprog) compatible with this change are available as downloads from the Nordic Semiconductor web page.
- The SoftDevice hex file no longer contains the SoftDevice size in the UICR.CLENR0 register. This means that the SoftDevice is no longer protected by default. The updated versions of the tools (nRFgo Studio, nrfjprog) will write the SoftDevice size to the UICR.CLENR0 by default, thereby restoring default protection. Having protection enabled will not allow Device Firmware Update to a SoftDevice of larger size than the original; therefore the tools make it optional to not set the UICR.CLENR0 register.
- `sd_softdevice_enable()` no longer requires CLENR0 and RBPCONF.PR0 to be set as required by previous SoftDevice versions. This is to fully accommodate flexible SoftDevice sizes when performing Device Firmware Update.
- The `sd_softdevice_forward_to_application()` call has been replaced with `sd_softdevice_vector_table_base_set()`, which takes the forwarding address as an argument.
- SVC number changes.
- The Radio Disable feature supported in previous SoftDevice version is replaced by the Concurrent Multiprotocol Timeslot feature (refer to New Functionality section for more detail).
- Flash API operation behavior changes. Flash writes/erases retries are now based on timeout values rather than fixed number of retries. Total operation timeout is a combination of low priority timeout (1ms) and normal priority timeout (100ms). If the initial low priority flash operation could not be scheduled within low priority timeout, the operation is raised to normal priority. If then the operation could not be scheduled within normal priority timeout, the flash operation is treated as being timed out.

- **ANT**

- The `sd_ant_prox_search_set()` API call now takes an additional parameter to specify custom (non-ANT indexed) proximity value. See the API documentation for more information on the usage.

## 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.

## Known Issues

- No known issues.

# s210\_nRF51422\_3.0.0

This is the first web release of the S210 SoftDevice, implementing ANT stack on the nRF51422 chip.

## New functionality

- **ANT Improvements**
  - Enabled 60 kbps encrypted burst transfers mode. This was previously restricted to 40 kbps.
  - Prevent ANTRF Active Notifications from being enabled if ANT Async transmit channel is assigned. Done in order to avoid false notifications.
  - Added ANT enhanced channel spacing feature to prevent continuous consecutive channel collisions events for multiple tracking channels.
  - Added requested custom TX power level.
- **SOC Radio Notification Interrupt**
  - [s210\\_nrf51422\\_3.0.0](#) SoftDevice introduces radio notification feature for alerting the application of radio activity/inactivity. A dedicated software interrupt (SWI1\_IRQn) is used to signal the application of these events.
    - Note: ANTRF Active notification feature is still available for use through the ANT API set. However, it is recommended that the SoC Radio Notification is used instead due to the following reasons:
      - SoC Radio Notification uses dedicated software interrupt for immediate notification to the application. ANT RFActive Notification uses the ANT event queue for notification and could be subjected to application event processing delay.
      - Using SoC Radio Notifications is forward compatible with S310 SoftDevice where multiple concurrent protocols may be active. ANT RFActive Notification only applies to synchronous ANT channel protocol events.
- **Softdevice Bootloader**
  - Support for bootloader has been added. Please refer to S210 SDS for more information.

## Bugfixes

- Fixed issue where calling `sd_ant_stack_reset` or `sd_softdevice_disable` or `ant_cw_test_mode_init` after closing all running ANT channels may cause a SoftDevice freeze. Issue could only occur if the last channel closed is a master channel or a non-tracking slave channel.
- Fixed issue where `sd_ant_stack_reset` did not properly end ANT CW test mode (started by `sd_ant_cw_test_mode`).
- Fixed issue where `sd_ant_burst_handler_request` did not accept a single START & END segment and prevented the application from requesting bursts with less than 2 segments.
- Added missing `RADIO_TX_POWER_LVL4` to ANT parameters header file. This mode was supported in [s210\\_nrf51422\\_2.0.0](#); however it was missing in the header file.
- Fixed issue where ANT TX and RX burst transfer did not work with ANT ARCT test box.
- Fixed issue where ANT receivers could not alter peer transmit timing behaviour when required.
- Fixed issue where ANT RX scanning channel timestamps did not report extended RX timestamp values correctly.



- Fixed issue where ANT broadcast data may be corrupted if changed by `sd_ant_broadcast_message_tx` exactly upon radio preparation time.

## Changes

- Code and RAM region Changes**

	OLD	NEW
	<a href="#">s210_nrf51422_2.0.0</a> (ANT AXX1.06B00)	<a href="#">s210_nrf51422_3.0.0</a> (ANT AXX3.00B03)
SoftDevice Reserved Code	40 kB (0x00000000 – 0x00009FFF)	48 kB (0x00000000 – 0x0000BFFF)
SoftDevice Reserved RAM	2 kB (0x20000000 – 0x200007FF)	2.25 kB (0x20000000 – 0x200008FF)
SoftDevice Callstack	400 bytes(*)	1 kB(*)
Application Start Code	0x0000A000	0x0000C000
Application Start RAM	0x20000800	0x20000900

- (\*) maximum estimation based on simulation.
- Softdevice Protected Peripherals**

OLD	NEW
<a href="#">s210_nrf51422_2.0.0</a> (ANT AXX1.06B00)	<a href="#">s210_nrf51422_3.0.0</a> (ANT AXX3.00B00)
POWER_CLOCK	POWER_CLOCK
RADIO	RADIO
RTC0	RTC0
RNG	RNG
ECB	ECB
CCM_ARR	CCM_ARR
PPI	PPI
SWI4_IRQn	SWI4_IRQn
SWI5_IRQn	SWI5_IRQn
	TIMER0
	NVMC
	TEMP

- Timer0 is now protected by the SoftDevice and used for internal timing operations. It cannot be directly accessed by the application.
  - NVMC is now protected by the SoftDevice. It cannot be directly accessed by the application. Flash write/erase procedures must be done through the newly provided flash access APIs in [s210\\_nrf51422\\_3.0.0](#) SoftDevice.
  - TEMP is now protected by the SoftDevice. It can be only accessed via the newly provided temperature read API.
- ANT SVC Base Change**

- ANT SVC stack base changed from 0x50 to 0xC0.  
(Aligned with ANT SVC stack base used by S310 for cross compatibility).
- **ANT Event Software Interrupt**
  - ANT event software interrupt moved from SWI3\_IRQn to SWI2\_IRQn.  
(Aligned with interrupt scheme used by S310 SoftDevice for cross compatibility).
- **ANT API SVC Changes**
  - `sd_ant_channel_radio_tx_power_set`
    - Additional parameter to specify custom (non-ANT indexed) TX power value. See API documentation for more information on usage.
  - `sd_ant_cw_test_mode`
    - Additional parameter to specify custom (non-ANT indexed) TX power value. See API documentation for more information on usage.
- **ANT API SVC Additions**
  - Reserved APIs to house future API calls. None currently assigned.
    - `SVC_ANT_RESERVED0`
    - `SVC_ANT_RESERVED1`
    - `SVC_ANT_RESERVED2`
  - Extended APIs to house future API calls using extended messages IDs.
    - `SVC_ANT_EXTENDED0`
    - `SVC_ANT_EXTENDED1`
    - `SVC_ANT_EXTENDED2`
- **SOC API SVC Changes**
  - `sd_power_perpower_set / sd_power_perpower_clr / sd_power_perpower_get / sd_power_perrdy_get`
    - Removed. PERPOWER register is not available on next generation (XLR2) NRF51 chip; therefore the SVC function has been removed. Refer, to PAN documentation (PAN-028 v1.6) “System: Manual setup is required to enable use of peripherals” for workaround to enable peripherals for current (XLR1) generation NRF51 chips.
- **SOC API SVC Additions**
  - `sd_power_gpregret_set / sd_power_gpregret_clr / sd_power_gpregret_get`
    - APIs added for access to general purpose retention register.
  - `sd_power_dcdc_mode_set`
    - Note: DC/DC converter may interfere with radio function. As a result, this API should not be modified by the application. The mode must not be set to `NRF_POWER_DC_DC_MODE_AUTOMATIC` or `NRF_POWER_DCDC_MODE_ON` at any time.
  - `sd_radio_notification_cfg_set`
    - Configures SoC Radio Notification feature. When enabled, signals application radio activity/inactivity.
  - `sd_ecb_block_encrypt`
    - The ECB block is a SoftDevice protected peripheral. This API allows the application to use the encryption engine when the SoftDevice is enabled.
  - `sd_radio_session_open / sd_radio_session_close / sd_radio_request`
    - Radio disabling scheduling feature. APIs allow the application to schedule pockets of time to run activities that do not overlap on top of radio activity. Useful for offsetting peak current draw activity. Radio protocol priority may override or delay this request.
  - `sd_evt_get`
    - SOC generated events are deposited into the SoC event queue. This API must be called by the application to retrieve the event when signaled by the softdevice via SWI2\_IRQn.
  - `sd_temp_get`
    - API call to retrieve the temperature measured on the chip.
  - `sd_flash_erase_page / sd_flash_write / sd_flash_protect`

- New APIs have been introduced to improve concurrent behavior of flash write and erase activities during radio and stack protocol operations. Radio protocol priority may override or delay requested flash operation.
- **Softdevice API SVC Changes**
  - Addition of 5 more options for LFCLK source. **Please note, if running RC option, LFCLK source for ANT should be set to NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_250MS\_CALIBRATION.** The new RC options are as follows:
    - NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_500MS\_CALIBRATION
    - NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_1000MS\_CALIBRATION
    - NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_2000MS\_CALIBRATION
    - NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_4000S\_CALIBRATION
    - NRF\_CLOCK\_LFCLKSRC\_RC\_250\_PPM\_8000MS\_CALIBRATION
  - `sd_softdevice_forward_to_application`
    - Added to support bootloader feature. Used to forward interrupts to the application and must be called before the bootloader starts the application.
- **SOC Event Software Interrupt**
  - SoftDevice SOC features will now generate certain events when enabled. For a full list of generated events, refer to NRF\_SOC\_EVTS in `nrf_soc.h` and `nrf_radio_disable.h`. Signals new events to application using SWI2\_IRQn. Events must be serviced by the application by calling `sd_evt_get` or else the device may not be able to be put into low power mode using `sd_app_evt_wait`.

## Limitations

- No known Limitations

## Known issues

- No known issues