S210 nRF51422

ANT™

SoftDevice Specification v0.6

Key Features

- Embedded ANT stack
  - Simple to complex network topologies:
    - Peer-to-peer
    - Practical mesh
    - Star
    - Tree
  - Supports up to 8 concurrent communication channels
  - Supports public, private, and managed networks
  - Supports ANT+ device profile implementations enabling multi-vendor interoperability
  - Interoperable with nRF24AP1, nRF24AP2 variants, and Dynastream ANT chipset/module based products
  - Broadcast, acknowledged, and burst communication modes
  - Configurable channel period 5.2 ms - 2 s
  - Built-in device search and pairing
  - Built-in interference handling
  - Built-in timing and power management
  - 60 kbps burst rate
- Asynchronous event-driven behavior
- Complementary nRF514 SDK including ANT profiles and example applications
- Low link-time dependencies
- No RTOS dependency
  - A RTOS of your choice can be used
- Safe application and protocol coexistence through memory isolation
- Standard ARM® Cortex™ -M0 project configuration for application development
- Thread-safe supervisor-call based API

Applications

- Personal area networks
  - Health/fitness sensors and monitoring devices
  - Medical devices
  - Key-fobs
- Active RFID
- Environmental sensor networks
- Home and industrial automation
- Logistics/goods tracking

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Life support applications

Nordic Semiconductor's products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

Contact details

For your nearest dealer, please see http://www.nordicsemi.com.
Information regarding product updates, downloads, and technical support can be accessed through your My Page account on our homepage.

Main office: Otto Nielsens veg 12
7052 Trondheim
Norway

Phone: +47 72 89 89 00
Fax: +47 72 89 89 89

Mailing address: Nordic Semiconductor
P.O. Box 2336
7004 Trondheim
Norway

RoHS and REACH statement

Nordic Semiconductor's products meet the requirements of Directive 2002/95/EC of the European Parliament and of the Council on the Restriction of Hazardous Substances (RoHS) and the requirements of the REACH regulation (EC 1907/2006) on Registration, Evaluation, Authorization and Restriction of Chemicals.

The SVHC (Substances of Very High Concern) candidate list is continually being updated. Complete hazardous substance reports, material composition reports and latest version of Nordic's REACH statement can be found on our website www.nordicsemi.com.

Revision History

Date	Version	Description
September 2012	0.6	

Contents

1 Introduction

S210 SoftDevice is an ANT protocol stack solution that provides a full, flexible application programming interface (API) for building ANT System on Chip (SoC) solutions for Nordic Semiconductor nRF51 ICs. S210 SoftDevice reduces the need for a secondary application host MCU to ANT solutions.

This document contains information about the SoftDevice features and performance which are subject to change between revisions.

1.1 Required reading

The nRF51422 Product Specification and the nRF51 Series Reference Manual are essential developer resources for ANT solutions from Nordic Semiconductor. The Software Architecture section of the nRF51 Series Reference Manual is essential reading for understanding the resource usage and performance related chapters of this document.

Knowledge of the ANT protocol is required to use the S210 correctly and to understand the terminology used in this document.

1.2 Writing conventions

This Product Specification follows a set of typographic rules to ensure that the document is consistent and easy to read. The following writing conventions are used:

- Command and event names, bit state conditions, and register names are written in `Lucida Console`.
- Pin names and pin signal conditions are written in **`Consolas.`**
- File names and user interface components are written in **bold**.
- Internal cross references are italicized and written in ***semi-bold.***
- Placeholders for parameters are written in *italics*. For example, a syntax description of SetChannelPeriod will be written as: SetChannelPeriod(*ChannelNumber, MessagingPeriod*).
- Fixed parameters are written in regular font. For example, a syntax description of SetChannelPeriod will be written as: SetChannelPeriod (0, Period).

2 Product overview

S210 SoftDevice is precompiled, pre-linked, and pre-programmed software implementing an ANT protocol stack. The S210 is compatible with selected nRF514xx System on Chip (SoC) devices. The application programming interface (API) is implemented through supervisor calls resembling regular C-functions. This enables complete application compiler and linker independence from the SoftDevice implementation. The SoftDevice enables the application programmer to develop their code as a standard ARM® Cortex™-M0 project without the need to integrate proprietary chip-vendor software frameworks. This means that any ARM Cortex™-M0 compatible toolchain can be used to develop ANT/ANT+ applications with the S210 SoftDevice.

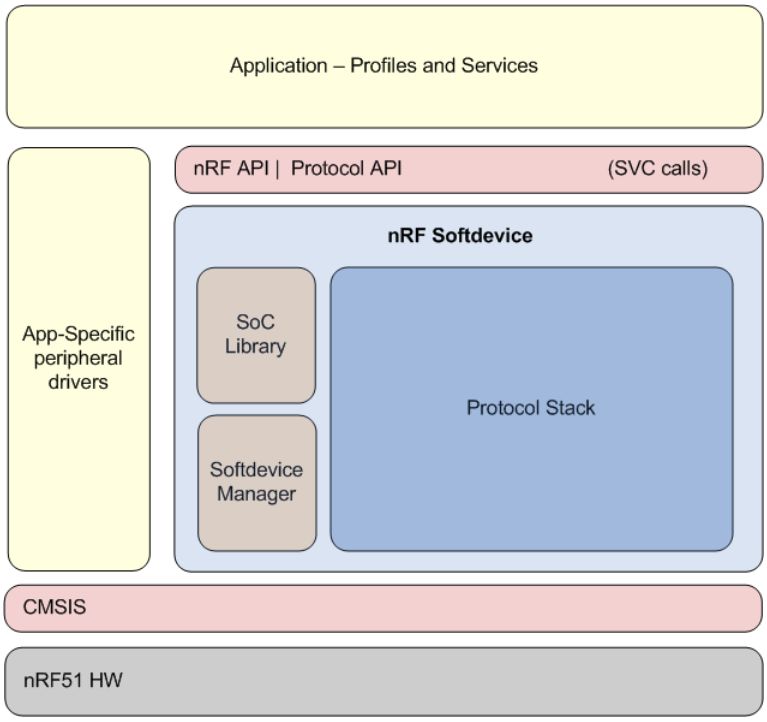


Figure 1 *System on Chip application with the SoftDevice*

This specification outlines features and support for the production version of the S210 SoftDevice. Pre-production versions may not have support for all features. To find information on any limitations or omissions, the S210 SoftDevice release notes will contain a detailed summary of the release status.

3 ANT Protocol Stack

S210 SoftDevice is a fully qualified ANT compliant stack that is pre-programmed on selected devices in the nRF514 family. S210 SoftDevice supports all ANT core functionality, including ANT+ profiles.
See http://thisisant.com for detailed information on ANT stack features.

Note: ANT+ implementations must pass the ANT+ certification process to receive ANT+ Compliance Certificates. See http://thisisant.com for further information.

Note: ANT+ network keys are needed to make ANT+ compliant products. The keys can be obtained by registering as an ANT+ adopter at http://thisisant.com.

3.1 Overview

The nRF514 Software Development Kit (SDK) supplements the ANT protocol stack with complete peripheral drivers, example applications, and ANT+ profile implementations.

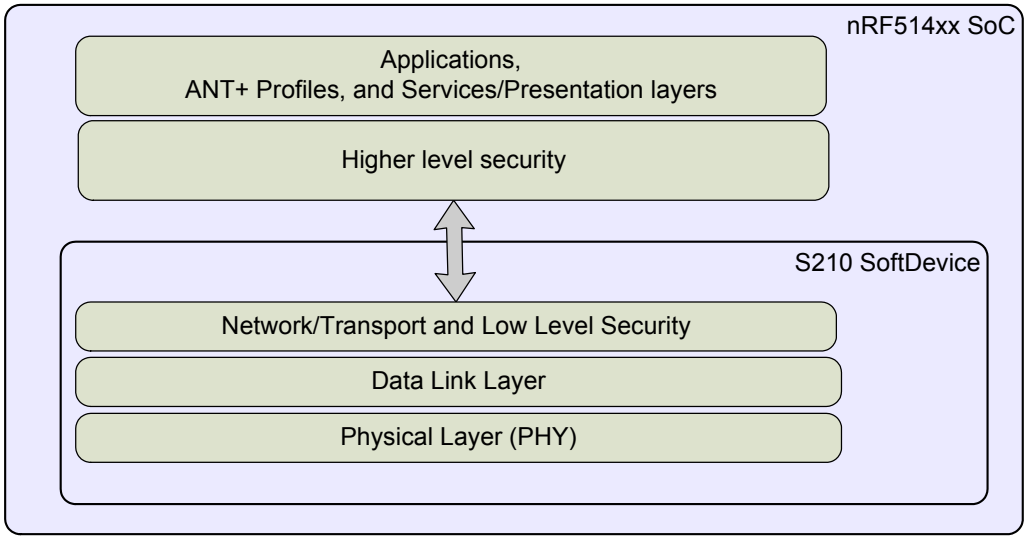


***Figure 2** ANT stack architecture*

3.2 ANT profile and feature support

All ANT nRF24AP1 and nRF24AP2 profiles and features are fully supported.

4 SoC library

The SoC library ensures that application access to SoC hardware resources used by the SoftDevice happens in a controlled and secured manner. The following API's are available in S210.

Feature	Notes
`nrf_mutex_` API	Atomic mutex API without the application needing to disable global interrupts.
`nrf_nvic_` API	Gives the application access to all NVIC features without corrupting SoftDevice configurations.
`nrf_rand_` API	Gives access to the true random number generator.
`nrf_power_` API	Safe power configuration interface.
`nrf_clock_` API	Safe clock configuration interface.
`nrf_wait_for_app_event`	Simple power management hook for the application.
`nrf_ppi_` API	Safe PPI access to dedicated application PPI channels.

5 SoftDevice manager

The SoftDevice manager allows the application to perform top level management of the SoftDevice operation (such as one command enabling or disabling the SoftDevice operation).

Feature	Notes
`nrf_softdevice_` API	Gives control of the SoftDevice state.

6 S210 Resource requirements

The S210 SoftDevice resides in the lower part of the SoC code memory space. When enabled by the nRF_SoftDevice API it reserves RAM space for its operation. Furthermore, it will control access to hardware resources used by the SoftDevice that then only can be accessed by the application using the SoC library API's. This section gives an overview of the resource allocations made by the S210 SoftDevice.

6.1 Memory resource map and usage

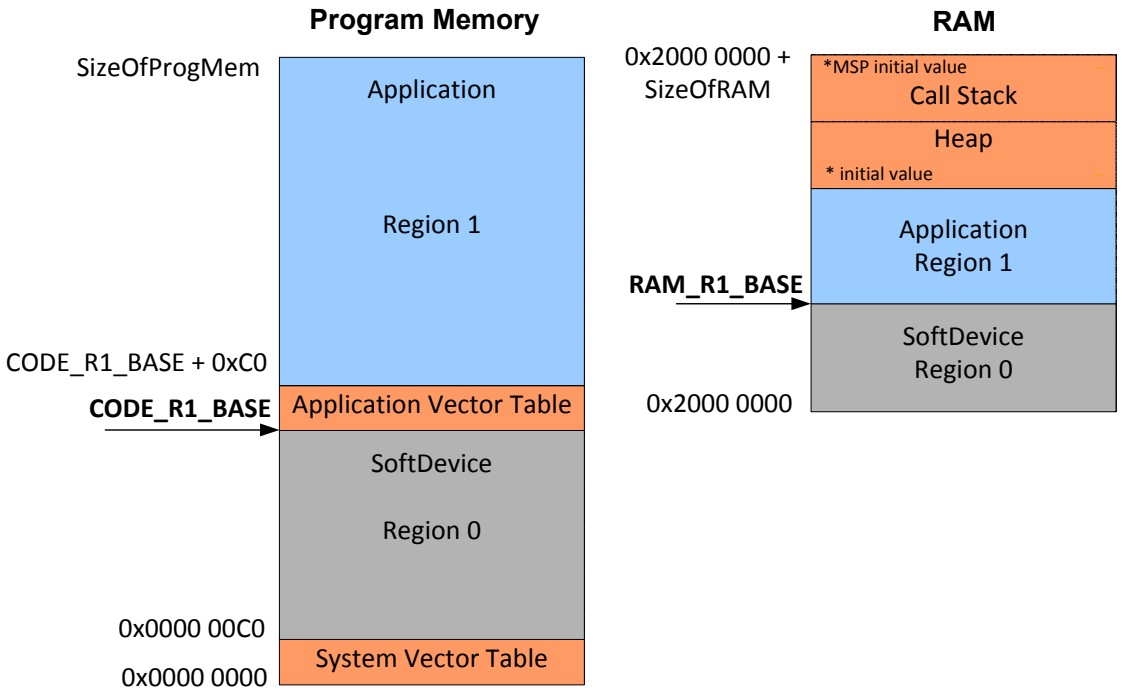


Figure 3 *Memory resource map*

Flash	S210 Enabled	S210 Disabled
Amount	32 kB	32 kB
CODE_R1_BASE	0x0000 A000	0x0000 A000

RAM	S210 Enabled	S210 Disabled
Amount	2 kB	0 kB
RAM_R1_BASE	0x2000 0800	0x2000 0000

Call stack	S210 Enabled	S210 Disabled
Maximum usage	384	0x00

Heap	S210 Enabled	S210 Disabled
Maximum allocated bytes	0 bytes (0x00)	0x00

Table 1 *Memory resource requirements*

6.2 Hardware blocks and interrupt vectors

Access type	Definition
Restricted	Used by the SoftDevice and outside the application sandbox. Application has restricted access via the SoftDevice API.
Blocked	Used by the SoftDevice and outside the application sandbox. Completely inaccessible to the application.
Open	Unused by the SoftDevice and in the application sandbox.

Peripheral protection and usage by SoftDevice				
ID	**Base address**	**Instance**	**Access (S210 enabled)**	**Access (S210 disabled)**
0	0x40000000	POWER	Restricted	Open
0	0x40000000	CLOCK	Restricted	Open
1	0x40001000	RADIO	Blocked	Open
2	0x40002000	UART0	Open	Open
3	0x40003000	SPIM0/ 2W0	Open	Open
4	0x40004000	SPI1/ 2W1	Open	Open
..			-	-
6	0x40006000	Port 0 GPIOTE	Open	Open
7	0x40007000	ADC	Open	Open
8	0x40008000	TIMER0	Open	Open
9	0x40009000	TIMER1	Open	Open
10	0x4000A000	TIMER2	Open	Open
11	0x4000B000	RTC0	Blocked	Open
12	0x4000C000	TEMP	Open	Open
13	0x4000D000	RNG	Restricted	Open
14	0x4000E000	ECB	Blocked	Open
15	0x4000F000	CCM	Blocked	Open
15	0x4000F000	AAR	Blocked	Open
16	0x40010000	WDT	Open	Open
17	0x40011000	RTC1	Open	Open
18	0x40012000	QDEC	Open	Open
..			-	-
20	0x40014000	Software interrupt	Open	Open
21	0x40015000	Software interrupt	Open	Open
22	0x40016000	Software interrupt	Open	Open
23	0x40017000	Software interrupt	Restricted[1]	Open
24	0x40018000	Software interrupt	Blocked	Open
25	0x40019000	Software interrupt	Blocked	Open
..			-	-
30	0x4001E000	NVMC	Open	Open
31	0x4001F000	PPI	Restricted	Open
NA	0x50000000	GPIO P0	Open	Open
NA	0x????	NVIC	Restricted[2]	Open

1. May be used by the application, but is also used for protocol to application events.
2. Not Sandbox protected. For robust system function, application must comply with the restriction.

Table 2 *Peripherals used by the SoftDevice*

6.2.1 PPI channels

Range	User	Notes
0 - 7	Application	Available for the application
8 - 15	SoftDevice	Used by the SoftDevice

6.2.2 PPI groups

Range	User	Notes
0	Application	Application configurable through the PPI API
1 - 3	SoftDevice	Used by the SoftDevice

6.2.3 SVC number ranges

Range	User	Notes
0x00 - 0x0F	Application	Always forwarded to the application
0x10 - 0xFF	SoftDevice	Not forwarded to the application

7 Performance

7.1 Interrupt latency

Latency, additional to ARM Cortex™-M0 hardware architecture latency, is introduced by SoftDevice logic to manage interrupt events.

Interrupt	Additional latency in CPU cycles
Open peripheral interrupt	50
Blocked or restricted peripheral interrupt (only forwarded when SoftDevice disabled)	63
Application SVC interrupt	14

7.2 Processor availability

Figure 4 illustrates the parameter values in ***Table 3*** on page 14. The SoftDevice architecture chapter of the nRF51 Reference Manual describes exception (interrupt) management in SoftDevices and is required knowledge for this section.

The parameters of interest are defined around LowerStack and UpperStack exceptions. These exceptions process real time protocol events and API calls (or deferred internal SoftDevice tasks) respectively.

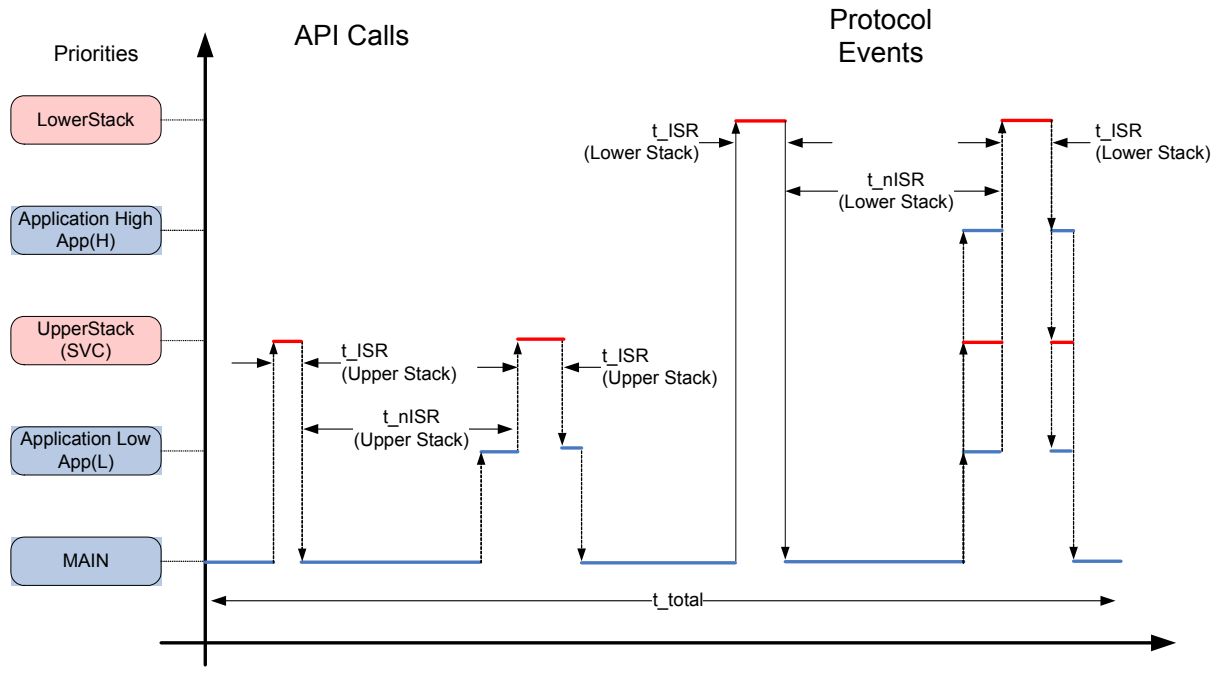


Figure 4 *Interrupt latencies due to SoftDevice processing*

		Lower stack			Upper stack		
Parameter	**Description**	**Min.**	**Nom.**	**Max.**	**Min.**	**Nom.**	**Max.**
t_ISR	Maximum interrupt processing time	-	TBD	400 µs	-	TBD	500 µs
t_nISR	Minimum time between interrupts	230 µs	-	-	[1]	-	-
duty_cycle	Maximum percentage time in interrupt	[2]	2	2	1	1	1

1. This data is application dependent. Calls to the SoftDevice API trigger the Upper stack interrupt.
2. This data is stack dependent.

***Table 3** S210 interrupt latency numbers*

$$duty\_cycle = \frac{\sum_{num\_ISRs}(t_{ISR})}{t_{total}}$$

Application Low, App(L), can be blocked for a maximum of:

$$App(L)_{latency_{\max}} = t_{ISR_{\max(Upper\ Stack)}} + t_{ISR_{\max(Lower\ Stack)}}$$

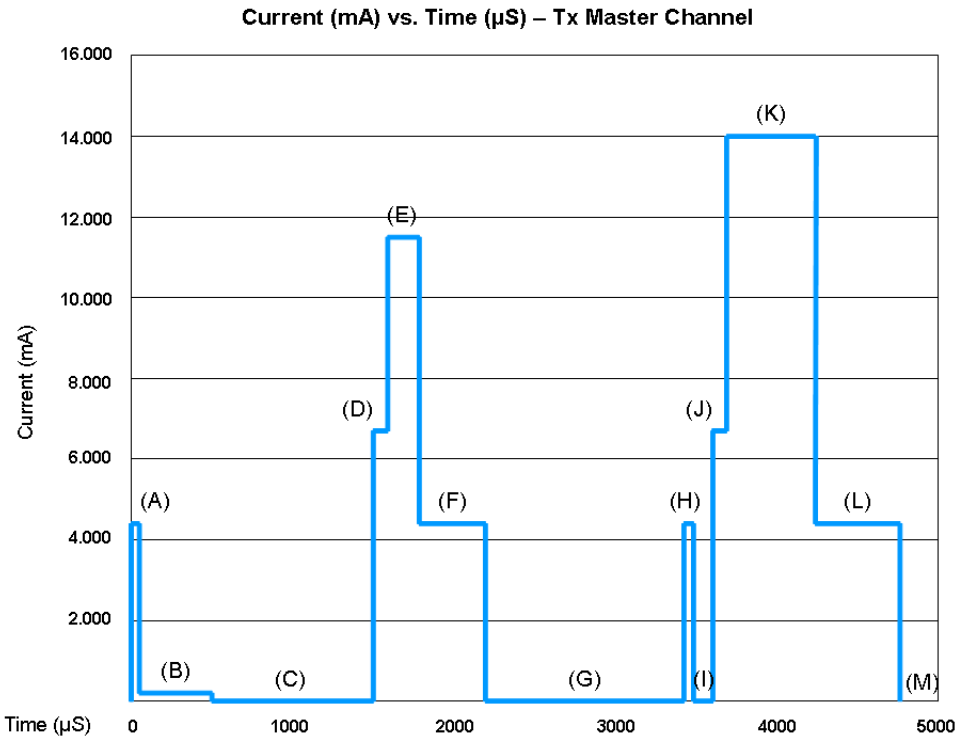
Application High, App(H), can be blocked for a maximum of:

$$App(H)_{latency\_\max} = t_{ISR_{\max(Lower\ Stack)}}$$

8 Power profiles

These profiles give a detailed overview of the stages of a radio event, the approximate timing of stages within the event, and how to calculate the peak current at each stage.

8.1 TX master channel

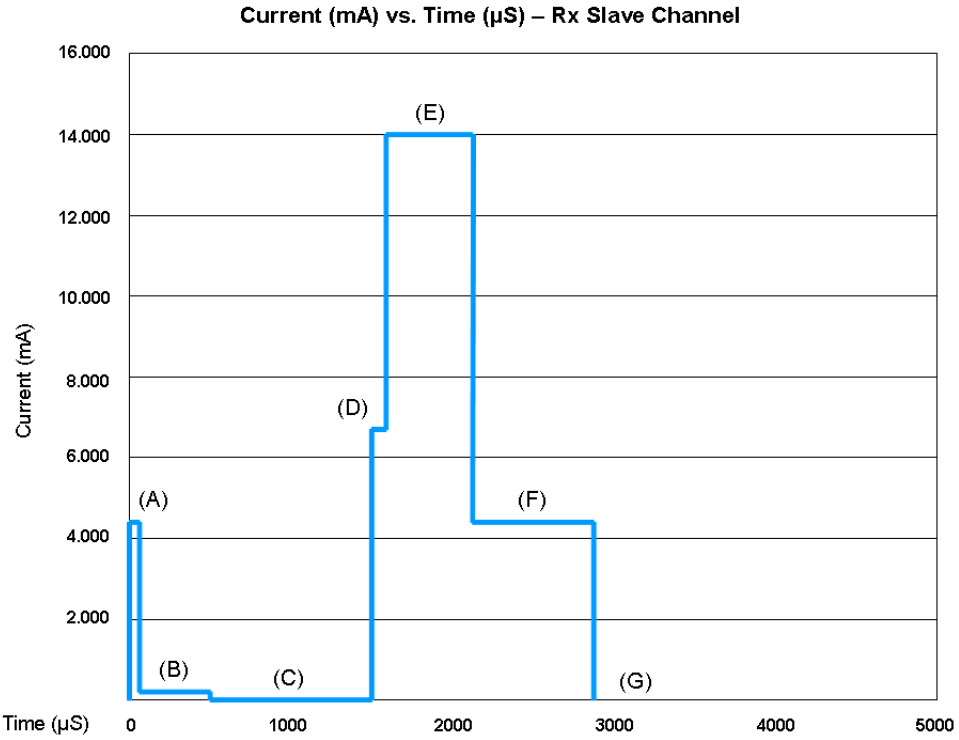


Stage	Description	Current Calculations[1]
(A) and (H)	Preprocessing	$I_{RTC} + I_{X32k} + I_{CPU,Flash}$
(B)	Standby + XO ramp	$I_{RTC} + I_{X32k} + I_{START,X16M}$
(C), (G) and (I)	Standby	$I_{RTC} + I_{X32k} + I_{X16M}$
(D)	Radio Start	$I_{RTC} + I_{X32k} + I_{X16M} + \int (I_{START,TX})$
(E)	Radio TX	$I_{RTC} + I_{X32k} + I_{X16M} + I_{TX,0dBm}$
(F)	Post-processing	$I_{RTC} + I_{X32k} + I_{X16M} + I_{CPU,Flash}$
(J)	Radio turn-around/Start	$I_{RTC} + I_{X32k} + I_{X16M} + \int (I_{START,TX})$
(K)	Radio RX	$I_{RTC} + I_{X32k} + I_{X16M} + I_{RX}$
(L)	Post-processing	$I_{RTC} + I_{X32k} + I_{CPU,Flash}$
(M)	Idle - TX channel	$I_{RTC} + I_{X32k}$

1. See the nRF51422 Product Specification for the symbol values.

Note: I_{RC32k} should be substituted for I_{X32k} when using the 32k RCOSC

8.2 RX channel



Stage	Description	Current Calculation[1]
(A)	Pre-processing	$I_{RTC} + I_{X32k} + I_{CPU,Flash}$
(B)	Standby + XO ramp	$I_{RTC} + I_{X32k} + I_{START,X16M}$
(C)	Standby	$I_{RTC} + I_{X32k} + I_{X16M}$
(D)	Radio start	$I_{RTC} + I_{X32k} + I_{X16M} + \int (I_{START,RX})$
(E)	Radio RX	$I_{RTC} + I_{X32k} + I_{X16M} + I_{RX}$
(F)	Post-processing	$I_{RTC} + I_{X32k} + I_{CPU,Flash}$
(G)	Idle - RX channel	$I_{RTC} + I_{X32k}$

1. See the nRF51422 Product Specification for the symbol values.

Note: I_{RC32k} should be substituted for I_{X32k} when using the 32k RCOSC

9 SoftDevice compatibility and selection

9.1 S210 SoftDevice identification and version scheme

S210 SoftDevice will be identified by the S210 part code, a qualified device part code (e.g. nRF51422), and a version number consisting of major, minor, and revision numbers.

For example: S210_nRF51422_1.2.3

Table 4 outlines how version numbers will increment.

Revision	Description
Major increments	Modifications to the API or the function or behavior of the implementation or part of it have changed. Changes as per minor increment may have been made. Application code will not be compatible without some modification.
Minor increments	Additional features and/or API calls are available. Changes as per revision increment may have been made. Application code may be modified to take advantage of new features.
Revision increments	Issues have been resolved or improvements to performance implemented. Existing application code will not require any modification.

Table 4 *Revision scheme*

Additionally, for revisions of the SoftDevice which are not production qualified, the qualification level, alpha or beta, and sequence number will be appended. The following are examples.

- An alpha qualified revision: S210 nRF51422 1.2.3alphaX
- The same SoftDevice with beta qualification: S210 nRF51422 1.2.3betaY
- The same SoftDevice with production qualification: S210 nRF51422 1.2.3

Table 5 outlines the test qualification levels.

Qualification	Description
Alpha	Development release suitable for prototype application development. • Hardware integration testing is not complete. • Known issues may not be fixed between alpha releases. • Incomplete and subject to change.
Beta	Development release suitable for application development. In addition to alpha qualification: • Hardware integration testing is complete. • Stable, but may not be feature complete and may contain known issues. • Protocol implementations are tested for conformance and interoperability.
Production	Qualified release suitable for product integration. In addition to beta qualification: • Hardware integration tested over supported range of operating conditions. • Stable and complete with no known issues • Protocol implementations conform to standards

Table 5 *Test qualification levels*

9.2 nRF514xx device compatibility and qualification

S210 SoftDevice is compatible with one or more nRF514xx devices and is qualified for mass production with some of these devices. The following is a list of S210 versions versus compatible devices and qualification status current at the time of publication of this document.

SoftDevice	Compatible IC	Qualification
S210_nRF51422_1.0.1.alpha	nRF51x22	Alpha