nRF52 SDK

A brief introduction – Global Tech Tour 2015

About these slides

- What is the nRF52 SDK
- Why you want to use it
- Features highlights
- Product Specific SDKs
- Summary

SDK in numbers

- nRF52 SDK is a collection of code to make nRF52 development easy

Drivers 15+	Libraries 10+	Modules 7+	Example applications	Externals
TWI	CRC	NFC	Hardware Peripheral 40+	SoftDevices
UART	FIFO	BLE Security manager	Bluetooth 20+	MDK
PWM	HASH	DFU	ANT 10+	RTOSes

Compiler and IDE support

- Support for multiple compilers: IAR, Keil and GCC.
- Not locked to a Proprietary IDE. Use whatever OS and IDE you want.
  - some guides are shared on dev-zone.
  - Keil as our supported reference IDE

Where to get it

- Free and easy to get
  - Zip developer.nordicsemi.com
  - Documentation infocenter.nordicsemi.com
  - CMSIS-packs from Keil 5 Pack Installer

- Meant to be used together with the nRF52 Development kits, but switching to your production PCB or other boards is intuitive.

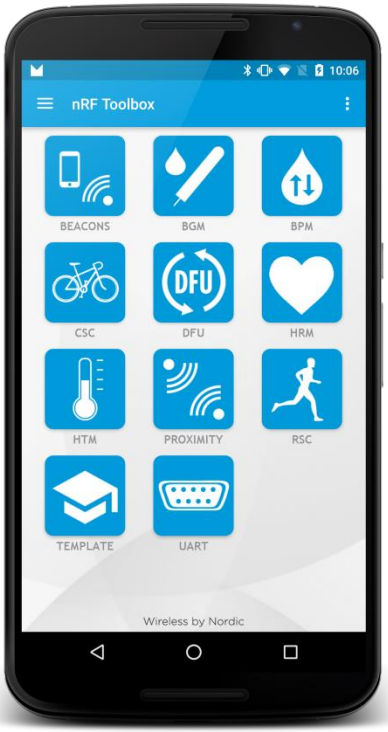
Why use the nRF52 SDK

- Speed up development
- Compilable and runnable code ready to be expanded upon
- Code is written by a team with deep knowledge of the hardware
- Stay updated on the latest developments – New SoftDevices, drivers, modules, services and profiles

- Tech Support knows the product well, and respond quickly
- Great community on Nordic Developer Zone
- Same codebase used for other Nordic SDKs

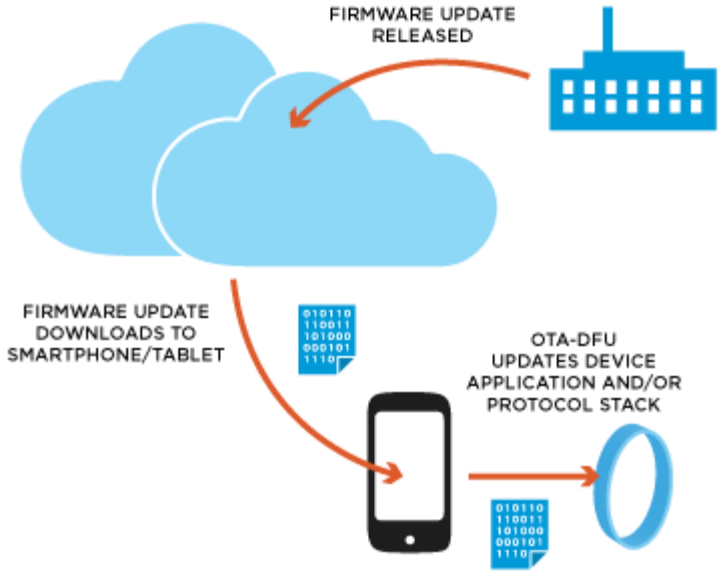
Why use the nRF52 SDK

- Complemented by software for iOS, Android
- Master Control Panel for both PC and Android
- Available on Play-and App store, and as BSD on GitHub!



DFU

- Over the air wireless device firmware update
- Bootloader with custom BLE service
- Upload any combination of Application, SoftDevice or new bootloader

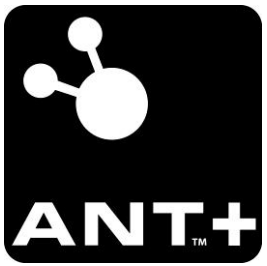


DFU with Asymmetric Signing

- Add-on to regular DFU implementation
  - Uses same custom BLE service
  - Easy to integrate
- Firmware images signed with elliptic curve cryptography
  - ECDSA using elliptic curve P-256 and SHA-256
  - Popular “footprint vs security” ratio
- Signing secures the OTA firmware upgrades
  - Trusted source
  - Trusted operation
- Based on Private/Public key-pair
  - Nobody can create a valid firmware image without access to the private key
  - Public keys are stored in flash
- Tools provided to generate keys
  - Python based for automated operation

Peripherals and Utility

- Abundance of example applications.
- Hardware peripheral drivers
  - HAL ensures easy migration from nRF51
  - All hardware has at least one example to see the driver in use.
- Generic utility modules and Libraries.
  - Flash storage management
- Common RTOS support (RTX and FreeRTOS)
  - RTOS agnostic



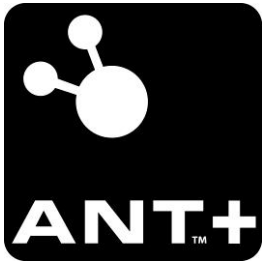
BLE

- Wide range of qualified Bluetooth LE services and profiles
  - Heart Rate, HID, Alert Notification, Glucose ++
- Proprietary BLE services
  - UART, DFU, BLE Blinky
- Bluetooth LE modules
  - Security management
  - Advertising management
- Direct Test Mode (DTM)
- DFU over BLE and UART
- NFC touch to pair

ANT

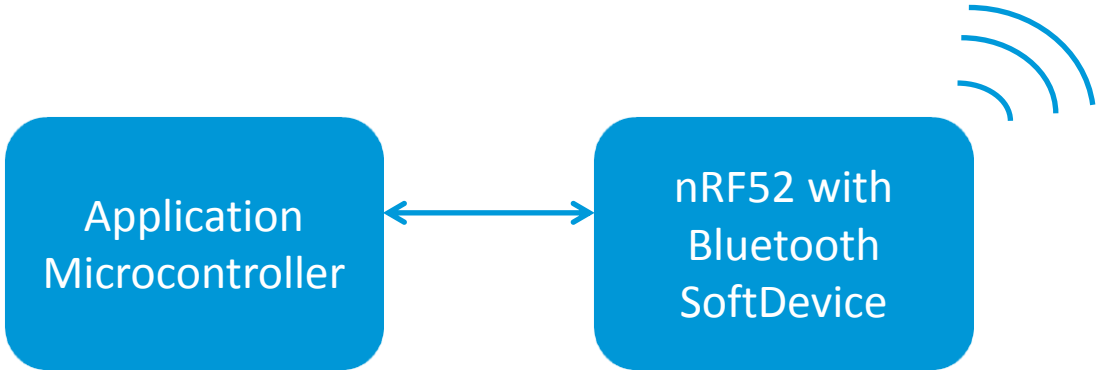
- Wide range of ANT profiles and examples
  - Heart Rate, Bicycle Power, Bike Speed & Cadence and more
- Support for up to 15 ANT channels
- ANT modules
  - File sharing
  - Encryption
- ANT DFU

- Proprietary radio protocol library (ESB, Gazell)



Serialization

- Bluetooth LE SoftDevice serialization, converting our nRF52 into a connectivity chip



Product specific SDK's

- Some SDK do live their own life outside of the generic nRF52 SDK,
- All based on nRF52 SDK, and holds same basic code base
- Separate because of product maturity (and/or licensing)

Rezence SDK	IoT SDK	SDK for HomeKit
		

Rezence SDK

- Specialized SDK to target wireless charging
- Rezence: Wireless standard that uses BLE as signaling protocol

- Feature complete A4WP Baseline System Specification (BSS) 1.3
- Device Firmware Update (DFU) support
- Simple device integration
- Reference design 2016 Q1

Available from http://www.rezence.com/

IoT SDK

- SDK where core functionality is to enable IPv6 over BLE
  - 6LoWPAN compression/decompression
- 100% based on open standards
- Security through well tested TLS and DTLS libraries
- Support and examples for major M2M protocols:
  - CoAP
  - MQTT
  - LWM2M
- Examples for Linux based router
- Utilizes the strength of BLE support in mobile devices to ensure user friendly deployment.

MQTT.ORG

oma Open Mobile Alliance For a Connected World

CoAP

SDK for HomeKit*

- SDK that enables Single-Chip HomeKit BLE solution
- HomeKit: Communicate with and control accessories in a home.
- SDK Approved by Apple
- Software to support all features needed for
  - Highly optimized HomeKit Application Protocol (HAP)
  - HomeKit profiles
  - MFI authentication
- Device Firmware Update (DFU) support
- Cross platform support

**Only available to verified MFI members*

Summary

- nRF52 SDK aims to make development fast and fun.
- Drivers, libraries, modules, MDK, SoftDevices, example applications
- ANT and Qualified Bluetooth LE examples.
- Powerful features such as the Device Firmware Upgrade & NFC touch to Pair.
- Helps you stay updated on valuable nRF52 features

- Several Compilers supported
- Custom SDKs for specific use cases
- Excellent Tech Support and Community

Available for free at developer.nordicsemi.com