Developing Thread products using nRF Connect SDK

Nordic Tech Webinar

Jørgen Holmefjord / Application Engineer
Łukasz Duda / Senior Firmware Engineer
November 2020

1

Today's hosts

Jørgen Holmefjord

Application Engineer
Technical Support

NORDIC SEMICONDUCTOR

Łukasz Duda

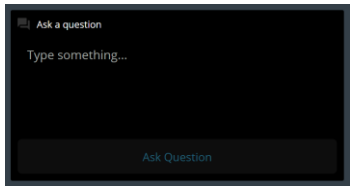
Senior Firmware Engineer
Thread Team Lead

NORDIC SEMICONDUCTOR

2

Practicalities

- Duration: about 60 minutes
- Questions are encouraged!
  - Please type questions in the top of the right sidebar
  - All questions are anonymous
  - Try to keep them relevant to the topic
  - We will answer towards the end
- The chat is not anonymous, and do not use for questions
- Go to DevZone if you have more questions
- A recording of the webinar will be available together with the presentation at webinars.nordicsemi.com



{ DevZone

3

Introduction to Thread

Available introductory webinar:

- Introduction to Thread
  Pär Håkansson and Krzysztof Loska

NORDICTECH WEBINARS

Missed this webinar?
Sign up: **nordicsemi.com/webinars**

4

Agenda

- Introduction (10 minutes)
  - Introduction to software solution
  - Supported architectures
- Getting started (10 minutes)
  - Development tools
  - Setting up the toolchain
  - Building your first Thread application in nRF Connect SDK
- Coding examples (30 minutes)
  - Building applications from command line
  - Configuring OpenThread and project options
- Q&A session (10 minutes)

5

Introduction

Introduction to software solution

6

OpenThread project and Nordic involvement

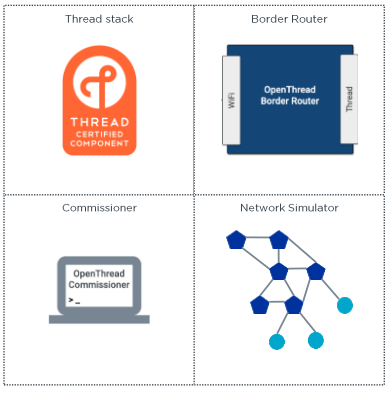
- OpenThread released by Google is an open-source implementation of Thread networking protocol and other necessary components
- Nordic joined OpenThread project from the beginning of its public existence in 2016
- Nordic provides full support for its devices within OpenThread and actively participates in the development of OpenThread by co-authoring some core modules
- Licensed under 3-clause BSD license
- Open-source, community-based project hosted on GitHub: https://github.com/openthread/

OPENTHREAD
released by Google

7

OpenThread key components

- Key components:
  - Thread stack – implementation of Thread 1.1.1 and Thread 1.2 specifications
  - Border Router – implementation of Thread Border Router
  - Commissioner – implementation of Thread Commissioner
  - Network Simulator – simulate large scale Thread networks
- More information: https://openthread.io



OPENTHREAD
released by Google

8

Nordic's Thread offering – Software

nRF5 SDK for Thread & Zigbee

Comprehensive SDK

Market-proven

Support up to Thread 1.1.1

Under maintainance

nRF Connect SDK

New SDK from Nordic

Modern tools

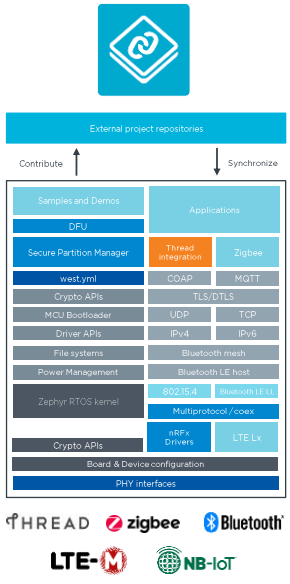
Support for Thread 1.2 and beyond

Under active development

9

Complete software solution

- Based on nRF Connect SDK
- OpenThread stack integration
  - Support for pre-compiled libraries for Thread Certification by inheritance
- Thread examples presenting variety of IP-based application layers and architectures
- Support for protocols co-existence
  - For example, Thread with Bluetooth 5.0
- Advanced build system
- Security components
- Drivers and generic libraries



10

Introduction

Supported architectures

12

Single-chip, single protocol (SoC)

- Single-chip solution
- Thread stack and application runs on same processor
- Low-cost, low-power, and simple design
- Recommended nRF devices:

nRF5340

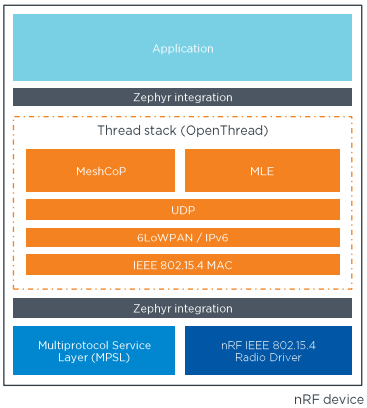
Flash: 1 MB+256 kB
RAM: 512 kB+64 kB
Temp: -40 to 105 °C

nRF52840

Flash: 1 MB
RAM: 256 kB
Temp: -40 to 85 °C

nRF52833

Flash: 512 kB
RAM: 128 kB
Temp: -40 to 85 °C



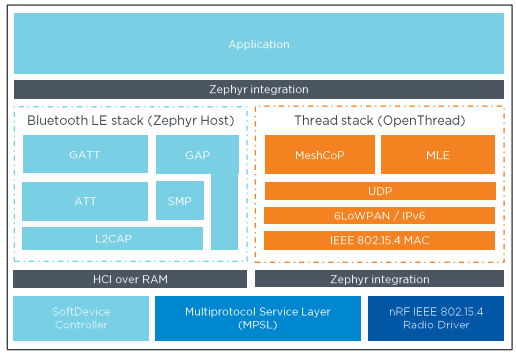
nRF device

13

Single-chip, multiprotocol (SoC)

- Single-chip solution supporting simultaneous 802.15.4 and Bluetooth LE operations
- An application, Thread and Bluetooth LE stacks run on the same processor
- Low-cost and low-power consumption
- Recommended nRF devices:

nRF5340	nRF52840	nRF52833
Flash: 1 MB+256 kB RAM: 512 kB+64 kB Temp: -40 to 105 °C	Flash: 1 MB RAM: 256 kB Temp: -40 to 85 °C	Flash: 512 kB RAM: 128 kB Temp: -40 to 85 °C



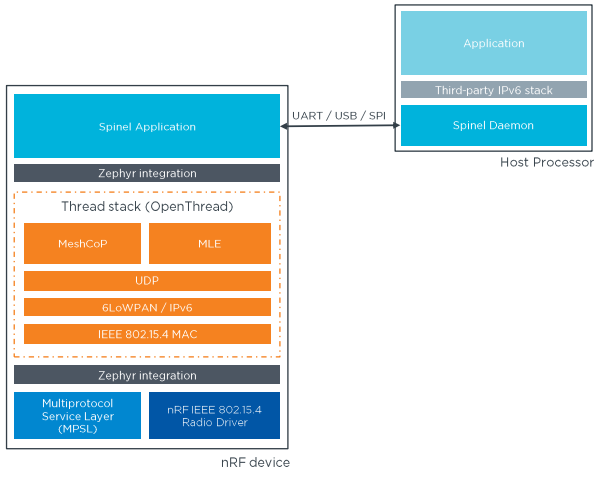
nRF device

14

Network Co-Processor (NCP)

- Used most for Border Router and Gateway designs
- Thread stack runs on the Nordic device
- An application runs on host processor
  - NCP controllers: wpantund, pyspinel
- Clear logical separation
- Recommended nRF devices:

nRF5340	nRF52840	nRF52833
Flash: 1 MB+256 kB RAM: 512 kB+64 kB Temp: -40 to 105 °C	Flash: 1 MB RAM: 256 kB Temp: -40 to 85 °C	Flash: 512 kB RAM: 128 kB Temp: -40 to 85 °C



nRF device

15

Radio Co-Processor (RCP)

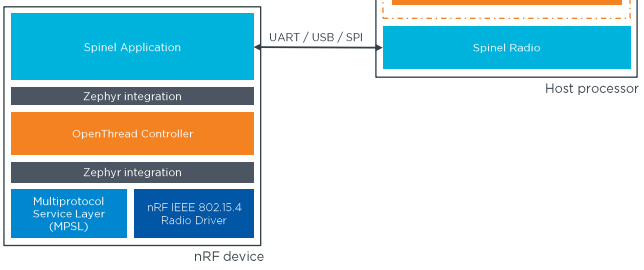
- Used most for Border Router and Gateway designs or devices that have other processing demands like IP cameras
- Thread stack and application layer run on a host processor
- Minimal controller runs on the Nordic device
- Recommended nRF devices:

nRF52811

Flash: 192 kB
RAM: 24 kB
Temp: -40 to 85 °C

nRF52820

Flash: 256 kB
RAM: 32 kB
Temp: -40 to 105 °C



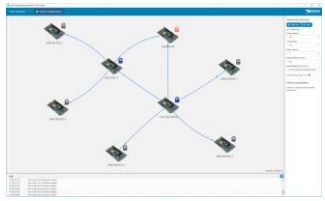
16

Getting started

17

Development tools

- Hardware Development Kits
- Thread Topology Monitor
  - Visualization of Thread network
  - Use CLI firmware to communicate between PC and Thread network
- nRF Sniffer for 802.15.4
  - Capture on-air packets in IEEE 802.15.4 based networks
  - Supported devices: nRF52840 DK & Dongle
- SEGGER Embedded Studio (Nordic Edition)
- Debugging tools: e.g. SEGGER Ozone, J-Link GDB



18

Setting up the toolchain

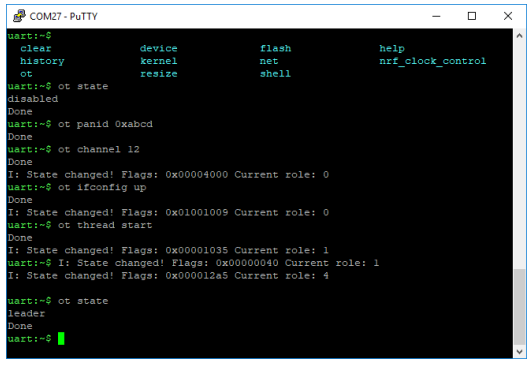
- nRF Connect for Desktop Toolchain Manager
  - Simplified installation for Windows
  - Experimental support for macOS
- nRF Connect Getting Started Assistant
  - Supported for Linux and macOS
- Manual installation

Hands-on

19

Thread CLI example application

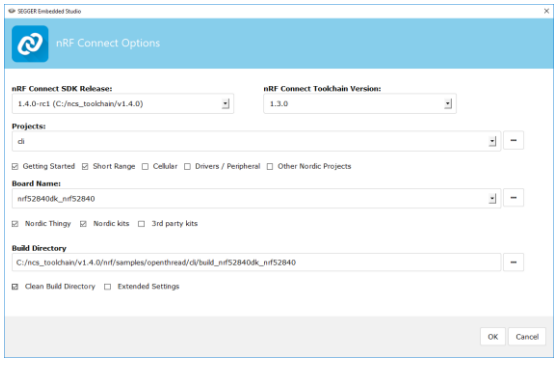
- Demonstrates a minimal OpenThread application that uses a basic command-line interface to expose:
  - OpenThread configuration
  - Data interfaces
- Translates ASCII commands to OpenThread API
- Serial transport: UART or USB CDC
- CLI is used as THCI in Certification Thread Test Harness



20

Building your first Thread application

- Steps:
  1. Open existing CLI example project in SES (Nordic Edition).
  2. Build and flash the example.
  3. Form a simple topology from two nodes.
  4. Use nRF Thread Topology Monitor to visualize the topology.
  5. Use nRF Sniffer for 802.15.4 to sniff over-the-air traffic.



Hands-on

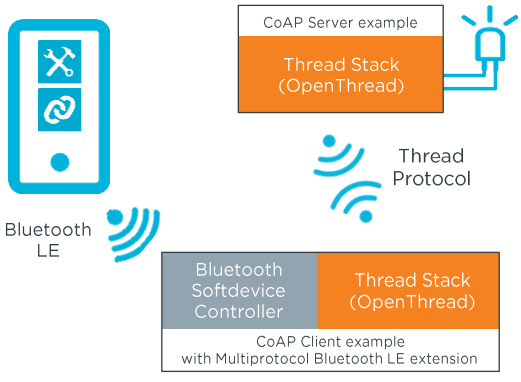
21

Coding examples

22

Building applications from command-line

- Steps:
  1. Open existing OpenThread project.
  2. Use west to build CoAP examples, including multiprotocol.
  3. Control CoAP server LED state from phone using Nordic UART Service (NUS).



Hands-on

23

Configuring OpenThread options

- Steps:
  1. Modify Thread operational parameters, such as Mode (FTD/MTD), Channel, etc.
  2. Configure OpenThread and Zephyr logging.
  3. Enable the optional OpenThread features, such as Thread 1.2 support.
  4. Change UART pins.

```
# Thread network configuration options

# Copyright (c) 2020 Nordic Semiconductor ASA
# SPDX-License-Identifier: Apache-2.0

config OPENTHREAD_PANID
        int "Default PAN ID"
        default 43981

config OPENTHREAD_CHANNEL
        int "Default Channel"
        default 11

config OPENTHREAD_XPANID
        string "Default Extended PAN ID"
        default "de:ad:00:be:ef:00:ca:fe"
        help
          Extended PAN ID for OpenThread with
          format "de:ad:00:be:ef:00:ca:fe"
```

Hands-on