

# QSG170: Silicon Labs OpenThread Quick-Start Guide

This document describes how to get started with OpenThread development using the Silicon Labs OpenThread software development kit (SDK) and Simplicity Studio<sup>®</sup> 5 with a compatible wireless starter kit.

#### KEY POINTS

- About the Silicon Labs OpenThread SDK
- About Example Applications and Demos
- Getting started with development
- Next steps
- Development tools

## 1 Introduction

Google's OpenThread is an open-source implementation of Thread. Google has released OpenThread to make the networking technology used in Google Nest products more broadly available to developers, in order to accelerate the development of products for the connected home and commercial buildings.

With a narrow platform abstraction layer and a small memory footprint, OpenThread is highly portable. It supports both system-on-chip (SoC) and radio co-processor (RCP) designs. Note that the network co-processor (NCP) design model is not supported or tested, although experimental support is available with the OpenThread stack. OpenThread implements all features defined in the Thread 1.3.0 specification and the draft 1.3.1 specification (backward compatible with both the 1.1 and 1.2 specifications). This specification defines an IPv6-based reliable, secure, and low-power wireless device-to-device communication protocol for home and commercial building applications.

This guide describes how to get started developing OpenThread applications using the Silicon Labs OpenThread SDK, delivered with the Gecko SDK, and Simplicity Studio 5. Simplicity Studio 5 (SSv5) includes everything needed for IoT product development with Silicon Labs devices, including a resource and project launcher, software configuration tools, full IDE with GNU toolchain, and analysis tools. This document focuses on use and development in the Simplicity Studio 5 environment. Alternatively, the Gecko SDK may be installed manually by downloading or cloning the latest from GitHub. See <a href="https://github.com/SiliconLabs/gecko">https://github.com/SiliconLabs/gecko</a> sdk for more information.

Our newest embedded software platform, the Simplicity Software Development Kit (SDK), is designed specifically for our Series 2 and Series 3 devices. The Simplicity SDK is a significant advancement in IoT development, providing our customers with a cohesive development environment for wireless innovation. It enables wireless developers to utilize advanced features and capabilities with the most recent Silicon Labs IoT devices. See <a href="https://github.com/SiliconLabs/simplicity\_sdk">https://github.com/SiliconLabs/simplicity\_sdk</a> for more information.

Meanwhile, our Gecko Software Development Kit (GSDK) will continue to be available for users of our Series 0 and Series 1 devices. For additional information on Series 0 and Series 1 devices please reference: Series 0 and series 1 EFM32/EZR32/EFR32 device (silabs.com).

#### 1.1 About the Silicon Labs OpenThread SDK

The Silicon Labs OpenThread SDK is based on the Gecko Platform component-based design, where each component provides a specific function. Components are made up of a collection of source files and properties. The component-based design enables customization by adding, configuring, and removing components. The application developer can use SSv5's Project Configurator and Component Editor to easily assemble the desired features by including those components that match the required functionality and by configuring the various properties associated with those components.

The Silicon Labs OpenThread SDK is based on the OpenThread stack available in GitHub but with a number of enhancements, including support of additional platforms and functionality, additional example applications, and additional metadata to allow for the seamless integration into SSv5.

**Platforms and functionality**: Silicon Labs has enhanced the OpenThread source code and the platform abstraction layer supporting the EFR32 platform in order to offer additional features provided by the Gecko Platform, such as:

- Wi-Fi coexistence
- Antenna diversity
- FreeRTOS support
- Non-Volatile Memory storage support (NVM3)
- Hardware acceleration with MbedTLS
- Multi-PAN Radio Co-Processor (RCP)
- Co-Processor Communication support (CPC)
- PSA Vault support
- Proprietary Sub-GHz
- Power manager support
- TrustZone support

**Example applications:** The example applications from GitHub are included in the SDK, along with new example applications showcasing additional functionality, such as OpenThread/Bluetooth Low Energy Dynamic Multiprotocol support or Sleepy Device behavior (see the sleepy device section in <u>Developing and Debugging Silicon Labs OpenThread Applications</u> for more information). The examples are described in section 2 About Example Applications and Demos.

The Silicon Labs OpenThread SDK contains the complete OpenThread GitHub directory structure but does not use the makefile build system provided by the GitHub solution. Instead, the Silicon Labs OpenThread SDK uses the build system provided by SSv5. The GitHub

makefile build options have been made available as component configuration options and are discussed in section 3.2 Configuring the Project. When using the Silicon Labs OpenThread SDK to build a Host-RCP application (such as an OpenThread Border Router), make sure to use compatible RCP and host code from the same SDK.

See the release notes available in Simplicity Studio and through <u>docs.silabs.com</u> for a description of the special features and the Open-Thread stack version used for the release.

#### 1.1.1 Simplicity Studio 5 (SSv5)

SSv5 is the core development environment designed to support the Silicon Labs IoT portfolio of system-on-chips (SoCs) and modules. It provides access to target device-specific web and SDK resources; software and hardware configuration tools; an integrated development environment (IDE) featuring industry-standard code editors, compilers and debuggers; and advanced, value-add tools for network analysis and code-correlated energy profiling.

SSv5 is designed to simplify developer workflow. It intelligently recognizes all Silicon Labs evaluation and development kit parts and, based on the selected development target, presents appropriate software development kits (SDKs) and other development resources.

The Silicon Labs OpenThread SDK is downloaded through SSv5. The GNU Compiler Collection (GCC) is provided with SSv5. Other important development tools provided with SSv5 are introduced in section 5 Development Tools.

#### 1.1.2 Gecko Bootloader

A bootloader is a program stored in reserved flash memory that can initialize a device, update firmware images, and possibly perform some integrity checks. Silicon Labs networking devices use bootloaders that perform firmware updates in two different modes: standalone (also called standalone bootloaders) and application (also called application bootloaders). An application bootloader performs a firmware image update by reprogramming the flash with an update image stored in internal or external memory. Silicon Labs recommends that you always flash a bootloader image along with your application, so that flash memory usage is appropriately allocated from the beginning. For more information about bootloaders see *UG103.6: Bootloader Fundamentals*.

#### 1.1.3 Gecko Platform

The Gecko Platform is a set of drivers and other lower layer features that interact directly with Silicon Labs chips and modules. Gecko Platform components include EMLIB, EMDRV, RAIL Library, NVM3, and MbedTLS. For more information about Gecko Platform, see release notes that can be found in SSv5's Documentation tab, as well as online API documentation in <a href="https://docs.silabs.com/">https://docs.silabs.com/</a>.

#### 1.2 Prerequisites

Before following the procedures in this guide you must have:

- Purchased one of the Wireless Gecko (EFR32) Portfolio Wireless Kits.
- Downloaded SSv5 and the Gecko SDK, which includes the Silicon Labs OpenThread SDK, and be generally familiar with the SSv5 Launcher perspective. SSv5 installation and getting started instructions along with a set of detailed references can be found in the online Simplicity Studio 5 User's Guide, available on <a href="https://docs.silabs.com/">https://docs.silabs.com/</a> and through the SSv5 help menu.

For a compiler, use the version of the GNU ARM toolchain installed with the Silicon Labs OpenThread SDK. The IAR Embedded Workbench for ARM (IAR EWARM) Compiler is currently not supported with OpenThread.

### 1.3 Support

Access the Silicon Labs support portal at <u>https://www.silabs.com/support</u> through SSv5's Welcome view under Learn and Support. Use the support portal to contact Customer Support for any questions you might have during the development process.

| Eile Edit Navigate Search Project Bun Window Help  |  |                              |  |  |  |  |
|--|--|------------------------------|--|--|--|--|
| ♠ Welcome ④ Recent 🏢 Tools 🛃 Install 🏟 Preferences                                       |  | A Launcher {} Simplicity IDE |  |  |  |  |
| 📑 Debug Adapters 🔅 🗖   |  |                              |  |  |  |  |
| S ★ C → K → K → C → C → C → C → C → C → C →  | Welcome to Simplicity Studio<br>Everything you need to develop, research, and configure devices for IoT applications.  |                              |  |  |  |  |
|  | Get Started<br>Select a connected device or search for a product by name to see available documentation, example projects, and demos.  |                              |  |  |  |  |
|  | Connected Devices         All Products   |                              |  |  |  |  |
|  | Recent Projects  |                              |  |  |  |  |
| 🗅 My Products 🗳 🕈 🕷 🖻 🕀 🖓 🗌  | Recent Projects<br>ot-cli-ftd • Open   |                              |  |  |  |  |
| Enter product name  C My Products 1  C My EFR32MG12 2.4 GHz 19 dBm Radio Board (BRD4161A |  |                              |  |  |  |  |
| EFR32MG12P432F1024GL125  | ∧ Learn and Support  |                              |  |  |  |  |
|  | Simplicity Studio User's Guide     OPEN     OPEN     Or collection of Simplicity Studio 5 User's Guide   | START                        |  |  |  |  |
|  | *Ö* Tips and Tricks         Useful tips and tricks to help you optimize your tools setup         START         Where users come together to learn, get help, and grow their skills | JOIN                         |  |  |  |  |
|  | Technical Support Greek START START  |                              |  |  |  |  |
| Log In 👻   | 714M of 1001M  | © 2020 Silicon Labs          |  |  |  |  |

#### 1.4 Documentation

Relevant OpenThread documentation is available on the OpenThread Developer Documentation site.

Documentation is also available through SSv5. It is filtered based on your selected part. Hardware-specific documentation can be accessed through links on the part OVERVIEW tab.

| EFR32MG12 2.4 GHz 10 dBm RB (ID:440085386) | EFR32MG12 2.4 GHz 10 dBm RB,                              | WSTK Mainboard (ID: 000440085                              | 386)  |  |  |  |  |
|--|---|--|---|--|--|--|--|
|  | OVERVIEW EXAMPLE PROJECTS & DEMOS DOCUME                  | NTATION COMPATIBLE TOOLS                                   |   |  |  |  |  |
|  |   |  | Create New Project                            |  |  |  |  |
| 1  | General Information                                       | Recommended Quick Start Guides                             |   |  |  |  |  |
|  | Connected Via: • J-Link Silicon Labs * Configure          | QSG170: Silicon Labs OpenThread Quick Start Guide          |   |  |  |  |  |
|  | Debug Mode: Onboard Device (MCU) Change                   | QSG169: Bluetooth® SDK v3.x Quick Start Guide              |   |  |  |  |  |
|  | Adapter FW: 1v3p3b928<br>Update to 1.4.4.1099   Changelog | QSG168: Proprietary Flex SDK v3.x Quick Start (            | Suide   |  |  |  |  |
|  | Preferred SDK:  |  |   |  |  |  |  |
| My Products 😂 🕈 🗶 📄 🕀 🗖                    | Gecko SDK Suite v3.1.0 Manage SDKs 👻                      |  |   |  |  |  |  |
| ter product name                           |   |  |   |  |  |  |  |
| My Products 1                              | Board   | Board  | Target Part                                   |  |  |  |  |
|  |   |  | E.  |  |  |  |  |
|  | Wireless Starter Kit Mainboard (BRD4001A Rev<br>A01)      | EFR32MG12 2.4 GHz 10 dBm Radio Board<br>(BRD4162A Rev A01) | EFR32MG12P332F1024GL125                       |  |  |  |  |
|  | View Documents -  | View Documents -   | View Documents -                              |  |  |  |  |
|  | <u></u>   | · · · · · · · · · · · · · · · · · · ·                      | , <b>, , , , , , , , , , , , , , , , , , </b> |  |  |  |  |

SDK documentation and other references are available through the DOCUMENTATION tab. Filter with the Thread Technology Type checkbox to see documentation most closely related to the OpenThread SDK.

| 🕈 Welcome 💿 Recent 🏢 Tools 🛃 Install 🌣 Preferences  |   | 😰 💋 Launcher 🚺 Simplicity   |
|---|---|---|
| Bebug Adapters      Construction      Const | EFR32MG12 2.4 GHz 1<br>OVERVIEW EXAMPLE PROJECTS & DE<br>Read documentation written for your device   |   |
|   |   | 28 resources found  |
|   | Filter on keywords            • Resource Type         • Clear Filter         • Application Notes (13)         •   | QSG170: Silicon Labs OpenThread Quick Start Guide<br>Provides basic information on configuring, building, and installing applications using the Silicon Labs OpenThread<br>stack.   |
|   | Data Sheets (0)  Errata (0)  Quick Start Guides (1)   | Silicon Labs OpenThread API Reference Guide<br>Contains a comprehensive list of APIs used to interface to the Silicon Labs OpenThread stack.  |
| <ul> <li>My Products</li> <li>▲ ¥ € □ □ □</li> <li>Enter product name</li> <li>→ My Products 1</li> <li>→ EFR32MG12 2.4 GHz 19 dBm Radio Board (BRD4161A Rev AC</li> <li>□ EFR32MG12P432F1024GL125</li> </ul>   | Reference Manuals (0)     Release Notes (2)     Schematic and Layout Files (0)     User's Guides (4)     White Papers (0)     Other (8)     Technology Type Clear Filter     32 Bit Mcu SDK (2) | Silicon Labs OpenThread Release Notes<br>Lists compatibility requirements and sources for all software components in the development environment.<br>Discusses the latest changes to the Silicon Labs OpenThread stack (and associated utilities) including<br>added/deleted/deprecated features/API, and lists bugs that have been fixed since the last release and any pending<br>ones. |
|   |   | Gecko Platform Release Notes<br>A detailed overview of all the changes, additions, and fixes in the Gecko Platform components. The Gecko Platform<br>includes: EMLIB, EMDRV, RAIL Library, NVM3, and the component-based infrastructure.  |
|   | Bluetooth (13) Bluetooth Mesh (0) Bootloader (1)  | Documentation Index<br>Provides an overview and hyperlinks to all packaged documentation.   |

SSv5 and its tools are documented in the online Simplicity Studio 5 User's Guide.

| pro                                  |  | environment designed to suppor   | t the Silicon Labs IoT portf                       | olio of system-on-chips (          | SoOs) an              |
|--------------------------------------|--|--|--|------------------------------------|-----------------------|
|                                      | vides access to target device-specific w   | eb and SDK resources; software   | e and hardware configuration                       | on tools; an integrated de         | evelopmer             |
| co                                   | rironment (IDE) featuring industry-stand<br>related energy profiling.  | ard code editors, compilers and  | debuggers; and advanced,                           | value-add tools for hetw           | ork analy:            |
| Overview                             |  |  |  |                                    |                       |
| etting Started                       |  |  |  |                                    | - 🗆 X                 |
| Overview >                           | net «Snellstaging,38 - Simplicity Studie ™<br>Elle Balt Maripate Seguch Dojent Ban (Eindow (Enle<br>▲ Welczone (E) Recent (王) Tools & Install (\$ Preferences  |  |  |                                    | 10 x Lander           |
| Prerequisites >                      | Bi Debug Adapters  | EFR32MG12 2400/915 MHz 19 dE   | m RB, WSTK Mainboard (ID: 000                      | 440085390)                         |                       |
| Install SSv5 and Software            |  | OVERVIEW EXAMPLE PROJECTS DOCUMENTATION  | DEMOS COMPATIBLE TOOLS                             |                                    |                       |
| Explore SSv5                         |  |  |  |                                    | New Project           |
| Start a Project >                    |  | General Information  | Board  | Board                              |                       |
|                                      |  | Debug Node: Onboard Device (MCU)   |  | <u>6.</u>                          |                       |
| out the Launcher                     |  | Adapter Fill: 1x4p4b1099 Lafeet  | Wreless Starter Kit Mainboard (BRD4001A Rev        | EFR32MG12 2400/915 MHz 19 dBm Dual |                       |
| Overview                             |  | Preferred SDK:   | Wretess Starter Kit Manboard (BRD4001A Rev<br>A01) | Radio Board (BRD4164A Rev A02)     | Band                  |
| Project                              |  | Bluetooth Mesh v1.6.2 Manage SDKs 👻  | View Documents •                                   | View Documents                     | •                     |
| Debug Adapters                       | Chy Products Character Cha | Target Part  |  |                                    |                       |
| My Products                          | My Products 1  |  |  |                                    |                       |
| Menu                                 |  | Care and the second sec |  |                                    |                       |
| Toolbar >                            |  | EFR02M012P403F10240L125  |  |                                    |                       |
| out the Simplicity IDE               |  | View Documents   | •  |                                    |                       |
| Overview                             |  |  |  |                                    |                       |
|                                      |  |  |  |                                    |                       |
| veloping for 32-Bit Devices          |  |  |  |                                    | : @ 2020 Silvers Labs |
| Overview                             | (jeven325@eathink.net *  |  |  |                                    | C 2027 Sector Cars    |
| Developing with Project Configurator |  |  |  |                                    |                       |
|                                      | nplicity Studio is designed to simplify de   |  |  |                                    |                       |
| Building and Flashing                | os and, based on the selected developme  | ent target, presents appropriate   | software development kits                          | (SDKs) and other develo            | pment re              |
| Testing and Debugging Sin            | nplicity Studio 5 (SSv5) focuses on deve   | loner experience, leveraging fee   | dhack from customers, err                          | nolovees and competitive           | reviews               |

The SSv5 User Guide pages are organized into the following groups. The first three are generally applicable. They are followed by groups specific to different target devices. Finally, a reference to the tools included with SSv5 is provided.

Getting Started describes how to install SSv5 and the software components relevant to the target device, and provides general overviews to the SSv5

## 2 About Example Applications and Demos

Because starting application development from scratch is difficult, the Silicon Labs OpenThread SDK comes with a number of built-in example applications and demos covering the most frequent use cases and are preconfigured code designed to illustrate common application functions. Silicon Labs strongly recommends starting development from one of the example applications.

Like everything in SSv5, the examples and the demos shown on the EXAMPLE PROJECTS & DEMOS tab are filtered based on the part you have connected or selected.

#### 2.1 Demos

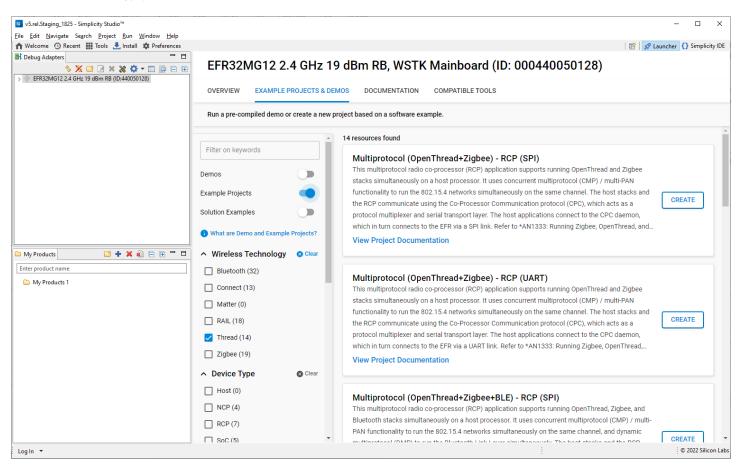
Demos are prebuilt firmware images that are ready to download to a compatible device. The quickest way to find if a demo is available for your part is by adding the part or board information in the My Products view and then navigating to the EXAMPLE PROJECTS & DEMOS tab in the Launcher Perspective. Turn off the Example Projects filter. The Solution Examples filter is provided for future use.

| S v5.rel.Staging_1825 - Simplicity Studio™   |  | - 🗆 X   |  |  |  |  |  |
|--|--|---|--|--|--|--|--|
| <u>File Edit Navigate Search Project Run Window H</u> elp<br>Tools Linstall Content III Tools Linstall   |  | P Jauncher () Simplicity ID   |  |  |  |  |  |
| Debug Adapters         Image: Control of the cont |  | 19 dBm RB, WSTK Mainboard (ID: 000440050128)  |  |  |  |  |  |
|  | OVERVIEW EXAMPLE PROJECTS &            | OVERVIEW EXAMPLE PROJECTS & DEMOS DOCUMENTATION COMPATIBLE TOOLS  |  |  |  |  |  |
|  | Run a pre-compiled demo or create a ne | w project based on a software example.  |  |  |  |  |  |
|  |  | Demo  |  |  |  |  |  |
|  | Filter on keywords                     | OpenThread - NCP (FTD) This is a simple OpenThread Full Thread Device NCP application. This is equivalent to the ot-ncp-ftd |  |  |  |  |  |
|  | Demos 🛑                                | application in the OpenThread GitHub repo.  |  |  |  |  |  |
|  | Example Projects                       | View Project Documentation  |  |  |  |  |  |
|  | Solution Examples                      | Demo  |  |  |  |  |  |
|  | What are Demo and Example Projects?    | OpenThread - NCP (MTD) This is a simple OpenThread Minimal Thread Device NCP application. This is equivalent to the ot-ncp- |  |  |  |  |  |
| 🗀 My Products 🔛 🕂 🗱 🗐 🕀 🖶  | A Wireless Technology S Clear          | mtd application in the OpenThread GitHub repo.  |  |  |  |  |  |
| Enter product name   | Bluetooth (3)                          | View Project Documentation  |  |  |  |  |  |
| My Products 1  | Connect (0)                            |   |  |  |  |  |  |
|  |  | Demo  |  |  |  |  |  |
|  | Matter (15)                            | OpenThread - RCP  |  |  |  |  |  |
|  | RAIL (1)                               | This is a simple OpenThread RCP application. This is equivalent to the ot-rcp application in the                            |  |  |  |  |  |
|  | Thread (12)                            | Open i nread GitHub repo.   |  |  |  |  |  |
|  | Zigbee (6)                             | View Project Documentation  |  |  |  |  |  |
|  | ∧ Device Type Second Clear             | Demo  |  |  |  |  |  |
|  | Host (0)                               | OpenThread - SoC CLI (FTD)  |  |  |  |  |  |
|  | NCP (2)                                | This is a very simple CLI application to test the OpenThread stack on a Full Thread Device. This is                         |  |  |  |  |  |
|  | RCP (5)                                | equivalent to the ot-cil-rtd application in the Open I nread GitHub repo.   |  |  |  |  |  |
|  | SoC (5)                                | View Project Documentation  |  |  |  |  |  |
| Log In 🔻   |  | © 2022 Silicon  |  |  |  |  |  |

A set of prebuilt demo applications are provided with the OpenThread SDK.

#### 2.2 Software Examples

Since typically you will finish by flashing a compiled application image to a device, connect a device to your computer and select it in the Debug Adapters view. In the EXAMPLE PROJECTS & DEMOS tab on the Launcher perspective, filter the example projects based on the 'Thread' technology type and turn off **Demos**.



The example applications provided with the Silicon Labs OpenThread SDK are as follows. Most of the examples come in two variants, Full Thread Device (FTD) and Minimal Thread Device (MTD), and are geared towards System-on-Chip (SoC) and Radio Co-Processor (RCP) application models.

In general, the example applications are simple examples illustrating the following functionality:

- Testing with a CLI
- Radio Co-Processor (RCP)
- Sleepy device
- Dynamic multiprotocol
- Multi-PAN RCP
- Multi-PAN RCP and CPC over SPI/UART

**OpenThread – SoC CLI (FTD):** A CLI application equivalent to the ot-cli-ftd application in the OpenThread GitHub repo to test the OpenThread stack on a Full Thread Device (FTD) for SoC designs. This application can be used to test the OpenThread stack against the Thread Test Harness for interoperability testing.

**OpenThread – SoC CLI (MTD):** A CLI application equivalent to the ot-cli-mtd application in the OpenThread GitHub repo to test the OpenThread stack on a Minimal Thread Device (MTD) for SoC designs.

**OpenThread – RCP:** An OpenThread Radio Co-Processor application equivalent to the ot-rcp application in the OpenThread GitHub repo. While the application layer and the OpenThread core is on the host processor, this application only runs the minimal OpenThread controller on the 802.15.4 SoC. This RCP software example must be used with a compatible OpenThread Border Router. This can be built from the same Silicon Labs OpenThread SDK or deployed from a docker image. For more details, refer to *AN1256: Using the Silicon Labs RCP with the OpenThread Border Router*.

**OpenThread – SoC Sleepy Demo (FTD):** An application to start and form a Thread network on a Full Thread Device (FTD) for the sleepy-demo. This application is used in conjunction with the **OpenThread – SoC Sleepy Demo (MTD)** app.

**OpenThread – SoC Sleepy Demo (MTD):** An application to demonstrate Sleepy End Device (SED) behavior on a Minimal Thread Device (MTD) that attaches to a Thread network started by a node running the **OpenThread – SoC Sleepy Demo (FTD)** app. This application demonstrates power manager feature support and deep sleep (EM2) mode for EFR32.

**OpenThread – SoC Synchronized Sleepy Demo (SSED):** An application to demonstrate Synchronized Sleepy End Device (SSED) behavior using Co-ordinated Sampled Listening (CSL) that attaches to a Thread network started by a node running the **OpenThread – SoC Sleepy Demo (FTD)** app. This application demonstrates power manager feature support and deep sleep (EM2) mode for EFR32.

**OpenThread BLE DMP – SoC Free RTOS:** An application to test Dynamic Multiprotocol (DMP) with OpenThread and Bluetooth running on FreeRTOS. For more details about this application, refer to *AN1265: Dynamic Multiprotocol Development with Bluetooth and OpenThread in GSDK v3.x.* 

**OpenThread (MultiPan) – RCP (UART):** A multipan 802.15.4 RCP (Radio Co-Processor) application. In this application, access to 802.15.4 is provided using SPINEL carried over the CPC (Co-Processor Communication) protocol using a UART connection.

**OpenThread (MultiPan) – RCP (SPI):** A multipan 802.15.4 RCP (Radio Co-Processor) application. In this application, access to 802.15.4 is provided using SPINEL carried over the CPC (Co-Processor Communication) protocol using a SPI connection.

**OpenThread (MultiPan/BLE) – RCP (UART)**: A multiprotocol and multipan RCP (Radio Co-Processor) application. This application runs multipan 802.15.4 and the Bluetooth Link Layer using DMP (Dynamic MultiProtocol). Access to 802.15.4 is provided using the SPINEL protocol, and access to Bluetooth is provided using the standard Bluetooth HCI (Host Controller Interface) protocol. Both are carried over the CPC (Co-Processor Communication) protocol using a UART connection.

**OpenThread (MultiPan/BLE) – RCP (SPI)**: A multiprotocol and multipan RCP (Radio Co-Processor) application. This application runs multipan 802.15.4 and the Bluetooth Link Layer using DMP (Dynamic MultiProtocol). Access to 802.15.4 is provided using the SPINEL protocol, and access to Bluetooth is provided using the standard Bluetooth HCI (Host Controller Interface) protocol. Both are carried over the CPC (Co-Processor Communication) protocol using a SPI connection.

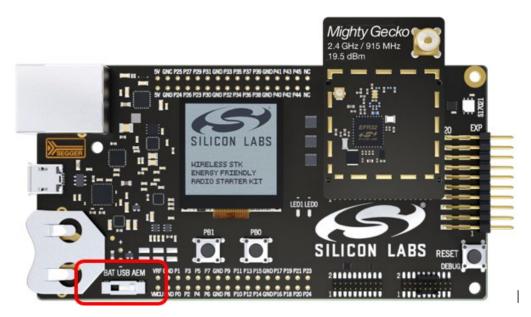
For more details about multi-PAN RCP support, refer to AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP.

## 3 Getting Started with Development

This section assumes that you have downloaded SSv5 and the Silicon Labs OpenThread SDK, and are familiar with the features of the SSv5 Launcher perspective.

The mainboard should be connected.

**Note:** For best performance in Simplicity Studio 5, be sure that the power switch on your mainboard is in the Advanced Energy Monitoring or "AEM" position, as shown in the following figure.



In these instructions you will compile and load a simple **ot-cli-ftd** application on two nodes. Section 3.5 Creating a Network describes how to use the examples to create a network. Section 5.4 Network Analyzer describes how to use Network Analyzer to observe traffic across the network.

When working with an example application in Simplicity Studio, you will be executing the steps in the following order:

- 1. Create a project based on an example.
- 2. Configure the project.
- 3. Build the application image and flash it to your device.

These steps are described in detail in the following sections. These procedures are illustrated for a mainboard with an EFR32MG12.

Note: Your SDK version may be later than the version shown in the figures.

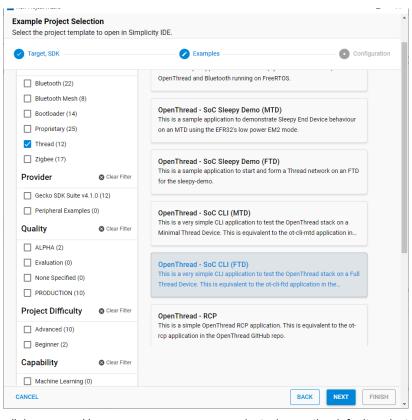
#### 3.1 Creating a Project Based on an Example

Simplicity Studio 5 (SSv5) offers a variety of ways to begin a project using an example application. The online *Simplicity Studio 5 User's Guide*, available both through <u>https://docs.silabs.com/</u> and the SSv5 help menu, describes them all. This guide uses the **File > New > Silicon Labs Project Wizard** method, because it takes you through all three of the Project Creation Dialogs.

1. Open SSv5's File menu and select New > Silicon Labs Project Wizard. The Target, SDK, and Toolchain Selection dialog opens. Do not change the default Simplicity IDE / GNU toolchain supported by OpenThread.

|  | New Project Wizard             |                     |                 |
|--|--------------------------------|---------------------|-----------------|
| Target, SDK, and Toolchain                               | Selection                      |                     |                 |
| Select the target board, device                          | , SDK, and IDE/toolchain to    | o use for the proje | ct.             |
| Target, SDK  | Examples —                     | (                   | • Configuration |
| Target Boards:   |                                |                     |                 |
| Search or Select   |                                |                     | •               |
| Wireless Starter Kit Mainboard                           | (BRD4001A Rev A01) 🛞           |                     |                 |
| EFR32MG12 2.4 GHz 19 dBm R                               | adio Board (BRD4161A Rev A     | 00) 🐼               |                 |
| Target Device:<br>Search or Select                       |                                |                     | •               |
| EFR32MG12P432F1024GL125                                  |                                |                     |                 |
| SDK:   |                                |                     |                 |
| Select SDK<br>Gecko SDK Suite: Bluetooth 3.0             | .0, Flex 3.0.0.0, OpenThread 1 | .0.0.0 (Commit-f411 | a412bee),   🔻   |
|  |                                | •                   | Manage SDKs     |
| IDE / Toolchain:   |                                |                     |                 |
| Select IDE / Toolchain<br>Simplicity IDE / GNU ARM v7.2. | 1                              |                     | -               |
| CANCEL   |                                | BACK                | FINISH          |

2. The Example Project Selection dialog opens. Use the 'Thread' Technology Type and Keyword filters to search for a specific example, in this case **OpenThread – SoC CLI (FTD)**. Select it and click **NEXT**.



3. The Project Configuration dialog opens. Here you can rename your project, change the default project file location, and determine if you will link to or copy project files. Note that if you change any linked resource, it is changed for any other project that references it. Click **FINISH**.

| Target,    | SDK Examples  | — 🕗 Configuratio |
|------------|---|------------------|
| Project na | ne: ot-cli-ftd  |                  |
| 🗸 Use      | default location  |                  |
| Location:  | C:\Users\CAOWENS\SimplicityStudio\v5.master.Staging_2269\ot-cli-ftd | BROWSE           |
|            |   |                  |
|            |   |                  |

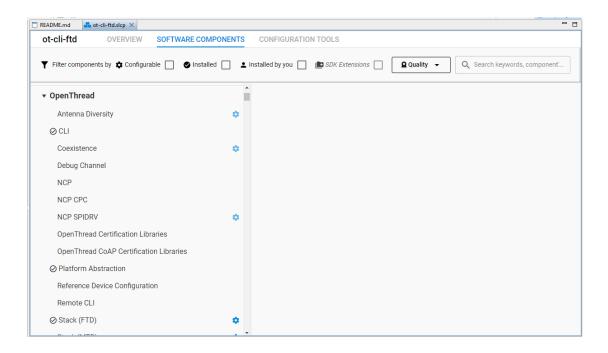
4. The Simplicity IDE Perspective opens with a description of the project on a readme tab. Click the <project>.slcp tab to open the Project Configurator's OVERVIEW tab. See the online *Simplicity Studio 5 User's Guide* for details about the functionality available through the Simplicity IDE perspective and the Project Configurator.

| volinasteriotaging_2209 - otherinta/otherintalsicp - orinpred  | ty studio   |           |                     |                                       |  |          |   |               | <b>_</b>        | ~        |
|--|---|-----------|---------------------|---------------------------------------|--|----------|---|---------------|-----------------|----------|
| <u>F</u> ile <u>E</u> dit <u>N</u> avigate Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> e |   |           |                     |                                       |  |          |   |               |                 |          |
| 🎋 ▼ 😪 ▼   📩 ▼ 🔜 🐚   🗞 ▼   🗞 ▼   🖢 ▼ 🕅 ▼  | 🎨 🕫 🗢 👻 📄 📷 🔡 🏠 Welcome 🚯 Rece  | nt Tools  | 📩 Install 🔅 Prefere | nces                                  |  |          |   | 🔡 🛛 🚼 Simp    | licity IDE 🚀 Li | auncher  |
| 🍋 Project Explorer 🛛 📄 😫 🍟 🗐 🗎   | 📥 ot-cli-ftd.slcp 🗙 🗖 README.md   |           |                     |                                       |  |          |   |               |                 | - 8      |
| v 📂 ot-cli-ftd [GNU ARM v10.3.1 - Default] [EFR32MG12P432  | ot-cli-ftd OVERVIEW SOL   | TWADE CO  | MPONENTS            |                                       | TION TOOLS                             |          |   |               |                 |          |
| > 🔊 Includes   | OU-CII-ITU OVERVIEW 301   | I WARE CO | INFONENTS           | CONFIGURA                             |  |          |   |               |                 |          |
| > 🗁 autogen<br>> 🗁 config  |   |           |                     |                                       |  |          |   |               |                 | <u>^</u> |
| > > gecko_sdk_4.1.0  |   |           | 1 C C               |                                       |  |          |   |               |                 |          |
| > 💽 app.c  | Target and Tool Settings  |           | Project [           | Details                               |  |          | Quick Linl                              | KS            |                 |          |
| > h app.h  |   |           |                     |                                       |  |          |   |               |                 |          |
| > 💽 main.c   |   |           |                     |                                       |  |          |   |               |                 |          |
| a ot-cli-ftd.slcp  |   |           | ot-cli-fto          | 1                                     |  |          |   |               |                 |          |
| ot-cli-ftd.slps  | Mighty Gecko  |           | This is a ve        | ry simple CLI ap                      | plication to test th                   | ne       |   | app.c         |                 |          |
| README.md  | 2.4 GPZ<br>19 dBm   |           |                     |                                       | Thread Device. Th                      | his is   |   |               |                 |          |
|  |   |           |                     | to the ot-cli-ftd a<br>d GitHub repo. | pplication in the                      |          |   |               |                 |          |
|  |   |           | opennice            | a onnao repo.                         |  |          |   |               |                 |          |
|  | I™  |           |                     |                                       |  |          | <                                       | main.c        |                 |          |
|  | <   | >         | Categor             |                                       |  |          | _                                       |               |                 |          |
|  |   |           | OpenThrea           | d Examples                            |  |          |   |               |                 |          |
|  | :   |           |                     |                                       |  |          |   |               |                 |          |
| < >>   |   |           | Preferre            | d SDK                                 |  |          | <>                                      | README.md     |                 |          |
| 📽 Debug Adapters 🗙 📴 Outline 📃 🗖   | TOUCH SLIDER  |           | Gecko SDk           | Suite: Amazon,                        | Bluetooth 3.4.0,                       |          | _                                       |               |                 |          |
|  |   |           |                     |                                       | erZNet 7.1.0.0, Fle                    |          |   |               |                 |          |
| > 🔅 EFR32MG12 2.4 GHz 19 dBm RB (ID:440050128)   |   |           |                     |                                       | CU 6.3.0.0, Micriur<br>(GitHub-9a2d84a |          |   |               |                 |          |
|  |   |           |                     |                                       | 0, Wi-SUN 1.3.0.0,                     |          | ·                                       | Pin Tool      |                 |          |
|  | EFR32MG12P432F1024GL125   |           | Wave SDK            | 7.18.0.0                              |  |          |   |               |                 |          |
|  |   |           |                     |                                       |  |          |   |               |                 |          |
|  | Wireless Starter Kit Mainboard (BRD4001A Rev<br>EFR32MG12 2.4 GHz 19 dBm Radio Board (BRD |           | Import M            | /lode                                 |  |          | dia | Memory Editor |                 |          |
|  | A02)  | 410TA Rev |                     |                                       |  |          | - e:                                    | Memory Editor |                 |          |
|  |   |           | Link sd             | k and copy pro                        | oject sources                          | <b>*</b> |   |               |                 |          |
|  | Selected SDK  |           |                     |                                       |  |          |   |               |                 |          |
|  |   | _         |                     |                                       |  |          |   |               |                 | •        |
|  | 🖹 Problems 🗙 🔗 Search 🏼 🦆 Call Hierarchy  | Console   |                     |                                       |  |          |   |               | 78              | - 0      |
|  | 0 items   |           | <b>D</b> -1         | 1                                     | -                                      |          |   |               |                 |          |
|  | Description   | Resource  | Path                | Location                              | Туре                                   |          |   |               |                 |          |
|  |   |           |                     |                                       |  |          |   |               |                 |          |
|  |   |           |                     |                                       |  |          |   |               |                 |          |
|  |   |           |                     |                                       |  |          |   |               |                 |          |
|  |   |           |                     |                                       |  |          |   |               |                 | 5        |
|  |   |           |                     |                                       |  |          |   |               |                 | · ·      |

#### 3.2 Configuring the Project

Silicon Labs OpenThread applications are built on a Gecko Platform component structure. Click the SOFTWARE COMPONENTS tab to see a complete list of Component categories.

**Note:** All EFR32 parts have a unique RSSI offset. In addition, board, antenna and enclosure design can also impact RSSI. When creating a new project, install the **RAIL Utility, RSSI** component. This feature includes the default RSSI Offset Silicon Labs has measured for each part. This offset can be modified if necessary after RF testing of your complete product.



The project is configured by installing and uninstalling components, and configuring installed components. Installed components are checked. Click **Installed Components** to see a filtered list of components installed by the example application.

Configurable components have a gear symbol. Select a component to see information about it.

| README.md 🚨 ot-cli-ftd.slcp 🗙                  |              |   |
|--|--------------|---|
| ot-cli-ftd OVERVIEW SOFTWARE CC                | OMPONENTS    | CONFIGURATION TOOLS   |
| ▼ Filter components by 🖨 Configurable 🗌 🔮 Inst | alled 🗌 💄 In | stalled by you 📄 💼 SDK Extensions 📄 🔽 Quality 👻 🔍 Search keywords, component' |
| <ul> <li>OpenThread</li> </ul>                 |              |   |
| Antenna Diversity                              | •            | Stack (FTD) Configure   |
| ⊘ CLI  |              |   |
| Coexistence                                    | •            | Description   |
| Debug Channel                                  |              | This component provides the OpenThread stack for a Full Thread Device (FTD)   |
| NCP  |              | Quality   |
| NCP CPC  |              | PRODUCTION  |
| NCP SPIDRV                                     | •            |   |
| OpenThread Certification Libraries             |              | Dependencies ~  |
| OpenThread CoAP Certification Libraries        |              | ot_stack_ftd requires 1 components  |
| Platform Abstraction                           |              | > OpenThread  |
| Reference Device Configuration                 |              | Dependents  |
| Remote CLI                                     |              | 1 components require ot_stack_ftd   |
| ⊘ Stack (FTD)                                  | 0            | ▶ OpenThread  |
| Stack (MTD)                                    | 0            | × Uninstall   |

If the component is configurable, click CONFIGURE to open the Component Editor.

For example, in the Stack (FTD) component you can enable or disable various stack functions. These are equivalent to build options that can be specified to the makefile when building the sample applications in GitHub.

| README.md | 🚢 ot-cli-ftd.slcp 🔯 Stack (FTD) 🗡   |   |   | - 0 |
|-----------|---|---|---|-----|
| Stack (F  | TD)   |   | Pin Tool  | ×   |
| » Det     | fault OpenThread Stack Configuratior  | 1   |   |     |
|           | Thread Stack Protocol Version<br>OPENTHREAD_CONFIG_THREAD_VERSION<br>Thread 1.3 |   |   |     |
|           | The following features require at least Three                                   | ead Stack Protocol Version 1.2                          |   |     |
|           | Backbone Router   | CSL Auto Synchronization using data polling             | CSL (Coordinated Sampled Listening) Debug               |     |
|           |   | •   |   |     |
|           | CSL (Coordinated Sampled Listening) Receiver                                    | DUA (Domain Unicast Address)                            | Link Metrics Initiator                                  |     |
|           | •   | •   | •   |     |
|           | Link Metrics Subject  | Multicast Listener Registration                         | DNS Client (Thread 1.3)                                 |     |
|           | •   | •   | •   |     |
|           | DNS-SD Server (Thread 1.3)  | Service Registration Protocol (SRP) Client (Thread 1.3) | Service Registration Protocol (SRP) Server (Thread 1.3) |     |
|           | <u></u>   |   |   |     |

Any changes made are autosaved, and project files are autogenerated. Progress is shown in the lower right corner of the Simplicity IDE perspective.



#### 3.3 Building the Project

You can either compile and flash the application automatically, or manually compile it and then flash it.

#### 3.3.1 Automatically Compile and Flash

1. You can automatically compile and flash the application to your connected development hardware in the Simplicity IDE. After you click **OK** in the Generation Confirmation dialog, the Simplicity IDE returns. Click the **Debug** control.

|   |                                  |                       | v5.rel           | .Staging_49 - |
|---|----------------------------------|-----------------------|------------------|---------------|
| 🐞 🚱 • i 🖻 • 🔛 🕼                             | 🛞 🔹 🚯 🖬 🖍 Welcome 🕑 R            | ecent 🗰 Tools 📩 Insta | II 🗱 Preferences |               |
| Project Explorer 🔀                          | 🖻 🕏 🍸 🕴 🗖 🖬                      |                       | Stack (FTD)      |               |
| ▼ Sot-cli-ftd [GNU ARI] ► Solution Includes | M v7.2.1 - Debug] [EFR32MG12P432 | OVERVIEW              | SOFTWARE COM     | PONENTS       |

2. Progress is displayed in the console and a progress bar in the lower right.

| Problems        | 🔗 Search        | Call Hierarchy    | 📮 Console 🔀           |   |
|-----------------|-----------------|-------------------|-----------------------|---|
| CDT Build Conso | le [ot-cli-ftd] |                   |                       |   |
|                 |                 |                   |                       |   |
| Building file   | e: /Applicat    | ions/Simplicitv S | tudio.app/Contents/Ec | lipse/developer/sdks/gecko sdk suite/v3.0/util/third party/openthread/src/cor |
| Invoking: GNU   |                 |                   |                       |   |
| arm-none-eab:   | i-g++ -g3 -g    | dwarf-2 -mcpu=cor | tex-m4 -mthumb -std=d | ++0x '-DRTT_USE_ASM=0' '-D0PENTHREAD_CORE_CONFIG_PLATFORM_CHECK_FILE="openthr |
| Finished bui    | lding: /Appl    | ications/Simplici | ty Studio.app/Content | s/Eclipse/developer/sdks/gecko_sdk_suite/v3.0/util/third_party/openthread/src |
|                 |                 |                   |                       |   |
| Building file   | e: /Applicat    | ions/Simplicity S | tudio.app/Contents/Ec | lipse/developer/sdks/gecko_sdk_suite/v3.0/util/third_party/openthread/src/cor |
| Invoking: GNU   |                 |                   |                       |   |
|                 |                 |                   |                       | ++0x '-DRTT_USE_ASM=0' '-DOPENTHREAD_CORE_CONFIG_PLATFORM_CHECK_FILE="openthr |
| Finished bui    | lding: /Appl    | ications/Simplici | ty Studio.app/Content | s/Eclipse/developer/sdks/gecko_sdk_suite/v3.0/util/third_party/openthread/src |
|                 |                 |                   |                       |   |
|                 |                 |                   | tudio.app/Contents/Ec | lipse/developer/sdks/gecko_sdk_suite/v3.0/util/third_party/openthread/src/cor |
| Invoking: GNU   |                 |                   |                       |   |
| arm-none-eab:   | 1-g++ -g3 -g    | dwart-2 -mcpu=cor | tex-m4 -mthumb -std=0 | ++0x '-DRTT_USE_ASM=0' '-DOPENTHREAD_CORE_CONFIG_PLATFORM_CHECK_FILE="openthr |
|                 |                 |                   |                       |   |
|                 |                 |                   |                       |   |

496M of 1115M 🏢 Launching ot-cli-ftd GN...- Debug: (4%) 👝 📹

3. When building and flashing are complete a Debug perspective is displayed. Click the **Resume** control to start the application running on the device.

| •••                                      |   | v5.rel.Stagi  | ng_49 - ot-cli-fto   | /main.c - Simplicity Studio ™ |     |   |                                     |             |                         |         |
|--|---|---|--|-------------------------------|-----|---|-------------------------------------|-------------|-------------------------|---------|
| 😂 🖉 🕨 🔳 🕺 🕹 • 3. 🕫 . g 🖬 🍓 • 🐐           | • 🥭 🥑 📑 🗤   | Welcome   | Recent III To  | ols 📩 Install 🗱 Preferences   |     |   | 😫 🖉 Lau                             | ncher {}Sir | nplicity IDE            | 🎋 Debug |
| 🎋 Debug 🔯 🦲 Project Explorer 🛛 📄 😓 🔹 🔋 🖓 | ot-cli-ftd_slcp   | Stack (FTD)   | C main.c 23  |                               | - 8 | (x)= V 23 %                               | B STOP R                            | Øg° E       | EF D                    |         |
|  | et-cli-ftd.skcp     77⊖ int main(is     77⇒ int main(is     77⇒ int main(is     77⇒ int main(is     78 {     79 otInst     80 {     81 #if OPENTH#     82 if (set     83 {     84 alt     85 #if OPENTH#     86 #if OPENTH#     86 #endif     91     92 #if OPENTH#     93 size t | Stack (FTD)<br>at argc, char *ar<br>ince *instance;<br>IEAD_EXAMPLES_SIM<br>IEAD_ENABLE_COVER<br>IEAD_ENABLE_COVER<br>IEAD_CONFIG_MULTI<br>otInstanceBuff<br>*etinstanceBuff<br>ett:<br>emory Execution | <pre>@ main.c 23 gv(]) ULATION AGE v); PLE_INSTANCE_E erLength = 0; er = NUL</pre> | NABLE                         |     | (x)= v 23 00<br>Name<br>00+argc<br>⇒ argv | B Store R<br>Type<br>int<br>char ** | Øg° E       | D<br>Loca<br>SR0<br>SR1 | i ion   |

Next to the Resume control are **Suspend**, **Disconnect**, **Reconnect**, and **stepping** controls. Click **Disconnect** when you are ready to exit Debug mode.

|   | v5.rel.Staging_49 - ot-cli-ftd/main            |
|---|--|
| 🗁 🕮 🗐 🔊 🖘 😓 🚱 🔝 💷 📲                           | 🔯 🕈 🥖 🔂 🖬 🐨 🔛 🦙 🏙 🖍 Welcome 🙆 Recent 🏢 Tools 📩 |
| 🎋 Debug 🔀 🎦 Project Explorer 🛛 📄 🍪 🖷 🗖        |  |
| Silicon Labs ARM MCU: EFR32MG12P432F1024GL125 | 70   |
| ▼ 🔐 ot-cli-ftd.axf                            | 77⊖ int main(int argc, char *argv[])           |
| main() at main.c:78 0x39a34                   | ▶ 78 {   |

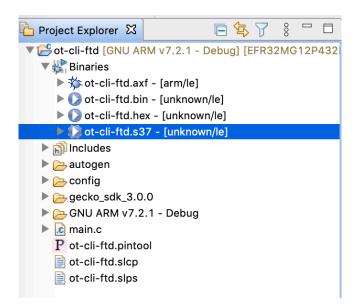
### 3.3.2 Manually Compile and Flash

1. After you generate your project files, instead of clicking Debug in the Simplicity IDE, click the **Build** control (hammer icon) in the top tool bar.

| • • •                   | _ |       |       |                   | v5.rel.Staging   | g_49 - ot-cli-ftd/main.c - |
|-------------------------|---|-------|-------|-------------------|------------------|----------------------------|
| <b>☆・ 😪 • 📑 • 🔡 🕲 🛞</b> | 5 | ·幻·初· | *:    | 🛃 🔝 🏦 Welcome     | 🖸 Recent 🗰 Tools | 📩 Install 🔹 Preferences    |
| Project Explorer 🛛      |   |       | 8 🗆 🗖 | evot-cli-ftd.slcp | Stack (FTD)      | 🔂 main.c 🔀                 |

2. You can load the binary image through Project Explorer view.

Locate the <project>.bin, .hex, or .s37 file in the compiler subdirectory.



Right-click the file and select **Flash Programmer**. Alternately, you may see a **Flash to Device...** option, in which case, you will need to select a device to program, which should take you to the Flash Programmer dialog. The Flash Programmer opens with the file path populated. Click **PROGRAM**.

|   | Flash Programmer   |
|---|--|
| Change Device<br>Device<br>Board Name: EFR32MG12 2.4 GHz 19 dBm Rad<br>Board Name: Wireless Starter Kit Mainboard<br>MCU Name: EFR32MG12P432F1024GL125<br>Adapter<br>Name: No name (10.4.187.163) | lio Board  |
| Flash Part  |  |
| File Type • hex • bin Base address  | V OxO  |
| File  |  |
| /Users/sachinch/SimplicityStudio/v5.rel.Stag  | ing_49/ot-cli-ftd/GNU ARM v7.2.1 - Debug/ot 🎴 Browse                 |
| Advanced Settings   | Erase Program  |
| Flash Erase/Write Protection  |  |
| <ul> <li>Select flash range</li> </ul>  | ▼ 0x0 ▲ → ▼ 0x100000 ▲   |
| Select default sections   | Lock Main Flash     Cock User Page     Protect     Remove Protection |
| Debug Lock Tools  |  |
| The unlock function only works using Silicon L<br>Unlocking the chip will erase all data on flash a   |  |
|   |  |
| ?   | Close  |

#### 3.4 Flashing a Bootloader

All Silicon Labs examples require that a bootloader be installed. By default, a new device is factory-programmed with a dummy bootloader. If you have a new device, haven't cleared the bootloader region for your part or have a supported bootloader image already flashed on your device, skip this step and continue with the next section.

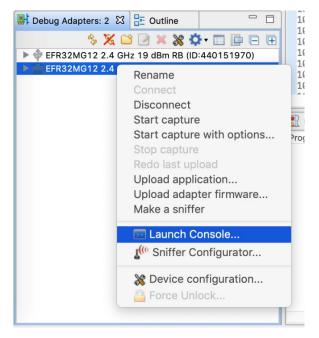
With Silicon Labs OpenThread, the bootloader serves to start the application code within the image you created and flashed in the previous procedure. Once you have installed a bootloader image, it remains installed until you erase the bootloader region for your device.

If you need to flash a bootloader, select a bootloader example and build and flash it as you would any application. See UG266: Silicon Labs Gecko Bootloader User's Guide for GSDK 3.2 and Lower or UG489: Silicon Labs Gecko Bootloader User's Guide for GSDK 4.0 and Higher for more information.

#### 3.5 Creating a Network

Depending on the example application, you may be able to interact with it through your development environment's Console interface using a CLI (command line interpreter). The console interface allows you to form a network and send data.

To launch the Console interface, in the Simplicity IDE perspective right-click on your J-Link device in the Devices View / Debug Adapters Window. Choose **Launch Console**. Alternatively, from the Tools icon in the Simplicity IDE toolbar, menu in the Launcher perspective select Device Console. To get a prompt on the Console, choose Serial 1 tab and press Enter.



To create a two-node network using the ot-cli-ftd example, flash the **ot-cli-ftd.s37** application image on two separate nodes. Launch the console for the two nodes and execute the commands given below:

#### Form a network

On Device A, enter:

```
> dataset init new
Done
> dataset commit active
Done
> ifconfig up
Done
> thread start
Done
```

The previous commands should initialize the node with new network parameters and form a network. Check the status of the node after a few seconds by executing the following command. The node should now be a leader.

> state leader Done

Check the network parameters by running the following command:

```
> dataset
dataset
Active Timestamp: 1
Channel: 18
Channel Mask: 07fff800
Ext PAN ID: 4e08d5129185b233
Mesh Local Prefix: fda6:83fa:7cc1:e809/64
Master Key: d5bf78c4bbd96605c6e55a32fa2c03ab
Network Name: OpenThread-21ec
PAN ID: 0x21ec
PSKc: 3be0de54e5d0f2cd5622de0716680588
Security Policy: 0, onrcb
Done
```

#### Attach a second node (device B) to the network

Note: Depending on the network parameters generated by node A (as seen above), update the channel number and the master key in the following commands.

#### On Device B enter:

```
> dataset channel 18
Done
> dataset networkkey d5bf78c4bbd96605c6e55a32fa2c03ab
Done
> dataset commit active
Done
> ifconfig up
Done
> thread start
Done
```

The above commands should cause the node to attach to the network started by node A. Check the status of the node after a few seconds by executing the following command. The node should now be a child (which will eventually update to a router).

> state child Done

#### Test network connectivity

For this step, compute Node B's IP addresses and ping it from Node A as follows.

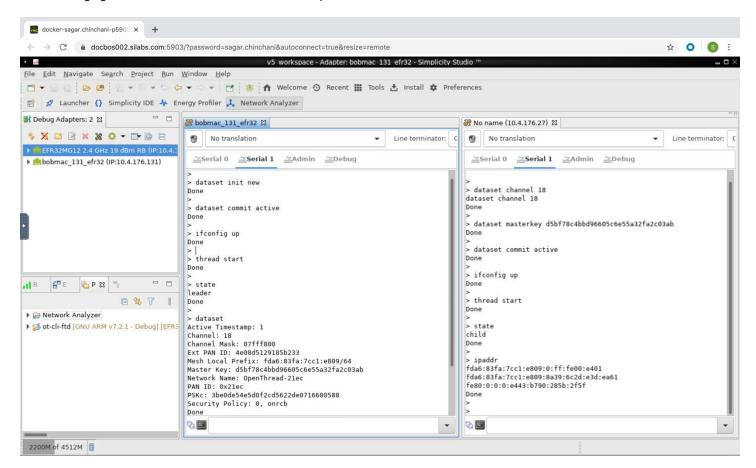
On device B enter:

```
> ipaddr
ipaddr
fda6:83fa:7cc1:e809:0:ff:fe00:e000
fda6:83fa:7cc1:e809:8a39:6c2d:e3d:ea61
fe80:0:0:0:e443:b790:285b:2f5f
Done
```

On node A enter:

```
> ping fe80:0:0:0:e443:b790:285b:2f5f
ping fe80:0:0:0:e443:b790:285b:2f5f
Done
> 
> 16 bytes from fe80:0:0:0:e443:b790:285b:2f5f: icmp_seq=2 hlim=64 time=21ms
```

The following figure shows the console at the end of the procedure.



## 4 Next Steps

Some next steps:

- Run additional cli commands supported by the ot-cli-ftd application (as covered in Chapter 3) to explore OpenThread functionality. A complete list of OpenThread CLI commands can be found here: <a href="https://github.com/openthread/openthread/blob/mas-ter/src/cli/README.md">https://github.com/openthread/openthread/blob/mas-ter/src/cli/README.md</a>
- Compile and flash a different example application and explore the functionality it provides. For example, sleepy end device (SED) behavior is supported by the sleepy-demo-mtd application. Some interesting details about this application have been covered in section 5.3 Multi-Node Energy Profiler.
- Explore configuring one of the example applications to meet your needs. Much of the software configuration can be done through components. Select a component to see more information about it and, check if it is configurable.
- Explore the documentation on the OpenThread developer documentation site.

## 5 Development Tools

## 5.1 Simplicity Commander

Simplicity Commander is a simple flashing tool, which can be used to flash firmware images, erase flash, lock and unlock debug access, and write-protect flash pages via the J-Link interface. Both GUI and CLI (Command Line Interface) are available. See UG162: Simplicity Commander Reference Guide for more information.

| 🤞 Sin glicity Commander   |   |
|---|---|
| <u>F</u> ile <u>H</u> elp   |   |
| Disconnect J-Link Device  | S/N 440048424 C J-Link Speed 12000 kHz  |
| Disconnect     J-Link Device       Kit       Flash       SWO Terminal | S/N 440048424       C J Link Speed 12000 kHz         Flash MCU         Binary File         ipp/bluetooth_framework_dev/appbuilder/sample-apps-bin/ncp-empty-target-brd4300areva03-iar.hex       Browse         Flash start address:       00000000 ⊕ Ø Reset MCU after flashing Ø Verify upload Flash         Flash Erase/Write Protection       Ø0000000 ⊕ > 0000000 ⊕ Protect         Write protect flash range       00000000 ⊕ > 0000000 ⊕ Protect         Debug Lock Tools       Remove Protection         The unlock function only works using Silicon Labs kits.       Unlock debug access         Unlock debug access       Lock debug access |
|   |   |
|   | Silicon Labs J-Link Pro OB compiled Apr 20 2016 12:17:55  |

Figure 5-1. Simplicity Commander

#### 5.2 Pin Tool

Simplicity Studio offers a Pin tool that allows you to easily configure new peripherals or change the properties of existing ones. In the CONFIGURATION TOOLS tab, click Open on the Pin Tool card or double-click the file <project>.pintool in the Project Explorer view.

|   | 🏘 ot-cli-ftd.slcp    | Stack (FTD)               | P ot-cli-ftd.pintool                    |                   |                    |                         |                 |  |
|---|----------------------|---------------------------|---|-------------------|--------------------|-------------------------|-----------------|--|
| 😴 ot-cli-ftd [GNU ARM v7.2.1 - Debug] [EFR32MG12P432 <br>🔻 🎇 Binaries   | DefaultMode P        | ort I/O: PORTIO           |   | Confi             | gure               |                         | Search:         | A 6  |
| ▶ ☆ ot-cli-ftd.axf - [arm/le]<br>▶ 🕐 ot-cli-ftd.bin - [unknown/le]  | •                    |                           |   | Pins F            | unctions Peri      | pherals                 |                 |  |
| <ul> <li>Ot-cli-ftd.bit - [unknown/le]</li> <li>Ot-cli-ftd.s37 - [unknown/le]</li> </ul>  |                      |                           |   | Pin #<br>M13      | Pin Name<br>PA0    | Function<br>USART0_TX   | Custom Pin Name | UARTDRV USART Dri                                  |
| includes     Autogen  | 000                  | 0000                      |   | L13<br>L12        | PA1<br>PA2         | USARTO_RX<br>USARTO_CTS |                 | UARTDRV USART Dri<br>UARTDRV USART Dri             |
| <ul> <li>Config</li> <li>Config</li></ul> | 000                  |                           |   | K13<br>K12<br>J13 | PA3<br>PA4<br>PA5  | USARTO_RTS<br>GPIO mode |                 | UARTDRV USART Dri<br>Board Control : SL_B.         |
| <ul> <li>GNU ARM v7.2.1 - Debug</li> <li>main.c</li> </ul>  | 000                  |                           |   | J12<br>J11        | PA6<br>PA7         | GFIG mode               |                 | Board Control : SL_B.                              |
| P ot-cli-ftd.pintool  |                      |                           |   | H13<br>H12        | PA8<br>PA9         |                         |                 |  |
| i ot-cli-ftd.slps   | 00                   |                           |   | F12<br>E13<br>E12 | PB6<br>PB7         |                         |                 |  |
|   |                      |                           |   | D13               | PB8<br>PB9<br>PB10 | GPIO mode               |                 | Board Control : SL B.                              |
|   | <u> </u>             |                           |   | D11<br>C13        | PB11<br>PB12       | PTI_DCLK<br>PTI_DOUT    |                 | RAIL Utility, Packet Ti<br>RAIL Utility, Packet Ti |
|   |                      |                           |   | C12<br>C11<br>C10 | PB13<br>PB14       | PTI_DFRAME              |                 | RAIL Utility, Packet Ti                            |
| Debug Adapters 🕱 📴 Outline 📃 🗆  |                      |                           | AND | A5<br>B5          | PB15<br>PC0<br>PC1 |                         |                 |  |
| % 💥 🗳 🕢 💥 🏷 - 🔲 📑 🕀   |                      | 125-pin BGA, 7x7 - (top v | (ew)                                    | C5<br>A4          | PC2<br>PC3         |                         |                 |  |
|   | Zoom:                | x) 🕵                      |   | 84<br>A3          | PC4<br>PC5<br>PC6  |                         |                 |  |
|   |                      |                           |   | (California)      | PLB                |                         |                 |  |
|   | Problems 🔗           | Search 🐉 Call Hi          | erarchy 📮 Console 🔀                     |                   |                    |                         |                 | 🖗 🛃 • 🔂 • 🧮  |
|   | Frogram Output Consc | ALC .                     |   |                   |                    |                         |                 |  |

Double-click a software component to open the Component Editor and configure that function. Pin Tool does not autosave.

#### 5.3 Multi-Node Energy Profiler

Multi-Node Energy profiler is an add-on tool, with which you can easily measure the energy consumption of your device in runtime. You can easily find peak and average consumption, and check for sleep mode current.

To profile the current project, drop down the Profile as menu in the Simplicity IDE perspective and select Profile as / Simplicity Energy Profiler target. This automatically builds your project, uploads it to the device, and starts Energy Profiler.

| • •               | •  | v5.rel.Staging_49 - ot-cli-ftd/ot-cli-ftd.slcp - S |
|-------------------|--|--|
| ** 9              | 🖢 र 📑 👘 🖓 र 🗞 र 🛃 र 🖏 र 🏷 🗘 र 🔿 र 📑 👘 Welcome                | e 🔁 Recent 🏢 Tools 📩 Install 🏟 Preferences         |
| Pr                | 🕵 1 ot-cli-ftd GNU ARM v7.2.1 - Debug (Exx32 energy profile) |  |
| <b>▼</b> <u>6</u> | Profile As   | 🕵 1 Simplicity Energy Profiler Target              |
|                   | Profile Configurations                                       | T Simplicity Energy Promet Target                  |
|                   |  |  |

For details on how to launch the energy profiler or switch between Simplicity IDE and Energy Profiler perspectives, refer to UG343: Multi-Node Energy Profiler User's Guide.

| •••  | Energy | Profiler - Simplic | ty Studio ™ |   |              |                     |                            |                   |
|--|--------|--------------------|-------------|---|--------------|---------------------|----------------------------|-------------------|
|  |        |                    |             |   | 🔡 🚀 Launcher | r {} Simplicity IDE | 🎋 Debug 🧘 Network Analyzer | A Energy Profiler |
|  |        | 0.00 s             | 0.00 mW     | e |              | ~                   | sleepy-demo-mtd.slcp       | 2                 |
| Quick Access 🗸 💽 Single-Node 💥 Multi-Node 💥 Scope View |        | 0.00 mA            | 0.00 Wh     |   |              | Ø                   | OVERVIEW SOF               | TWARE COM:20      |

For a node running the ot-cli-ftd application, the Energy Profiler capture should be as shown in the following figure:

|              |               |                     |              |                 |          | Energy Profiler - Sim | plicity Studio ™    |                                    |               |                           |  |
|--------------|---------------|---------------------|--------------|-----------------|----------|-----------------------|---------------------|------------------------------------|---------------|---------------------------|--|
|              |               |                     |              |                 |          |                       |                     | 😰 🚀 Launcher { S                   |               |                           | Analyzer Analyzer Profiler             |
|              |               |                     | 😽 Multi-Node | 10/ 0 X/        | - 🕨      | <b>131.64</b> s       | ى 17.00 mW          |                                    |               | ev ot-cli-ftd.slcp        | 3                                      |
| Quick Ac     | cess ~        | Single-Node         | Multi-Node   | Scope View      |          | Recording 5.14 mA     | 621.53 μWh          |                                    | 0             | OVERVIEW                  | SOFTWARE COM?0                         |
| Display Node | ~             | Search Q            |              |                 |          |                       | Current Voltage Avg | IRQ RX/TX Bookmai                  | rks           |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    | oiliatainitai | 1                         |  |
|              |               |                     |              |                 |          |                       |                     | 131.64 s 17.00 m<br>5.14 mA 621.53 |               | Target a                  | nd SDK                                 |
| ю            |               |                     |              |                 |          |                       |                     | 5.14 mA 621.53                     | μwn 🔳         | S                         |  |
| •            |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
| •            |               |                     |              |                 |          |                       |                     |                                    |               | M<br>EFR                  | lighty Gecko 📷                         |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           | s ygro 🔒                               |
| Θ            |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
| nent         |               |                     |              |                 |          |                       |                     |                                    |               | Ĭ                         |  |
| 5            |               | ~                   |              |                 |          |                       | ~~~ (`~             |                                    |               |                           | · · · · · · · · · · · · · · · · · · ·  |
|              |               |                     |              |                 |          |                       |                     |                                    | *****         |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    |               | EED22M                    | G12P432F1024GL1                        |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
| Ø            |               |                     |              |                 |          |                       |                     |                                    | _             | Wireless Sta<br>(BRD4001A | rter Kit Mainboard<br>Rev A01)         |
|              |               | -36.00 s            | -32.00 s     | -28.00 s -24    | .00 s -2 | 0.00 s -16.00 s       | -12.00 s            | -8.00 s -4.00 s                    | 0.00 s        | EFR32MG12                 | 2.4 GHz 19 dBm Radio<br>4161A Rev A02) |
|              |               |                     |              |                 |          |                       |                     | 4.00 s 🏏 😑 🛛 🔵                     | € ⊕           | board (bitb               |  |
| Energy F     | Profile (live | e) 🛱 📴 Energy Profi | ile (range)  | bug Adapters: 2 |          |                       |                     |                                    |               | -                         |  |
| No name (1)  |               |                     | C Function   | Ener            | gy (     | Contribution (%)      |                     |                                    | Threshold 1~  | CI                        | hange                                  |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    |               | Project                   | Details                                |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
|              |               |                     |              |                 |          |                       |                     |                                    |               |                           |  |
|              |               |                     |              |                 |          |                       |                     | 720M c                             | of 950M 📋 Ca  | pture with adapter        | .(10.4.187.163) 🚥 💼                    |

Figure 5-2. Current consumption capture on a node operating in EM0 (active) mode

Please note that the energy consumption results you observe may differ as the peripherals and their energy mode are highly dependent on the specific device.

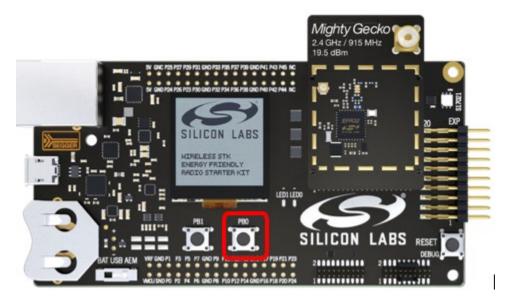
#### 5.3.1 Energy Profiler with Sleepy-demo Applications

While the energy profiler can be used on any running project, a good example is the **OpenThread – SoC Sleepy Demo (MTD)** application available with the Silicon Labs OpenThread SDK. It demonstrates Sleepy End Device (SED) behavior using the EFR32's low power EM2 mode.

For this exercise, you will create a two-node network using the **OpenThread – SoC Sleepy Demo (FTD)** and **OpenThread – SoC Sleepy Demo (MTD)** application and monitor the current consumption on the sleepy node.

- To build the OpenThread SoC Sleepy Demo (FTD) and OpenThread SoC Sleepy Demo (MTD) example applications, either flash the precompiled demo applications as described in section 2.1 Demos or build your own application as described in section 3 Getting Started with Development.
- Since the OpenThread SoC Sleepy Demo (FTD) and OpenThread SoC Sleepy Demo (MTD) applications are already configured with default network parameters, flashing the applications should automatically form a 2-node network (without specifying any additional commands). Accordingly,
  - On Node A, flash the OpenThread SoC Sleepy Demo (FTD) application. This should cause Node A to form a network.
  - On Node B, flash the **OpenThread SoC Sleepy Demo (MTD)** application. This should cause node B to attach as a Minimal Thread Device (MTD) to the network formed by Node A.]
  - Try testing network connectivity by sending a ping from Node A(FTD) to Node B(MTD) as described in section 3.5 Creating a Network.

3. Launch the Energy Profiler, start a capture on Node B(MTD), and press button 0 Node B's mainboard.



Pressing button 0 on the Node B(MTD) mainboard will cause the node to toggle between operating as a Minimal End Device (MED) and a Sleepy End Device (SED) with the RX off when idle. As the node uses EFR32's low power EM2 (deep sleep) mode, current consumption will significantly drop (from mA to µA range) as seen in the capture below.

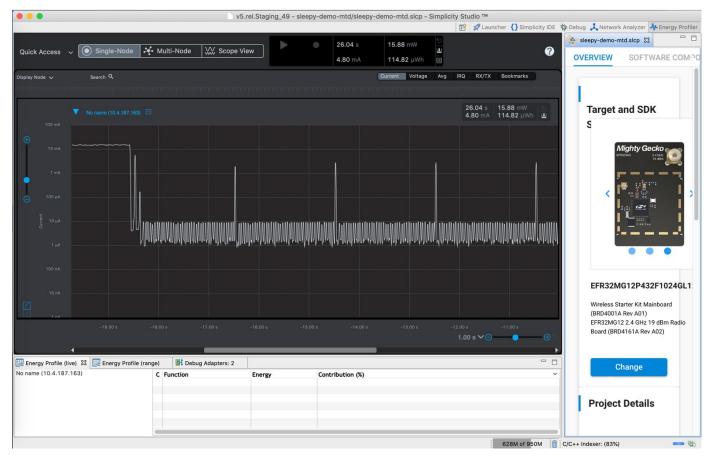


Figure 5-3. Current consumption capture on a node operating in EM2 (deep sleep) mode

The regular spikes seen every 2 seconds (i.e. the sleep period) represents the tiny interval during which the node briefly wakes up to send a data poll message to its parent, after which it goes back to sleep.

Profiling can be paused by clicking Play. For further analysis, click one of the peaks, and zoom in with time axis (x-axis) and the current axis (y-axis). Note that the maximum consumption may now be greater than it appeared on the diagram before you zoomed in. This is because in zoomed-out mode, the displayed values are averaged. If you need exact values, always zoom in. To measure average consumption, simply click and drag your mouse over a time interval. A new window appears in the upper right corner showing consumption information for the given interval.



Figure 5-4 Energy profiler capture on a node operating in EM2 (deep sleep) mode for a given interval

Multi-node Energy Profiler is also able to simultaneously measure the consumption of multiple devices. To start measuring a new device click the Quick Access menu (upper left corner), select **Start Energy Capture** and choose the **Multi-Node View**. To stop measuring, click the Quick Access menu, and select **End/Save session**.

| Quick Acce                                      | ss | Single-Nod           | Multi-Node  | Scope View                              | Running Recording     | 230.53 s<br>14.84 mA | 49.37 mW<br>3.16 mWh | 9<br>E              | an manufacture and a |            |                             |                      | Ø                |
|---|----|----------------------|-------------|---|-----------------------|----------------------|----------------------|---------------------|----------------------|------------|-----------------------------|----------------------|------------------|
| Display Nodes                                   |    |                      |             |   |                       |                      |                      |                     | Current              | Voltage Av | g IRQ RX/TX                 | Bookmarks            | ((0)) ((0))<br>0 |
| ⊖ 100<br>€ 10                                   |    | ▼ J-Link Silcon Labs | (440151970) |   |                       |                      |                      |                     |                      |            | 230.53 s<br>14.78 mA        | 49.18 mW<br>3.15 mWh |                  |
| 0 0 100<br>10 10<br>10 0 0<br>100<br>100<br>100 |    | No name (10.4.187    |             | unan na ana ana ana ana ana ana ana ana | ayaayaayaaawaaya baaa | Mumulika sharing     |                      | M ALAULAN M JAN M A |                      | NAMANANA   |                             | 94.98 µW<br>5.79 µWh |                  |
| 100   |    |                      |             |   |                       |                      |                      |                     |                      |            | -2.00 s<br>1.00 s 🏏 \ominus | -1.00 %              | 2 00.5<br>C ()   |

To learn more about how to use this tool, see UG343: Multi-Node Energy Profiler User's Guide.

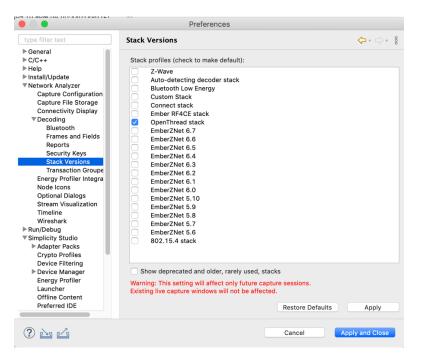
#### 5.4 Network Analyzer

Silicon Labs Network Analyzer is a packet capture and debugging tool that can be used to debug connectivity between wireless nodes running OpenThread stack on EFR32 platform. It significantly accelerates the network and application development process with graphical views of network traffic, activity, and duration. See the online *Simplicity Studio 5 User's Guide*, available both through <a href="https://docs.si-labs.com/">https://docs.si-labs.com/</a> and the SSv5 help menu, for a guide to Network Analyzer.

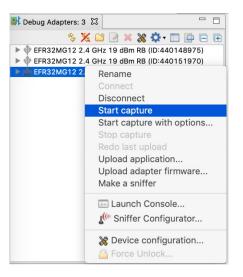
The Packet Trace application captures the packets directly from the Packet Trace Interface (PTI) available on the Wireless Gecko SoCs and modules. It therefore provides a more accurate capture of the packets compared to air-based capture.

To capture OpenThread packets using the Silicon Labs Network Analyzer:

1. In the Preferences window (Simplicity Studio > Preferences), under Network Analyzer > Decoding > Stack Versions, select the OpenThread stack option.



- In the same Preferences window (Simplicity Studio > Preferences), under Network Analyzer > Decoding > Security Keys, verify the correct Network Master Key used by the Thread network has been added to the list of known keys.
- 3. In the Debug Adapters view, right-click the device that is running on a Thread network and start capture.



You should now be able to see the Thread traffic as shown below. Clicking the packets to see more details about its contents in the Event Detail view (on the right).

|   | v5 rel Staging 49 - Live c                                    | apture stream - Simplicity Studio ™           |  |  |  |  |  |  |  |  |  |
|---|---|---|--|--|--|--|--|--|--|--|--|
| C. (#1 P. [] [] [] [] [] [] [] [] [] [] [] [] []  |   |   | 🚏 🐏 🏥 🏫 Welcome 🕑 Recent 🏢 Tools 📩 Install 🍁 Preferences   |  |  |  |  |  |  |  |  |
| A Launcher () Simplicity IDE 🏷 Debug 🙏 Network An   |   | • • • • • • • • • • • • • • • • • • •         | The state of the s |  |  |  |  |  |  |  |  |
| ■ Debug Adapters: 3 🖾 📃 🗆   | sleepy-demo-mtd.slcp  |   |  |  |  |  |  |  |  |  |  |
| % 💥 ピ 🗷 X X 🌣 🗖 🖻 🕀   |   | et" && event summary I= "EED Dy nacket"       |  |  |  |  |  |  |  |  |  |
| ▶ @ EFR32MG12 2.4 GHz 19 dBm RB (ID:440148975)  |   |   |  |  |  |  |  |  |  |  |  |
| FR32MG12 2.4 GHz 19 dBm RB (ID:440151970) EFR32MG12 2.4 GHz 19 dBm RB (IP:10.4.187.163) [ |   |   |  |  |  |  |  |  |  |  |  |
|   | Time:223.110546s Real time:N/A Nodes:1 Event:                 | EFR Tx packet                                 | 💈 Event Detail 🗢 🗖   |  |  |  |  |  |  |  |  |
|   | Transactions total:0 shown:0<br>Time Duratic Summary IPv6 Src | ⊽ 🗖<br>IPv6 Dest P# M# E# Error Statu Warning | MLE crypto: ROOT, 5A C7 28 28 8F 93 C8 9C F2 C4 A0 15 3E AA A0 C0           ▶IEEE 802.15 4.116 bytes]           ▶Lowpan UDP (7 bytes]           ▶Lowpan UDP (7 bytes]           >MLE Security (11 bytes]           Security Control: Cv15           Key Identifier Mode: 4 byte source (2)           Security Level: Encrypted, 4 byte MIC (5)           Frame Counter: 0x00000000           Key Identifier Mode: 4 byte source (2)           Full Sequence: 0x0000000           Key Identifier Adde: 8 byte source (2)           Full Sequence: 0x00000000           Key Identifier Adde: 8 byte Source (2)           Full Sequence: 0x0000000           Key Identifier Adde: 8 byte Source (2)           Full Sequence: 0x0000000           Key Identifier Adde: 8 byte Source (2)           Full Sequence: 0x0000000           Key Identifier Adde: 8 byte Source (2)           Full Sequence: 0x0000000           Key Identifier Adde: 8 byte Source (2)           Command: Network Data Response (0x08)           Me TV_0: Source Address (0x00)           Length: 2  |  |  |  |  |  |  |  |  |
|   | C   |   | Source: 0x4000   |  |  |  |  |  |  |  |  |
|   | Events total:249 shown:12 Decoders: OpenThread                | i stack, OpenThread stack                     | Mle Tlv_1: Leader Data (0x0B)  |  |  |  |  |  |  |  |  |
| , ] R X 🚰 E 🖧 C 🗭 D □ 🗖   | Time Type Summary   | MAC Src MAC Des Event error Event warn        | Length: 8  |  |  |  |  |  |  |  |  |
|   | 216.48 OutOfSeq Out of sequence: 440148975                    | Expected                                      | Partition Id: 0x284CA465<br>Weighting: 0x40  |  |  |  |  |  |  |  |  |
| 8   | 221.10 Packet MLE Parent Request                              | 76973 FFFF                                    | Data Version: 0x54   |  |  |  |  |  |  |  |  |
| Tx count: 1, Rx count: 0  | 221.85 Packet MLE Parent Request                              | 76973 FFFF                                    | Stable Data Version: 0xE7  |  |  |  |  |  |  |  |  |
| v (p) Sender: 440148975 [12E5]  | C 223.11 Packet MLE Network Data Response                     | 76973 FFFF                                    | Leader Router Id: 0x10   |  |  |  |  |  |  |  |  |
| Time: 223.110546  | 223.69 Packet MLE Advertise                                   | 76973 FFFF                                    | Mle Tlv_2: Active Timestamp (0x16)   |  |  |  |  |  |  |  |  |
| Bytes: 3F 7D FD 04 00 00  | 225.50 Packet MLE Advertise                                   | 76973 FFFF                                    | Length: 8  |  |  |  |  |  |  |  |  |
|   | 228.75 Packet MLE Advertise                                   | 76973 FFFF                                    | Active Timestamp: 1 sec, 0 ticks, U=false<br>Mle Tlv_3: Network Data (0x0C)  |  |  |  |  |  |  |  |  |
| Type: EFR Tx packet   | 235.61 Packet MLE Advertise                                   | 76973 FFFF                                    | Mile Tiv_5. Network Data (0x0C)  |  |  |  |  |  |  |  |  |
| Status: 00  | 253.34 Packet MLE Advertise                                   | 76973 FFFF                                    |  |  |  |  |  |  |  |  |  |
|   | 271.54 Packet MLE Advertise                                   | 76973 FFFF                                    | Hex Dump [81 bytes] 🗢 🗖  |  |  |  |  |  |  |  |  |
|   | 305.95 Packet MLE Advertise<br>342.39 Packet MLE Advertise    | 76973 FFFF<br>76973 FFFF                      | FC 48 41 D8 BF C5 CF FF FF 5F 61 A9 E1 16 32 97       .KAa   |  |  |  |  |  |  |  |  |
|   |   |   | MLE decrypted (1 more)   |  |  |  |  |  |  |  |  |

Figure 5-5. Thread Traffic Capture with Packet Trace

#### 5.5 Crash Handler

When debugging crashes it may be useful to have detailed information about core registers, information about the C stack, and reset information. This data can be automatically collected and printed on the next boot-up by including the OT Crash Handler in your project.

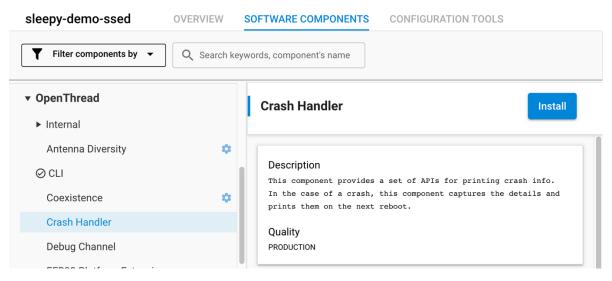


Figure 5-6. Crash Handler Component

The crash data is printed using the OpenThread logging system, meaning the output location of the crash data can be controlled by modifying the value of the OPENTHREAD\_CONFIG\_LOG\_OUTPUT configuration.

#### Example: Crash dump

- [C] Platform-----: Reset info: 0xa (FLT) [C] Platform-----: Extended Reset info: 0xA04 (USG)
- [C] Platform-----: Thread mode using main stack (20000000 to 20001200), SP = 20001040
- [C] Platform-----: 1072 bytes used (23%) in main stack (out of 4608 bytes total)
- [C] Platform-----: No interrupts active
- [C] Platform-----: Reset cause: Usage Fault
- [C] Platform-----: Instruction address: 8009272
- [C] Platform-----: CFSR.UNALIGNED: attempted an unaligned memory access
- [C] Platform-----: R0 = 0x0804f518, R1 = 0x00000000, R2 = 0x00000000, R3 = 0xdeadb000
- [C] Platform-----: R4 = 0x00000000, R5 = 0x200012f8, R6 = 0x080092a9, R7 = 0x00000008
- [C] Platform-----: R8 = 0x20007923, R9 = 0xffffffff, R10 = 0x1fff1200, R11 = 0x0000000
- [C] Platform-----: R12 = 0x20007e2f, R13(LR) = 0xffffff9, MSP = 0x20001040, PSP = 0x00000000
- [C] Platform-----: PC = 0x08009272, xPSR = 0x69000000, MSP used = 0x00000430, PSP used = 0x00000000
- [C] Platform-----: CSTACK bottom = 0x20000000, ICSR = 0x00000806, SHCSR = 0x000f0008, INT\_ACTIVE0 = 0x00000000
- [C] Platform-----: INT\_ACTIVE1 = 0x00000000, CFSR = 0x778fc80f, HFSR = 0x01000000, DFSR = 0x00000000
- [C] Platform-----: MMAR/BFAR = 0x00000000, AFSR = 0xe000ed34, Ret0 = 0x00000000, Ret1 = 0x080092b5
- [C] Platform-----: Ret2 = 0x08038d4f, Ret3 = 0x08016f17, Ret4 = 0x08028cff, Ret5 = 0x08010973

[C] Platform-----: Dat0 = 0x08038c71, Dat1 = 0x7f33fbff,

#### 5.6 OpenThread Diagnostics Support

OpenThread diagnostic capabilities can be enabled for source and certification library project builds by installing the Diagnostics component.

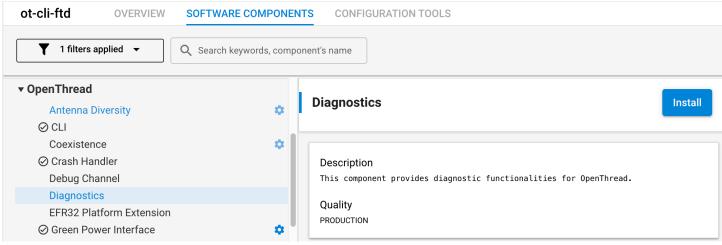


Figure 5-7 Diagnostics Component

This component will enable the standard OpenThread Factory Diagnostics Module in the project and implement the `diag` command in the OpenThread CLI. See <a href="https://github.com/openthread/openthread/blob/main/src/core/diags/README.md">https://github.com/openthread/openthread/openthread/blob/main/src/core/diags/README.md</a> for a detailed reference on the diagnostics CLI commands.

## **Simplicity Studio**

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



www.silabs.com/IoT



www.silabs.com/simplicity



www.silabs.com/quality



Support & Community www.silabs.com/community

#### Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the information supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon La

#### **Trademark Information**

Silicon Laboratories Inc.<sup>®</sup>, Silicon Laboratories<sup>®</sup>, Silicon Labs<sup>®</sup>, SiLabs<sup>®</sup> and the Silicon Labs logo<sup>®</sup>, Bluegiga<sup>®</sup>, Bluegiga Logo<sup>®</sup>, EFM<sup>®</sup>, EFM32<sup>®</sup>, EFR, Ember<sup>®</sup>, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals<sup>®</sup>, WiSeConnect, n-Link, EZLink<sup>®</sup>, EZRadio<sup>®</sup>, EZRadio<sup>®</sup>, Gecko<sup>®</sup>, Gecko OS, Gecko OS Studio, Precision32<sup>®</sup>, Simplicity Studio<sup>®</sup>, Telegesis, the Telegesis Logo<sup>®</sup>, USBXpress<sup>®</sup>, Zentri, the Zentri logo and Zentri DMS, Z-Wave<sup>®</sup>, and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc. 400 West Cesar Chavez Austin, TX 78701 USA

## www.silabs.com