AN1295: Developing with Thread 1.2

Thread 1.2 builds on Thread 1.1's robust foundation, and defines extensions to the mandatory base specification, low-power enhancements and optional advanced features, including Commercial Extensions. Thread 1.2 devices and networks are fully interoperable with existing Thread 1.1 devices and networks.

KEY POINTS

- Developing Thread 1.2 SoC applications
- Developing Thread 1.2 RCP applications
- Working with the OpenThread 1.2 Backbone Border Router

The features of Thread 1.2 are designed to improve the scalability of Thread networks by making them more responsive and capable of higher network density. New low-power features further reduce the power consumption, channel utilization, and communication latency of Sleepy End Devices (SEDs). New features for commercial enterprise networks allow for communication between multiple Thread networks and non-Thread devices in a large infrastructure deployment.

Silicon Labs provides components and configuration options that enable you to configure Thread 1.2 features with sample applications. Thread 1.2 is compatible with EFR32MG1x and EFR32MG2x SoCs, RCPs, and modules. This application note assumes you have a basic understanding of how Thread is implemented on EFR32 devices. For more information, see *UG103.11: Thread Fundamentals*.

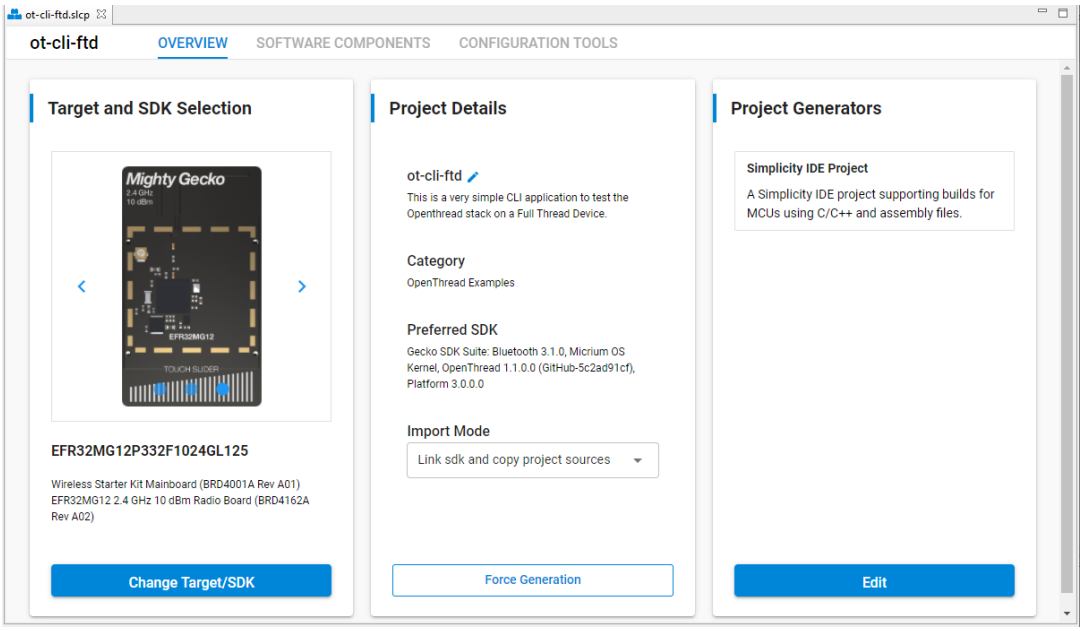
1 Developing Thread 1.2 SoC Applications

Silicon Labs provides a number of sample SoC OpenThread applications. You can modify these to work with Thread 1.2.

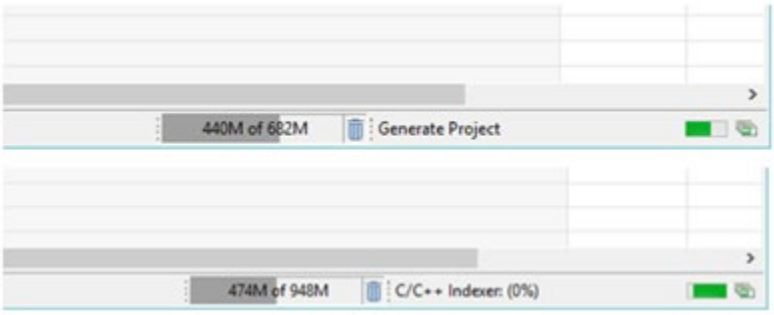
1.1 Enable Thread 1.2 Features on the ot-cli-ftd Application

The following example shows how to enable 1.2 features on the ot-cli-ftd application.

1. Select **ot-cli-ftd** as an example for your SoC image.
2. With your target part connected to your computer, open the Simplicity Studio 5 File menu and select **New > Silicon Labs Project Wizard**. The Target, SDK, and Toolchain Selection dialog opens. Click **NEXT**.
3. The Example Project Selection dialog opens. Use the Technology Type and Keyword filters to search for **ot-cli-ftd** as an example for the default SoC image. Select it and click **NEXT**.
4. 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.**
5. The Simplicity IDE Perspective opens with the Project Configurator open to the 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.



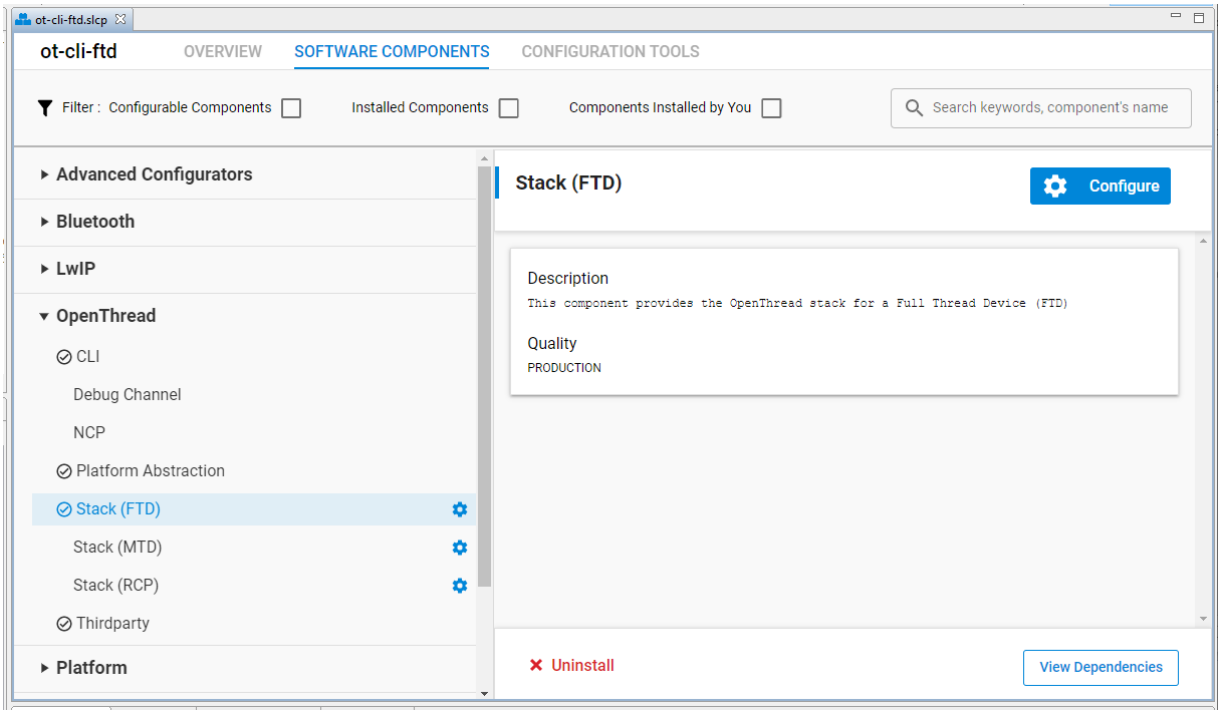
6. Make any configuration changes to the software components, as described in the next section. You can see autogeneration progress in the bottom right of the Simplicity IDE perspective. Make sure that progress is complete before you build.



7. Compile and flash the application image as described in *QSG170: Silicon Labs OpenThread Quick Start Guide*.

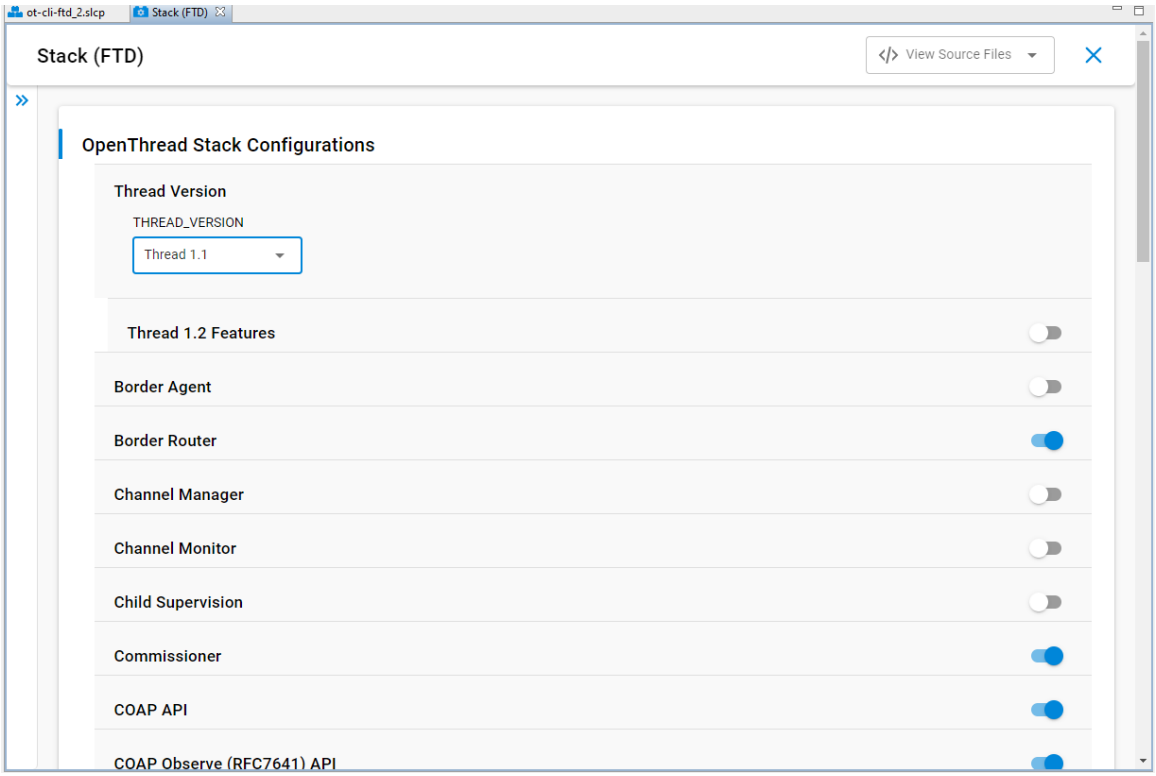
1.2 Configure OpenThread Options in the SoC Images Using Simplicity Studio 5

1. Under the SOFTWARE COMPONENTS tab in your FTD project (.slcp), expand the OpenThread group. Select the **Stack (FTD)** entry for an FTD build.

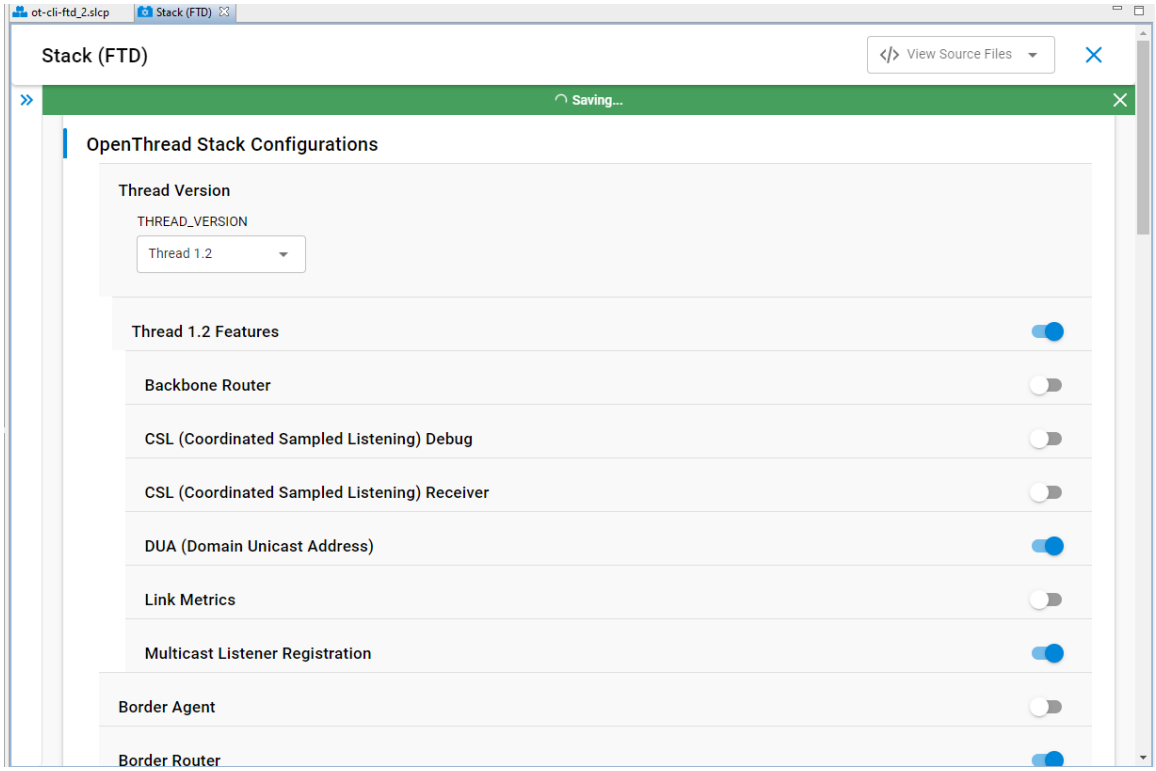


Note: You can select the Configurable Components and Installed Components checkboxes to filter to only those components you can configure successfully.

2. Click **Configure** to change the settings associated with the OpenThread build. The various build options are explained in the OpenThread documentation at https://openthread.io/guides/build.



3. **For Thread 1.2 features only**:
   - Under **Thread Version**, select Thread 1.2. Thread 1.2 must be selected as the Thread version for Thread 1.2 features to function.
   - Enable **Thread 1.2 Features**. A list of features is displayed. The description for each feature indicates whether is mandatory, optional, or recommended. **Do not enable** these features for a Thread 1.1 application. The following figure shows a Thread 1.2 configuration with mandatory features enabled.
     - `BACKBONE_ROUTER` (optional): If enabled, `BORDER_AGENT` (optional but recommended) and `BORDER_ROUTER` (mandatory) should also be enabled.
     - `CSL_DEBUG` (optional): Enable for debugging CSL features.
     - `CSL_RECEIVER` (optional but recommended): If enabled, low-power Coordinated Sampled Listening (CSL) features are available for use with Wake-on End devices (WEDs). A SED with this option selected is called a Synchronized SED (SSED).
     - `DUA` (mandatory): Enable Thread 1.2 Domain Unique Address feature for use in Thread domains. For more information, see *UG103.11 Thread Fundamentals*.
     - `LINK_METRICS` (recommended): Enable low-power link metrics and link metric probing features.
     - `MLR` (mandatory): Enable Multicast Listener Registration features in Thread 1.2 to enable off-mesh multicasts and multicasts in Thread domains. For more information, see *UG103.11 Thread Fundamentals*.
   - `Service Entries in Thread Network Data` (optional): Select if `BORDER_ROUTER` is enabled to indicate availability of Thread Management Framework (TMF) Network Data services as defined in the Thread specification.



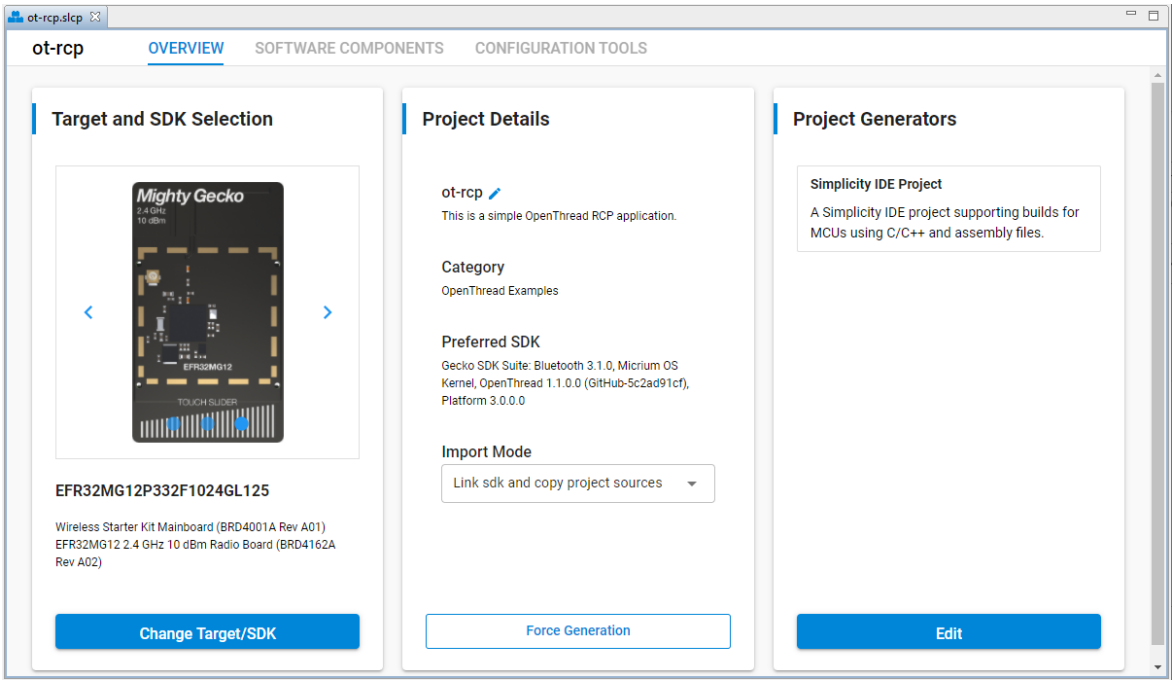
2 Developing Thread 1.2 RCP Applications

Silicon Labs provides a sample OpenThread RCP application. You can modify it to work with Thread 1.2.

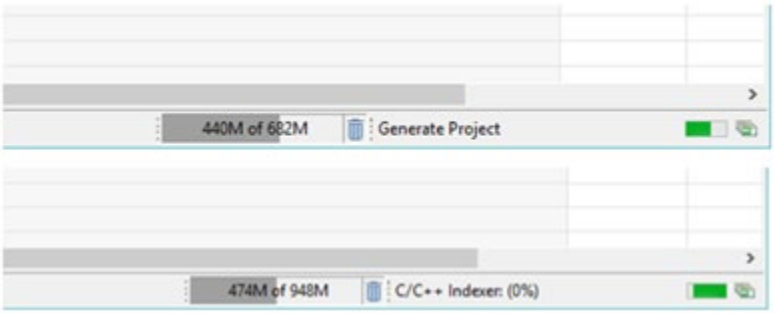
2.1 Enable Thread 1.2 Features on the ot-rcp Application

The following example shows how to enable Thread 1.2 features on the ot-rcp application.

1. Select **ot-rcp** as an example for your RCP image.
2. With your target part connected to your computer, open SSv5's File menu and select **New > Silicon Labs Project Wizard**. The Target, SDK, and Toolchain Selection dialog opens. Click **NEXT**.
3. The Example Project Selection dialog opens. Use the Technology Type and Keyword filters to search for **ot-rcp** as an example for the default SoC image. Select it and click **NEXT**.
4. 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.**
5. The Simplicity IDE Perspective opens with the Project Configurator open to the 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.



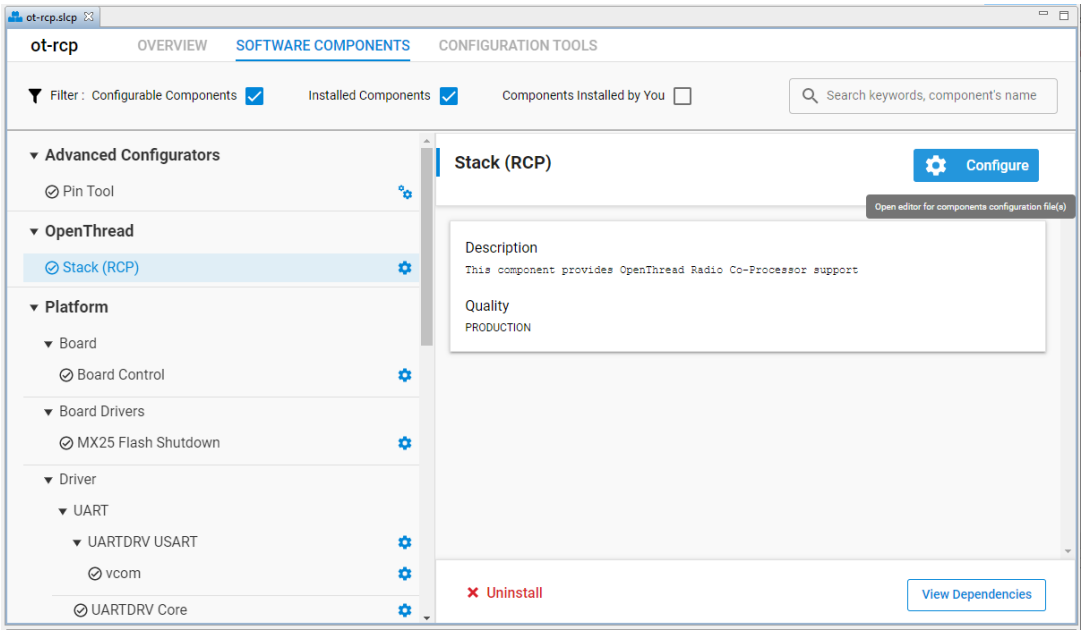
6. Make any configuration changes to the software components, as described in the next section. You can see autogeneration progress in the bottom right of the Simplicity IDE perspective. Make sure that progress is complete before you build.



7. Compile and flash the application image as described in *QSG170: Silicon Labs OpenThread Quick Start Guide*.

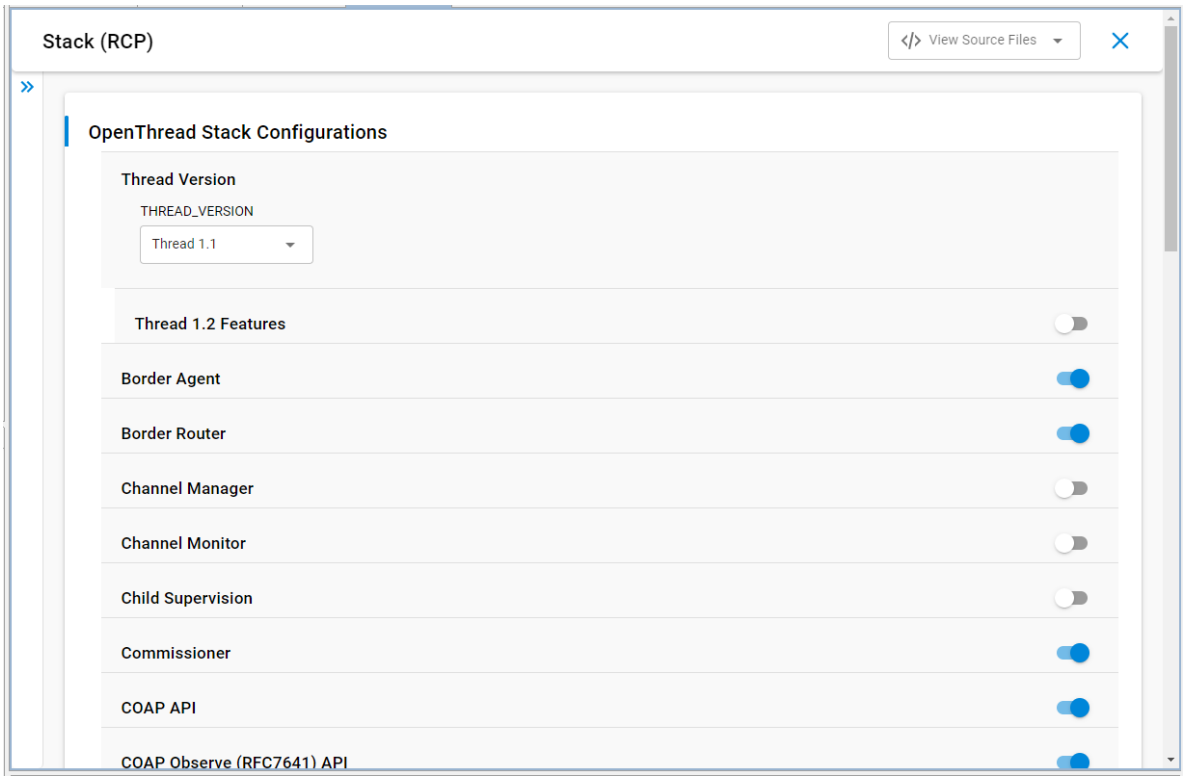
2.2 Configure OpenThread Options in the RCP Images Using Simplicity Studio 5

1. Under the SOFTWARE COMPONENTS tab in your RCP project (.slcp), expand the OpenThread group. Select the **Stack (RCP)** entry for an RCP build.

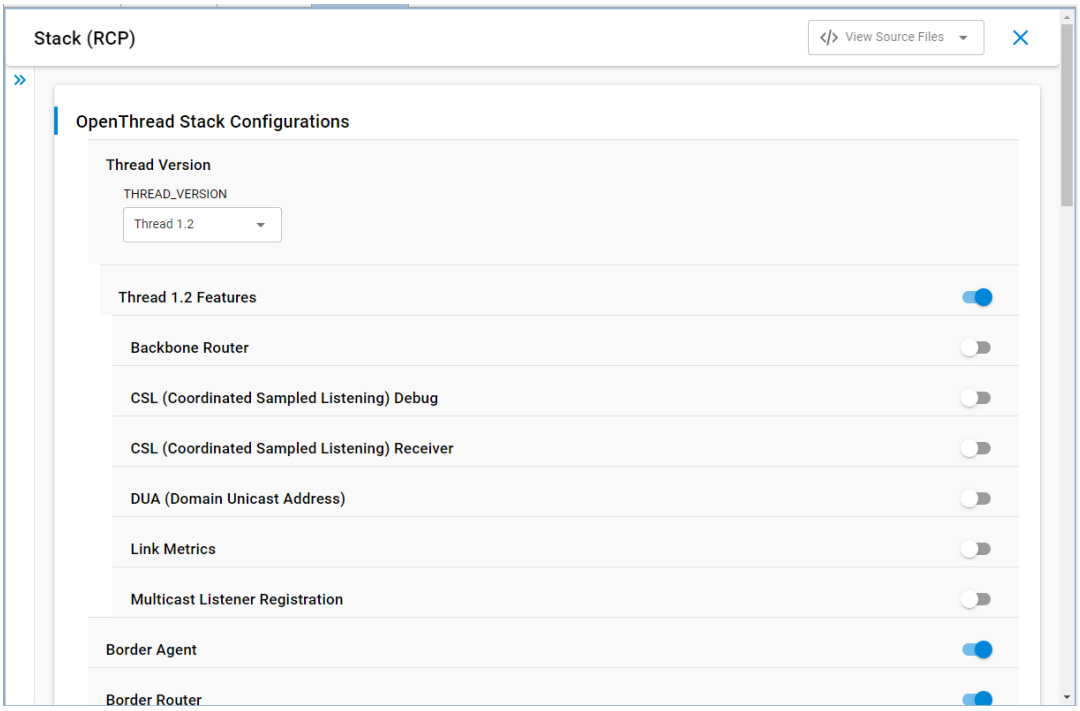


Note: You can select the Configurable Components and Installed Components checkboxes to filter down to only those components you can configure successfully.

2. Click **Configure** to change the settings associated with the OpenThread build. The various build options are explained in the OpenThread documentation at https://openthread.io/guides/build.



3. **For Thread 1.2 features only:**
   - Under **Thread Version**, select Thread 1.2. Thread 1.2 must be selected as the Thread version for Thread 1.2 features to function.
   - Enable **Thread 1.2 Features**. A list of features is displayed. The description for each feature indicates whether is mandatory, optional, or recommended. **Do not enable** these features for a Thread 1.1 application. The following figure shows a Thread 1.2 configuration with mandatory features enabled.
     - `BACKBONE_ROUTER` (optional). If enabled, `BORDER_AGENT` (optional but recommended) and `BORDER_ROUTER` (mandatory) should also be enabled.
     - `CSL_RECEIVER` (optional but recommended): If enabled, low-power Coordinated Sampled Listening (CSL) features are available for use with Wake-on End devices (WEDs). A SED with this option selected is called a Synchronized SED (SSED).
     - `CSL_DEBUG` (optional): Enable for debugging CSL features.
     - `LINK_METRICS` (recommended): Enable low-power link metrics and link metric probing features
   - `Service Entries in Thread Network Data` (optional): Select if `BORDER_ROUTER` is enabled to indicate availability of Thread Management Framework (TMF) Network Data services as defined in the Thread specification.



3 Working with the OpenThread 1.2 Backbone Border Router

Refer to *AN1256: Using the Silicon Labs RCP with the OpenThread Border Router* for detailed instructions on how to build an OpenThread Border Router for Raspberry Pi 3B+ or above. You have to use a Thread 1.2 RCP with an OpenThread 1.2 Backbone Border Router.

You can install a pre-built Docker container with OpenThread 1.2 Backbone Border Router:

https://hub.docker.com/r/siliconlabsinc/openthread-backbone-border-router/tags

Or you can manually install an OpenThread 1.2 Backbone Border Router by following the steps in *AN1256: Using the Silicon Labs RCP with the OpenThread Border Router* or https://openthread.io/guides/border-router/build.

Simplicity Studio

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

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

SILICON LABS

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

http://www.silabs.com