



AN1264: Using Open Thread with FreeRTOS™

This version of AN1264 has been deprecated. For the latest version, see docs.silabs.com.

The Silicon Labs OpenThread SDK provides support for running on top of FreeRTOS, a full-featured Real-Time Operating System (RTOS) for microcontrollers and small microprocessors. Support for FreeRTOS is integrated into Simplicity Studio.

FreeRTOS is supported on the EFR32MGxx family. For documentation on Free RTOS, see <https://www.freertos.org/>.

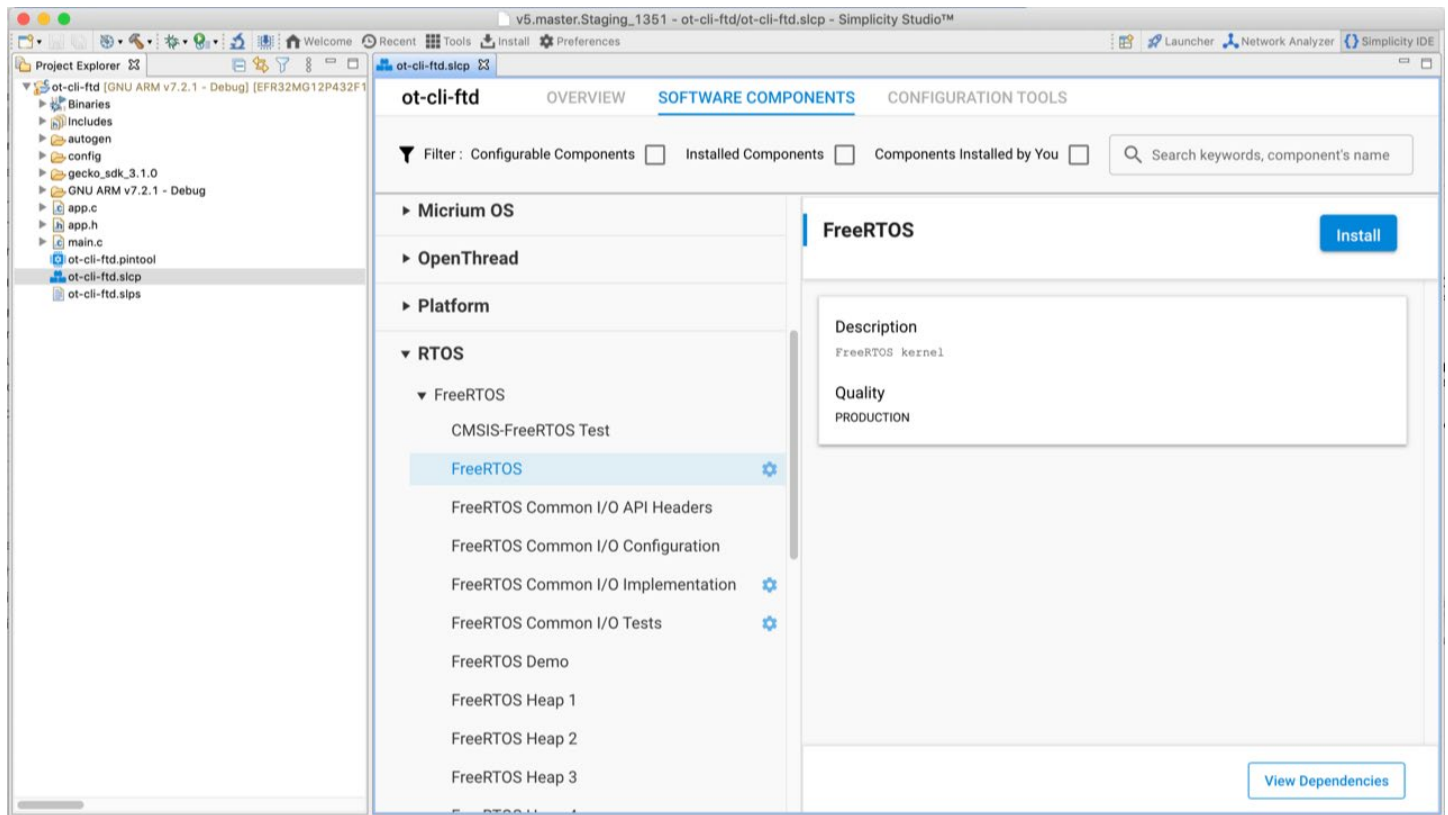
KEY POINTS

- Introduces FreeRTOS.
- Describes how to install FreeRTOS.
- Describes how to work with the FreeRTOS component most effectively.

1 Getting Started

Integrating FreeRTOS into your application is simply a matter of installing the FreeRTOS component for the project in Simplicity Studio. The ot-ble-dmp sample application runs on FreeRTOS by default and you can add FreeRTOS to any OpenThread project. The ot-cli-ftd sample application is a good starting example.

1. In your project, double-click the **.slcp** file for the project in the Project Explorer to open the project window.
2. Click the SOFTWARE COMPONENTS tab to see a complete list of Component categories.
3. Find the FreeRTOS component located under RTOS > FreeRTOS in the list of components.
4. Select **FreeRTOS** and then click **Install**.



This component brings all the FreeRTOS kernel files into your project, along with some integration files and some additional components it depends on.

5. Click **View Dependencies** to display the components.

One dependency is the CMSIS-RTOS2 component, which is an RTOS abstraction layer used by the integration files. Silicon Labs recommends that application developers use the FreeRTOS API directly rather than using the CMSIS-RTOS2 API.

Another dependency is the heap implementation used by FreeRTOS. FreeRTOS comes with five different heap implementations. Each appears as a component. By default, the FreeRTOS Heap 4 component is added to the project. You can change this by selecting a different heap component and clicking **Install**. For example, FreeRTOS HEAP 3 uses the system `malloc()` and `free()` implementation and is a common choice.

2 Main Function and Task Model

The FreeRTOS component is designed to be used along with the standard Silicon Labs main.c template. The standard main function works for both bare metal and kernel-based projects and takes care of all the required system initializations and task creation.

For OpenThread running on FreeRTOS, a single task is created that runs both the OpenThread stack and the application logic. It is not safe to call the OpenThread API from other tasks.

In a bare metal OpenThread application, application logic is placed in the `app_process_action()` callback. Instead, when running on FreeRTOS, application logic is placed in the `sl_ot_rtos_application_tick()` callback.

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

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 Labs product in such unauthorized applications.

Trademark Information

Silicon Laboratories Inc.[®], Silicon Laboratories[®], Silicon Labs[®], SiLabs[®] and the Silicon Labs logo[®], Bluegiga[®], Bluegiga Logo[®], EFM[®], EFM32[®], EFR, Ember[®], Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals[®], WiSeConnect, n-Link, EZLink[®], EZRadio[®], EZRadioPRO[®], Gecko[®], Gecko OS, Gecko OS Studio, Precision32[®], Simplicity Studio[®], Telegesis, the Telegesis Logo[®], USBXpress[®], Zentri, the Zentri logo and Zentri DMS, Z-Wave[®], 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