Connected Lighting Platform

Getting Started Guide

Attention: The Connected Lighting Platform is powered by AC Mains, and exposed to high voltage. Only trained personnel should manipulate and operate on the system. Ensure that all boards are properly connected before powering, and that power is off before disconnecting any boards. Do not attempt re-connecting any of these boards while under operation. During operation, some surfaces may become hot; the LED board can dissipate up to 55 watts. Use appropriate heat sink. The lighting platform is delivered with high luminous LEDs boards. Before any operation, use suitable Personal Protective Equipment (PPE) to protect the eyes. Failure to comply with appropriate safety precautions may result in personal injury or death, or equipment damage.

1. Overview

The Connected Lighting Platform is a modular kit for developing a complete industrial lighting solution with power supply, LED driving system and Bluetooth® Low Energy connectivity through the *RSL10 Sense and Control* mobile app.

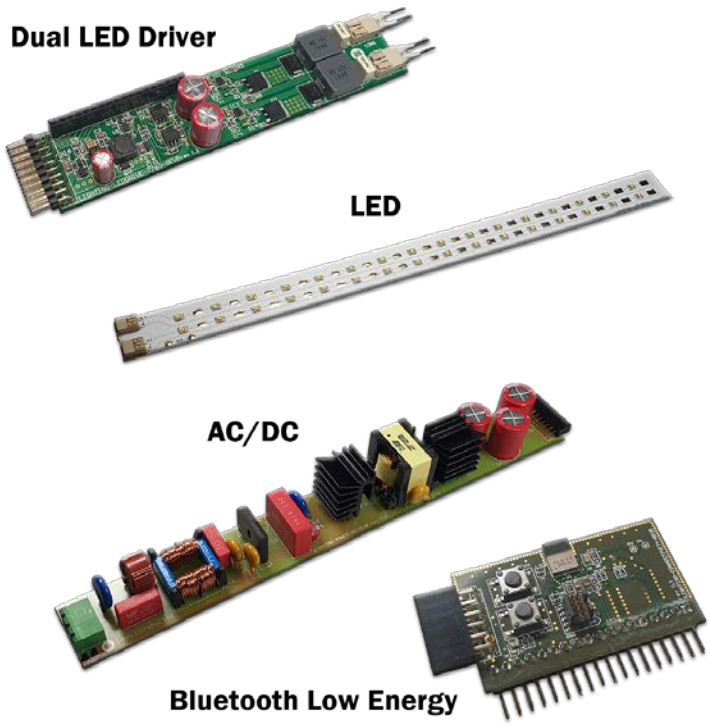


Figure 1. Connected Lighting Platform

The following modules are available as part of the platform (LIGHTING-1-GEVK):

- Dual LED driver
- AC/DC power supply
- Bluetooth Low Energy connectivity
- LED

For high-power connectivity, an IEEE 802.3bt compliant Power-Over-Ethernet (PoE) board is available separately.

Features

- High-power lighting features
  - Up to 2 Strings of 16 LEDs (7000 Lumen)
  - Dual independent LED Channel
  - White balance control (12-bit Dimmer from 0 to Max)
- Multiple connectivity options
  - Bluetooth® Low Energy
  - Power Over Ethernet (PoE)
  - Zigbee™ GreenPower (Q2/20)
- Compliant with multiple industry standards
- High efficiency power conversion (>90% at full load)

2. Prerequisites

Hardware

- Connected Lighting Platform (LIGHTING-1-GEVK)
- J-link Ultra+ Debug Probe
- AC power cord with (example here)

If Using the PoE Power Module

- Power Over Ethernet Power Module (LIGHTING-POWER-POE-GEVB)
- Phihong POE90U-1BT PSE (Amy to add note)

Software

- Bluetooth® IoT Development Kit CMSIS Pack (01.11.02 or higher)
- ON Semiconductor IDE Installer
- RSL10 Software Package
- RSL10 FOTA Package

Mobile Apps

- RSL10 Sense and Control (IoS®, Google Play™)
- RSL10 FOTA (IoS, Google Play)

3. Hardware Set-up

3.1 Setting up the Connected Lighting Platform Modules

1. Connect the Bluetooth Low Energy connectivity module to the LED driver module as shown in **Figure 2** below.

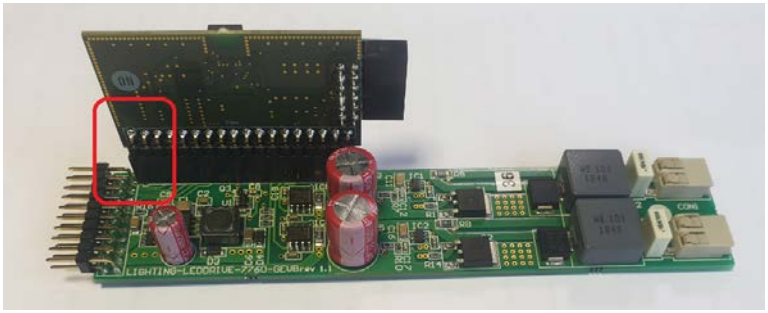


Figure 2. Connecting the Bluetooth Low Energy Connectivity Module to the LED driver module

2. Connect the remaining modules as depicted in **Figure 3.**



Figure 3. Connection of the whole string of Connected Lighting Platform modules

3.1.1. Powering the Platform using the AC/DC Module

Using an AC power cable, connect the phase, neutral and PE according to **Figure 4.**

NOTE: Ensure that all modules are connected properly before powering the platform.

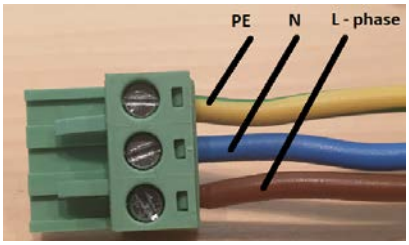


Figure 4. AC/DC Power Cord Wiring to AC Connector

3.1.2. Powering using the PoE Power Board

Connect OUT of the **Phihong POE90U-1BT PSE** power injector to the input connector of the PoE module.



Figure 5. Lighting Platform Hardware Setup with PoE power supply

4. RSL10 Sense and Control Mobile App

The RSL10 Sense and Control application enables monitoring and control of various peripherals connected to RSL10-based platforms. Available on IoS and Google Play, the app allows you to set and view light intensity, step of shift and current and voltage values of the Connected Lighting Platform.

4.1 Using the RSL10 Sense and Control Mobile App

After downloading and opening the app, a pairing message will prompt you to connect to the Lighting Platform over Bluetooth® Low Energy.

Note: In the event that you are having difficulty connecting to or recognizing a device, ensure you are using the latest version of the app.

1. Display measured LED channel currents and system voltage telemetry data.
2. Independently set PWM duty
3. Cycle of each LED channel.
4. Adjust PWM duty slider behavior by rounding values with given step and enable setting of both channels at the same time with single slider.
5. When powered by PoE, display information about negotiated power limits between PoE controller and PSE.
6. Swipe right to show graphs with past PWM settings and LED channel currents.

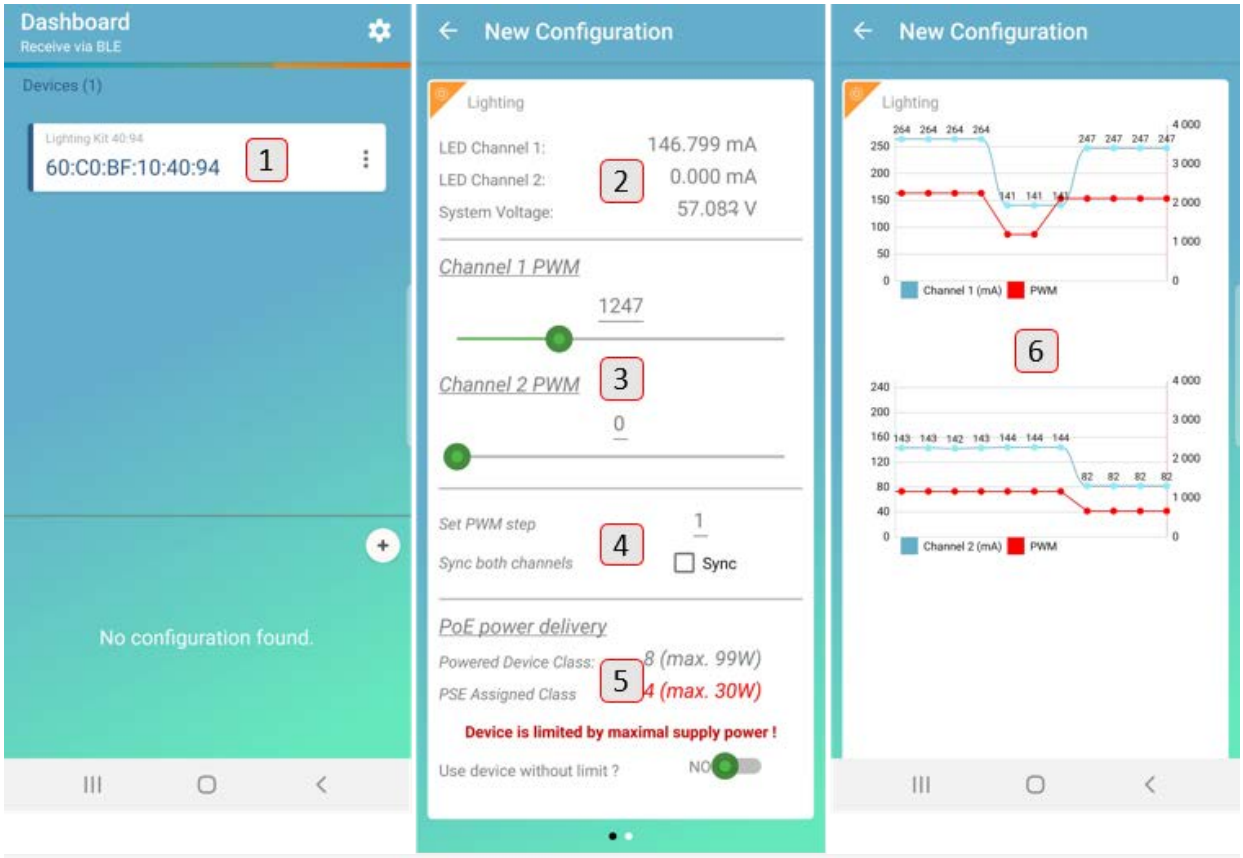


Figure 6. *RSL10 Sense and Control* User Interface (Android™)

4.1.1 Lighting Control

The lighting platform features two stripes of LED with different light colors. Using the RSL10 Sense and Control app, the LEDs can be controlled individually or synchronized. There are two seek bars to adjust the PWM (light intensity) for each stripe. The lighting platform's PWM range is 4000 steps. The **Sync** box enables the synchronized control of the two stripes.

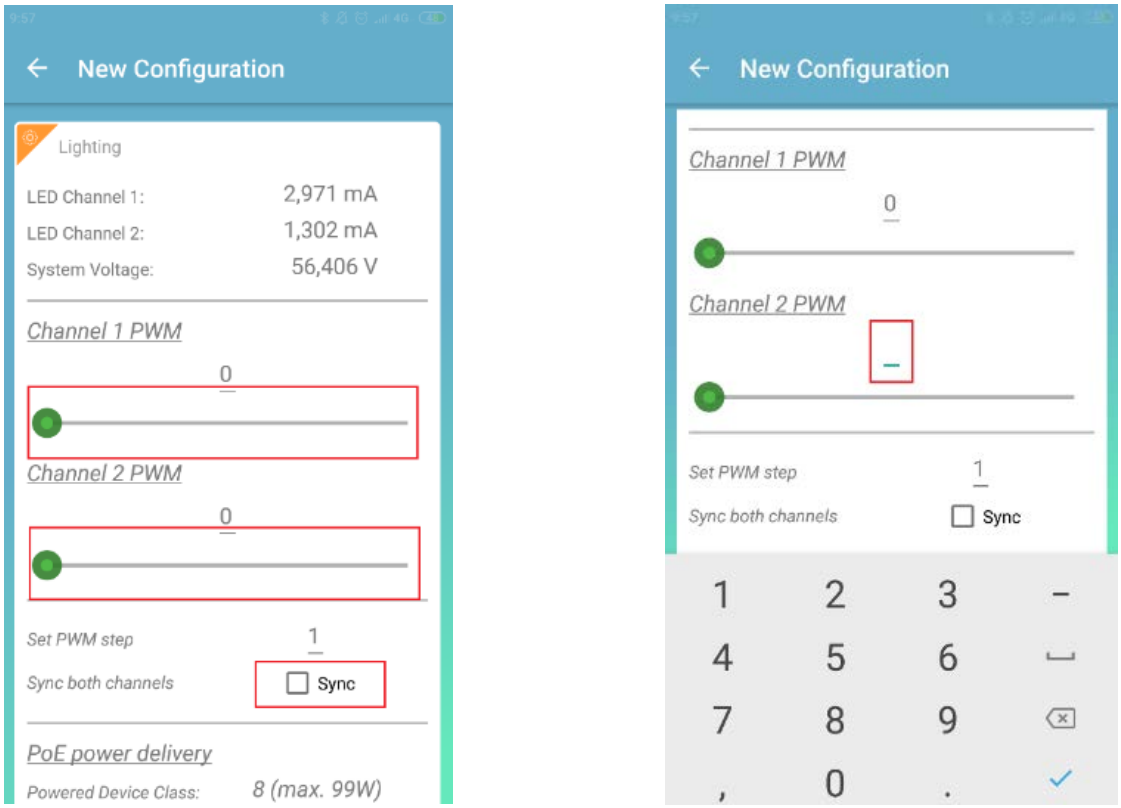


Figure 7. RSL10 Sense and Control App Panel

Another option to adjust PWM (light intensity) is to type the desired value in the window over the particular seek bar. Select the value window over the seek bar, and enter the value what you want to set. Confirm by selecting **OK**.

4.2 Setting the PWM Step

The PWM step or step of shift seek bar is value, which determines the difference between the two following steps on the progress bar. The PWM step is automatically set to 1 This can be changed by selecting the current step, and entering a new value.

4.3 Currents and Voltage Values

The LED strip operating current and system voltage values are displayed at the top of the app screen. These values are updated regularly to match the current PWM (Light Intensity) setting. Currents and PWM values are listed in plots. The first plot corresponds to the first LED channel; the bottom plot to the second LED channel.

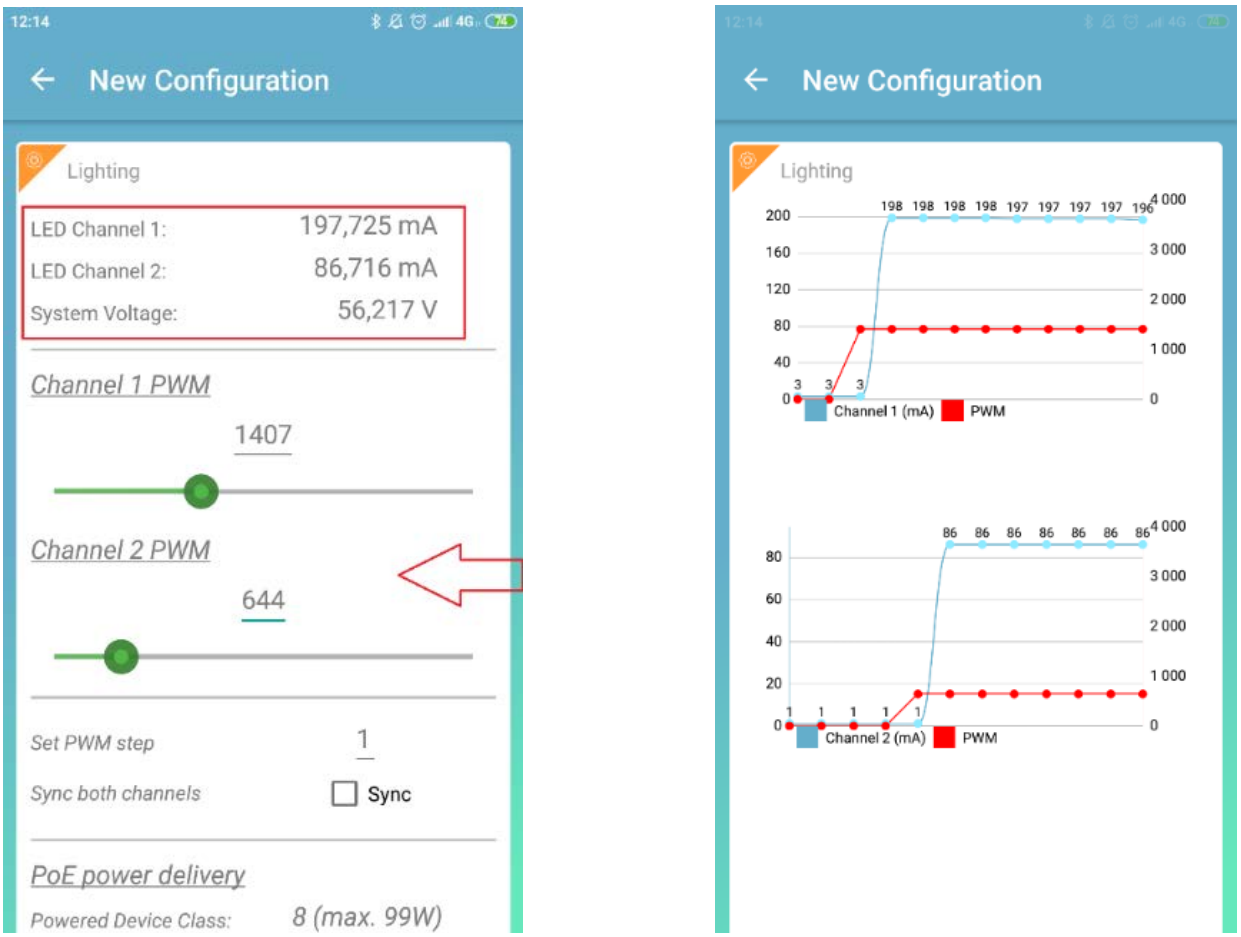


Figure 8. *RSL10 Sense and Control* App Telemetry Panel

4.4 Power Supply

4.4.1 AC/DC

The RSL10 Sense and Control App does not provide any limitation configuration as the AC/DC is compatible with the AC mains voltage range.

4.4.2 PoE Power Supply

If powering the connected lighting platform via PoE, the platform detects the type of power supply and sends a notification to the mobile app. Power limits, required power class of the Powered Device (PD and Power Supply Equipment (PSE) are displayed at the bottom of the screen, as shown in **Figure 9.**

If the power supply is of a lower class than required (Class 8, 90 W), the Lighting Kit will reduce power of light, causing the PWM range to be limited. For example, if power supply class is 4 (max. 30W), the maximum PWM range for both LED channels will be only 3600 steps. You can override this limit by switching the "Use Device without Limit" option to **Yes**.

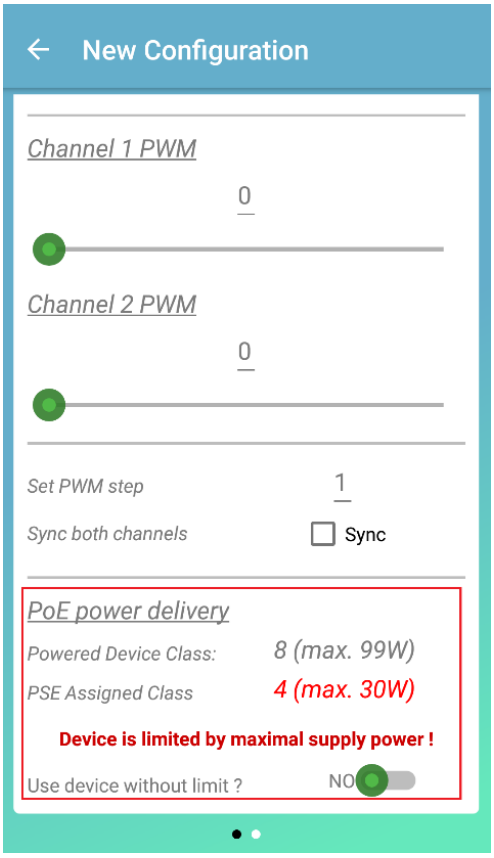


Figure 9. Power Limits Alert

4.2 Software Development Kit

The RSL10 firmware runs on FreeRTOS™ operation system, included as part of the CMSIS SDK release with detailed description of the code and services used in the threads.

The Connected Lighting Platform utilizes the following Bluetooth® Low Energy services:

- **Light Control Service**, used by connected peer devices to read and change state of kits LED strings.
- **Telemetry Service,** exposes variables measured by the platform to peer devices. This includes current flowing through each LED driver and system voltage.

- **PoE Power Delivery Service***, peer device can retrieve information about PoE imposed power limits on the device that were negotiated between the PoE injector and platform.

*If using the PoE power board

4.2.1 Light Control Service

This service exposes a Dimmable Light characteristic for each available LED string. The value of this characteristic directly relates to the duty cycle that will be applied to the corresponding LED channel. By default, the allowed duty cycle range is from 0 (0%) up to 4000 (100%). When multiple LED channels are present, the characteristic will always have a Characteristic Presentation Format descriptor that reports channel number in its description value, starting from 1 (first) up to the number of channels (second in this case).

If the kit is powered by PoE, there might be an imposed additional duty cycle limit so that the sum of all channel currents does not exceed capability of used PoE injector. Refer to Section **4.2.3 PoE Powered Device Service** for additional information.

4.2.2 Telemetry Service

The Telemetry service is used to report various system variables measured by the Lighting Kit. This includes currents flowing through LED channels and system voltage.

4.2.3 PoE Powered Device Service

The PoE Powered Device service provides information on the power levels for both the Powered Device and Power Supply Equipment.

4.2.3.1 PD Class Characteristic

The PD Class Characteristic reports the power level of Powered Device (Lighting Platform). The PD power level class will always be 8, which requests 70 Watts from the PSE to power both LED strings at full power.

4.2.3.1 PSE Class Characteristic

The PSE Assigned PD Class characteristic will change depending on the PoE injector. This is the device class assigned to the lighting platform during negotiation between injector and the PoE controller. If the assigned power level is lower than requested, additional duty cycle limits will be applied to the LED strings. Duty limits will be applied to the sum of all channel duty cycles for different PoE power level classes.

4.2.3.1 PSE PD Class Override

The *PSE PD Class Override* characteristic allows the peer device to disable the power limits described above. This can be used in the case of 2-pair 802.3at injectors that can provide

more than 71 W of power. Write 1 to this characteristic to disable any power limiting or 0 to apply limits.

Installation Prerequisites

1. Install 64-bit version of Java from https://www.java.com/en/download/
2. Install J-Link Version 6.32i or later from https://www.segger.com/downloads/jlink (select J-Link software and documentation pack)
3. Download and install "ON Semiconductor IDE Installer" from https://www.onsemi.com/PowerSolutions/product.do?id=RSL10
    a. Download the ''RSL10 SDK Getting Started Guide'' and RSL10 CMSIS pack under ''RSL10 Software Package'' from the above site. All of these are highlighted in the picture below. Save the CMSIS pack in a folder, for example, C:\cmsis_packs



4. Download the B-IDK CMSIS pack from https://www.onsemi.com/B-IDK and save it in the same folder as the RSL10 CMSIS pack (see 3.a above).
5. CMSIS pack at item 4. is dependent on ARM CMSIS pack as well. Please install ARM CMSIS pack 5.5.1 or higher after download from: https://github.com/ARM-software/CMSIS_5/releases
6. CMSIS pack at item 4. is also dependent on ARM CMSIS – FreeRTOS version 10.2.0 or higher for users exposed to design the code under FreeRTOS with RSL10: https://github.com/ARM-software/CMSIS-FreeRTOS/releases

Importing the CMSIS-Packs

Once all packs are successfully imported, they can be viewed in the CMSIS pack manager perspective as shown below (Figure 10).

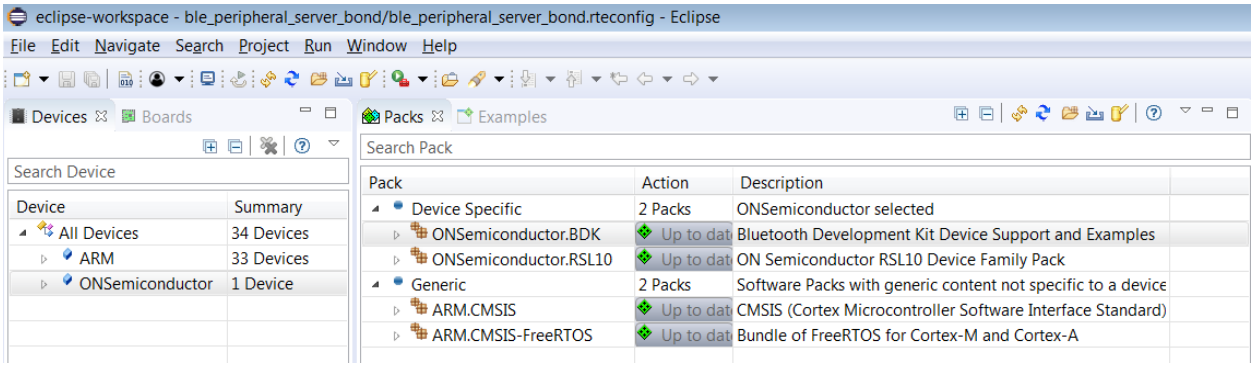


Figure 10. CMSIS Pack Manager Perspective

Lighting kit Use cases

Once the CMSIS packs are installed into ON Semiconductor IDE, user can access sample code related use cases under the Examples menu (Figure 11).

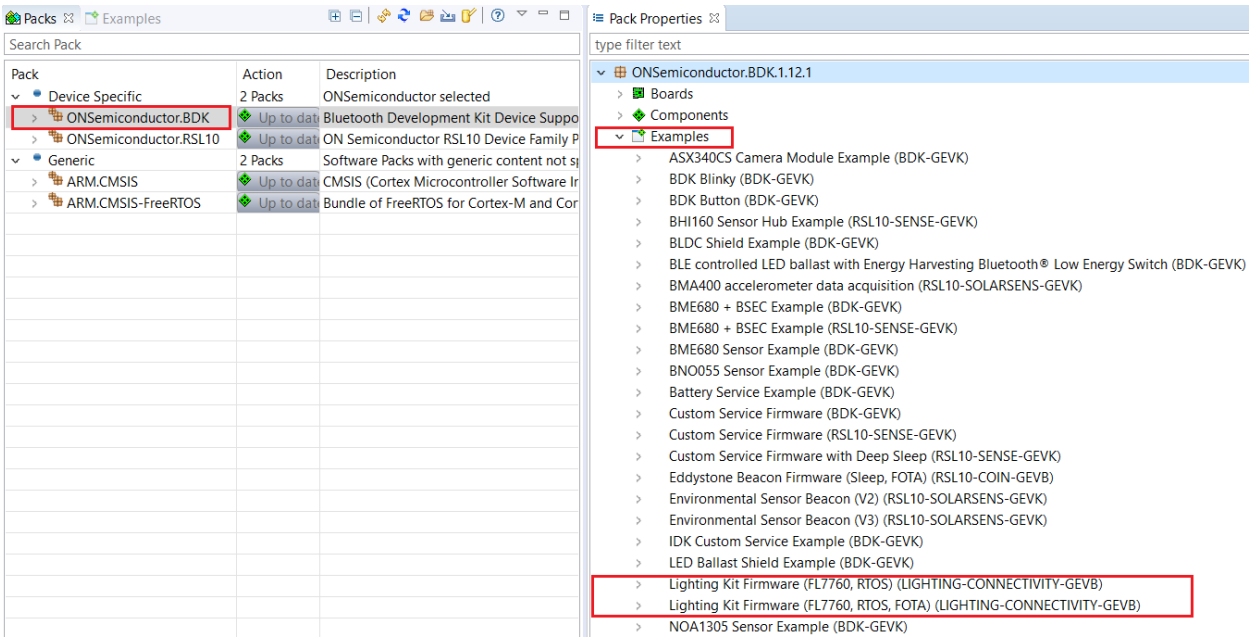
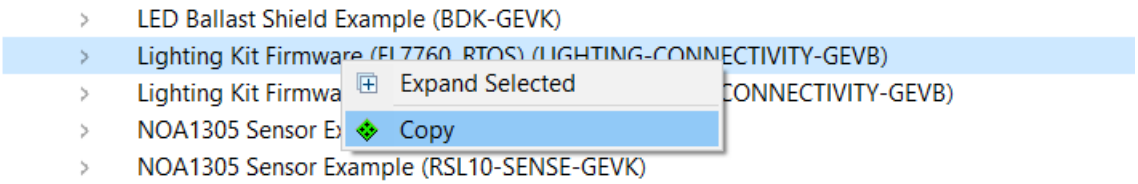
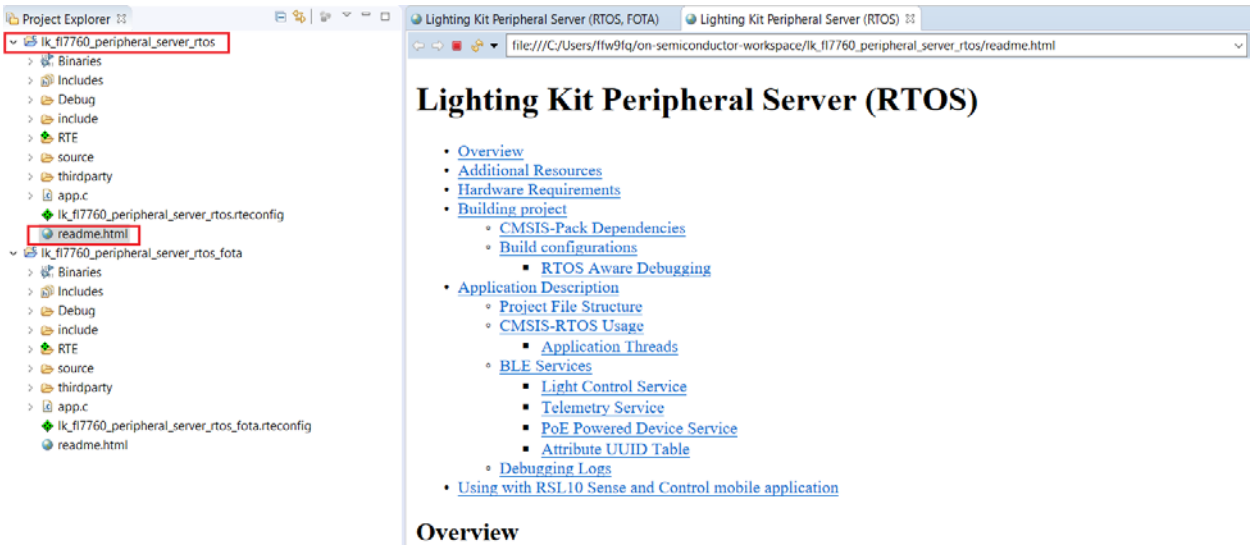


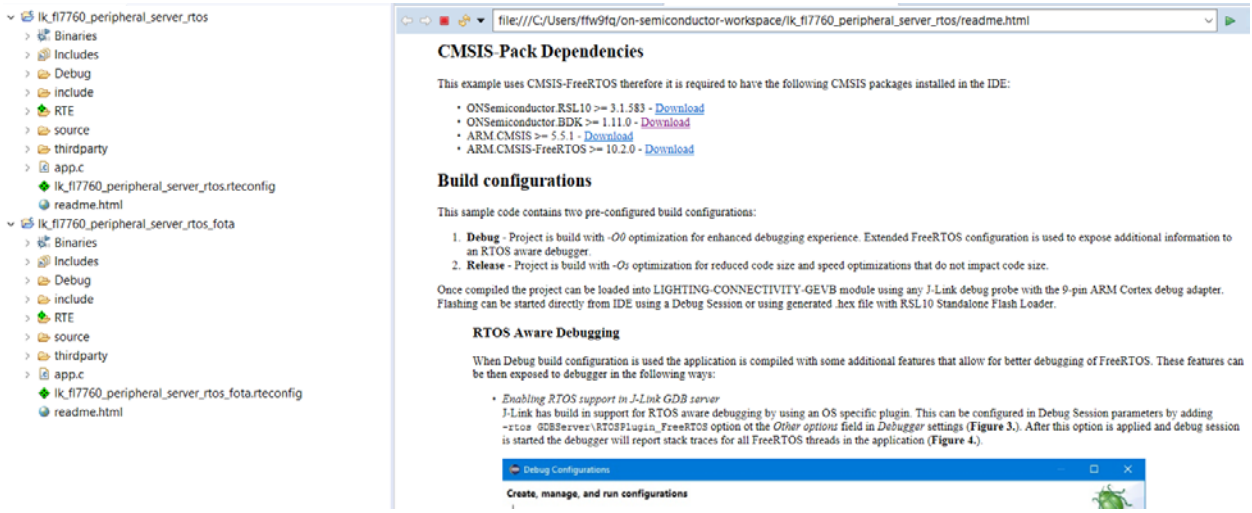
Figure 11. FreeRTOS based use cases supported by the Lighting Platform

Import of each sample applications is simple by right mouse click onto specific Example and copying into IDE's workspace



Once the project is copied, user can access all detailed help and setup of RTOS software under the *readme.html*. It includes also file structure, description of BLE services and special features like RTOS aware debugging etc.





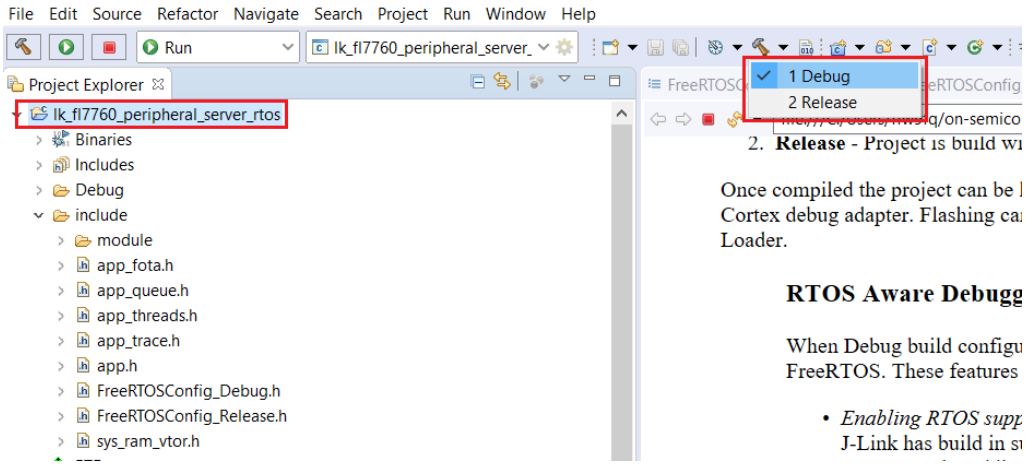
Build configurations

This sample code contains two pre-configured build configurations:

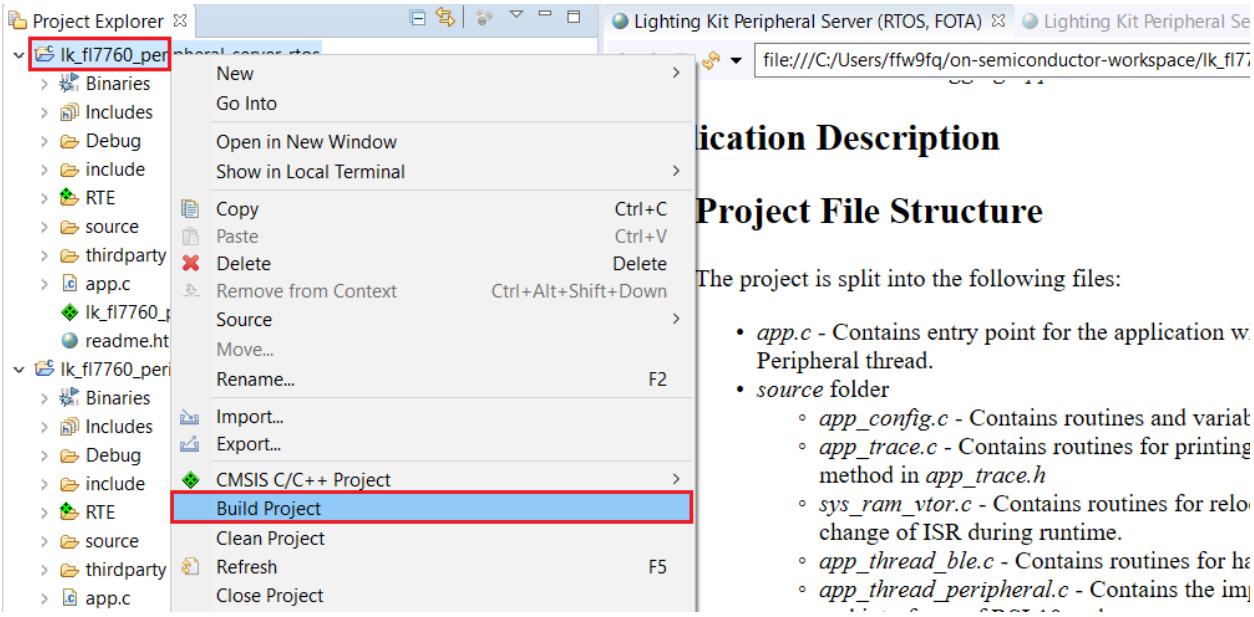
1. **Debug** - Project is build with *-O0* optimization for enhanced debugging experience. Extended FreeRTOS configuration is used to expose additional information to an RTOS aware debugger.
2. **Release** - Project is build with *-Os* optimization for reduced code size and speed optimizations that do not impact code size.

Once compiled the project can be loaded into LIGHTING-CONNECTIVITY-GEVB module using any J-Link debug probe with the 9-pin ARM Cortex debug adapter. Flashing can be started directly from IDE using a Debug Session or using generated .hex file with RSL10 Standalone Flash Loader.

User will highlight with mouse project workspace and under ''hummer'' icon, select between Debug or Release version.



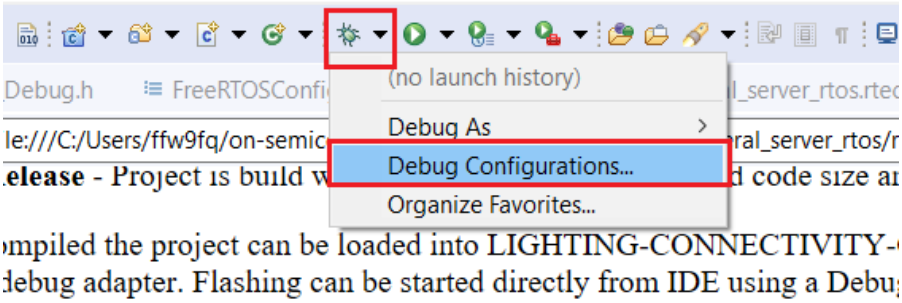
Optionally right mouse click onto the project workspace, user can build the project that generates binaries ready for flashing the application into RSL10 Connectivity module.



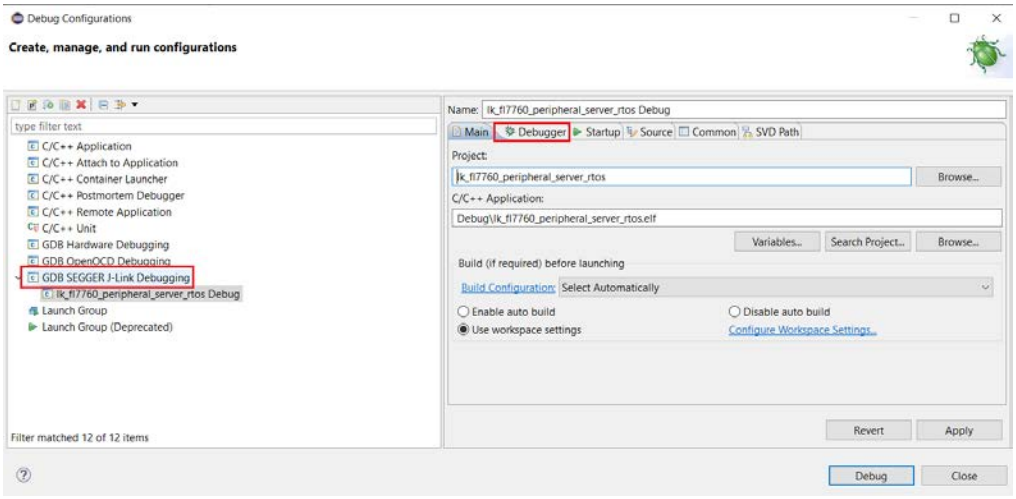
Flashing the application onto RSL10

Once the project is built user can flash the binaries into the MCU.

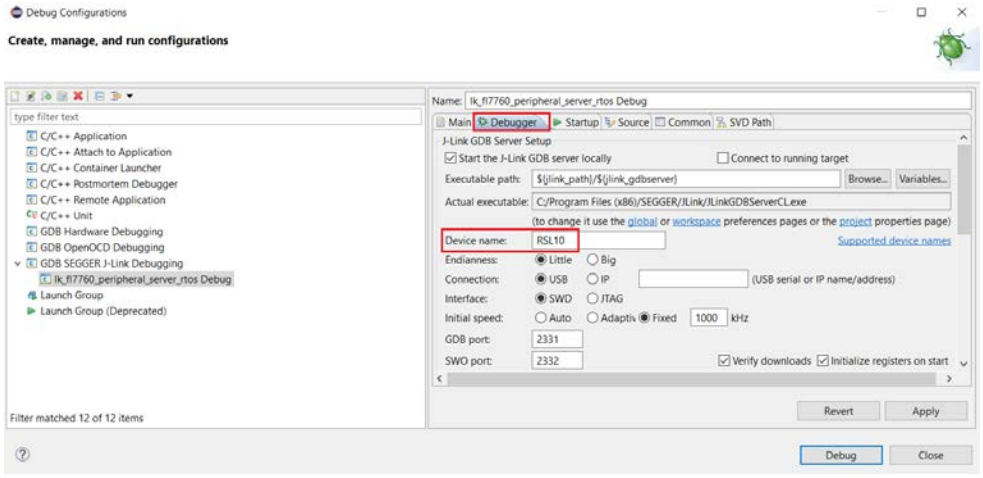
Open the Debug Configurations under the IDE



Double click onto GDB Segger J-Link Debugging. It automatically loads the built application.



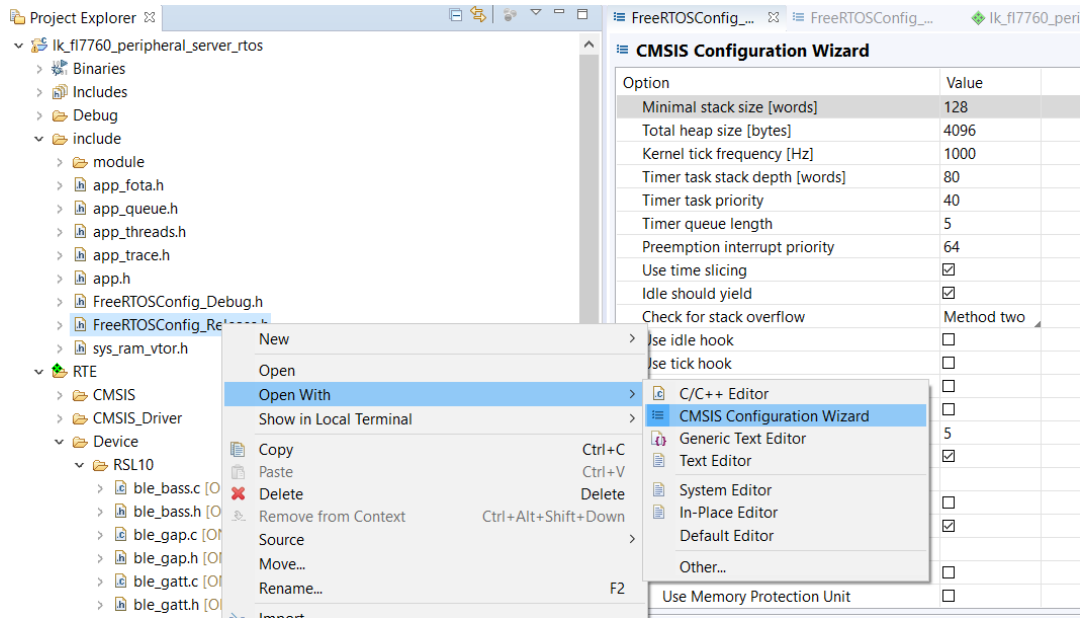
Under Debugger menu, don't forget to fill in RSL10 for Device Name. Click on Debug button.



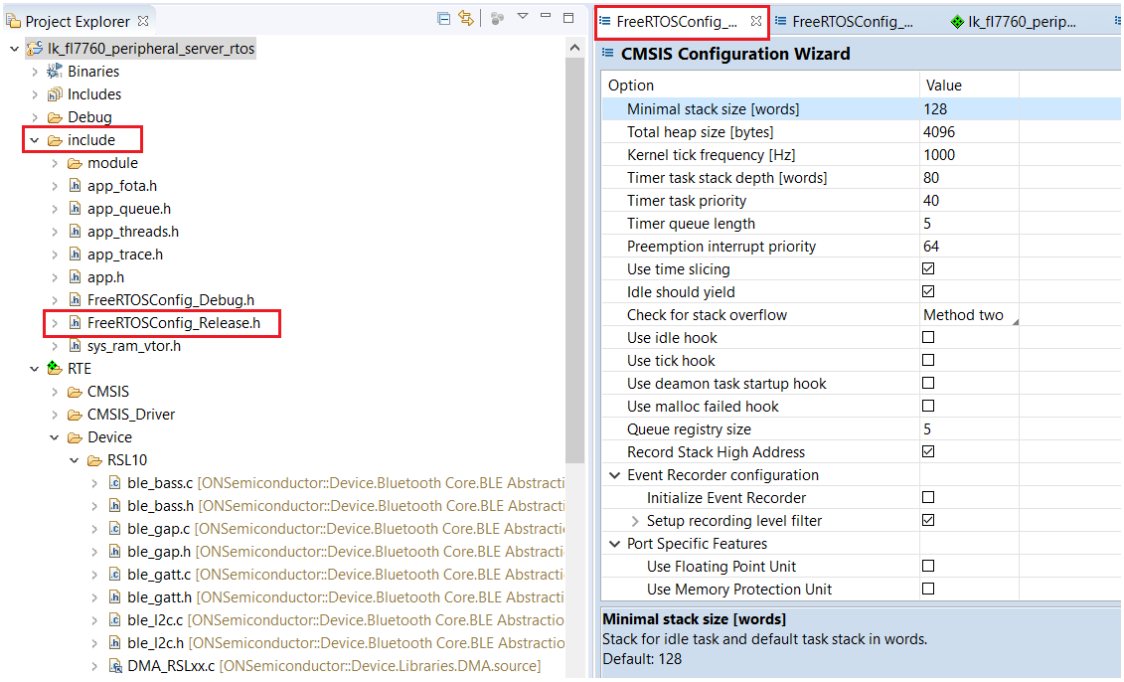
Configuration wizards

The configuration Wizards for Lighting kit allows user to set – up the parameters of the RTOS and application w/o debugging.

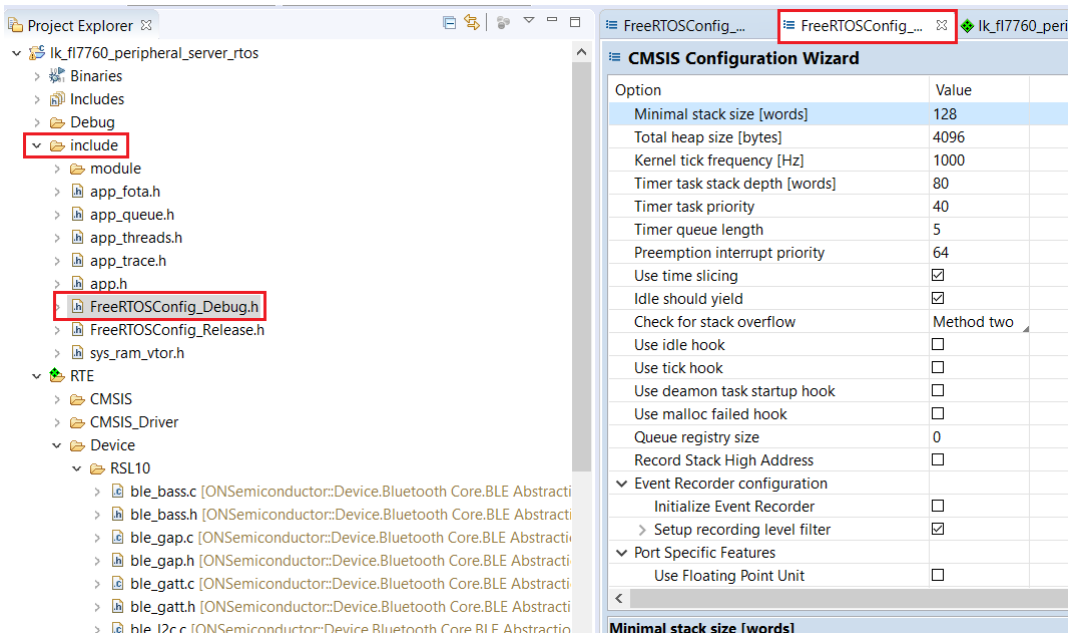
There are two options in place for configuring RTOS operation system within RSL10. Both are placed under *include* folder and can be accessed by right click -> Open With -> CMSIS Configuration Wizard



1. FreeRTOSConfig_Release.h – attached to the Release version while debugging



2. FreeRTOSConfig_Debug.h – attached to the Debug version of the code while debugging



User is allowed to set the parameters of the application w/o actual programming directly under RTE_Device.h

