EVBUM2623/D

RSL10 Solar Cell Multi-Sensor Platform User's Guide

ON Semiconductor®

www.onsemi.com

EVAL BOARD USER'S MANUAL

Introduction

The RSL10 Solar Cell Multi−Sensor Platform (RSL10−SOLARSENS−GEVK) is a comprehensive development platform for battery−free IoT applications for smart building, smart home, and Industry 4.0 verticals. Based on the industry's lowest power Bluetooth® Low Energy radio (RSL10), the board features sensors for environmental and motion sensing (BMA400−a smart 3−axis accelerometer, BME280− a smart environmental sensor, and the NCT203 wide−range digital temperature sensor).

The board also features a low weight, low profile 47 μF storage capacitor of; a programming and debug interface; and a connected solar cell.

Since the device harvests energy from a low current source, it is important to minimize leakage of the overall system during operation and standby. Along with other energy efficient devices, an ultra−low quiescent current LDO (NCP170) on the board significantly minimize leakage.

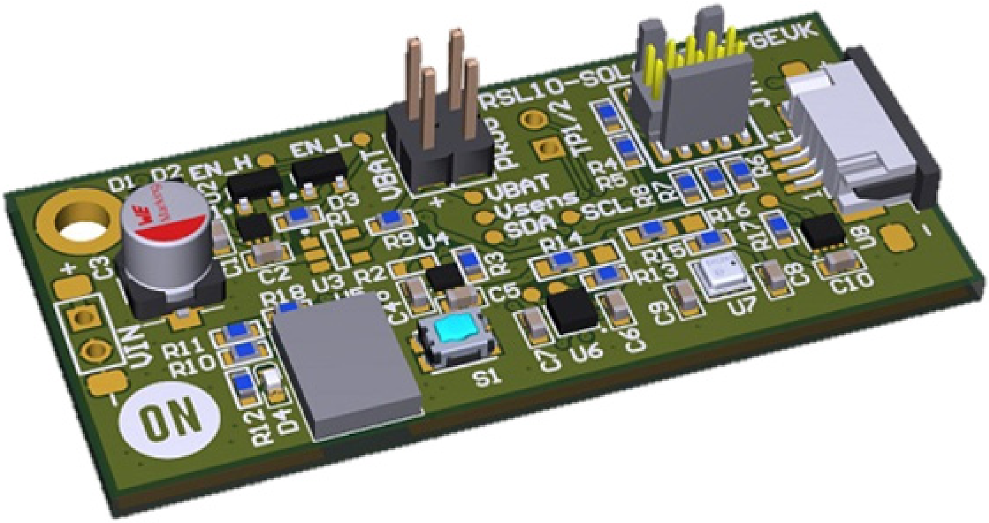


Figure 1. RSL10 Solar Cell Multi−Sensor Platform

Hardware Description

Default Configuration

The development platform includes a solar multi-sensor board hardware and a connected solar cell. If you need to reconnect the solar cell or would like to work with another solar cell than the one provided out of the box, follow the guidelines in section ‘Powering the Board’.

In addition to the RSL10 SIP (System−in−Package), the following sensors are present on the board.

- BMA400, 3−Axis Smart Accelerometer
- BME280, Environmental Sensor (temperature, humidity, pressure)
- NCT203, Wide−range (−40 to 125°C) Temperature Sensor

The platform also features an ultra−low quiescent LDO (NCP170) and a 100 μF capacitor to store energy.

Powering the Board

The board is powered by a solar cell. The default solar cell used is Panasonic Amorton AM−1522, which has a typical operating voltage of 3 V.

The circuitry is protected by a clamp at 3 V, and the operating domain is 1.6 V to 2.65 V.

Below 1.6 V, no transmission is allowed and the device is harvesting energy; Above 2.65 V, the device starts to operate and depletes energy buffering down to 1.6 V.

For more information about the power regulation section, refer to Continuous Harvesters and ON Semiconductor’s Low−Power RF Technology Close the Gap in Environmental and Accelerometer Sensors for IoT (TND6285/D). The powering cell or its equivalent can be mounted either by soldering both terminals or with the ZIF interface.

WARNING: ENSURE THE POLARITY OF THE PCB IS CORRECT WITH RESPECT TO THE ONE OF THE CELL.

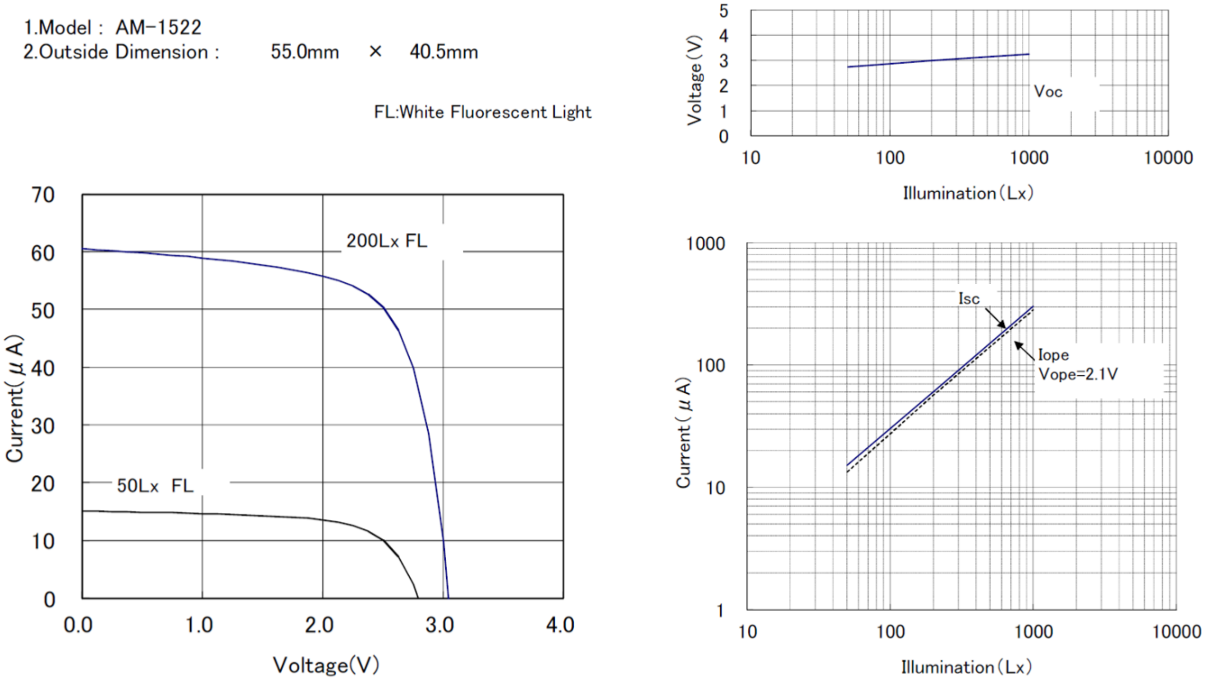


Figure 2. Electrical Specifications of the Panasonic Amorton AM−1522 Solar Cell

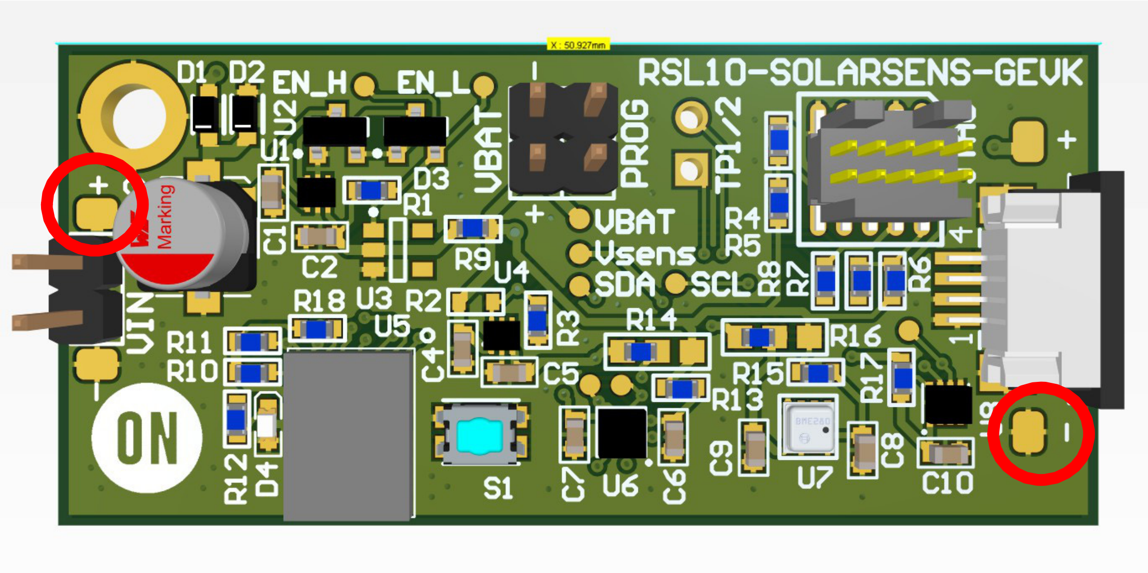


Figure 3. Connecting Cables to the Solder Pads

Normal Operations

Every transmission is signed with a LED pulse (**LED D4** located just at the Left of RSL10).

In case no transmission is seen or if the device looks to be hooked in out−of−operation, hit the **Reset** button (S1) and wait for a few seconds.

In normal lighting conditions, the LED will blink faster than once per second.

Operating Conditions

The device has been tested under the following lighting conditions:

Table 1. COMMON LIGHTING OPERATING CONDITIONS

	Light Source	**Time**	**Solar Cell Facing**	**Sensor Location**	**Lux Level** (Note 1)
Cloudy Winter	Natural	11:00am	Sky	Office, Near Window	415
“	“	“	Indoor	Office, Near Window	230
“	“	“	Outdoor	Office, Near Window	630
“	“	3:40pm	Indoor	Office desk	200
“	Ceiling Neon	11:00am	Ceiling	Office Corridor	340
“	“	“	White wall	Office Corridor	220
“	“	“	Ground	Office Corridor	140 (Note 2)
“	“	4:30pm	Ceiling	Office desk	250
“	Natural	9:00am	Window	Automotive Dashboard	700
“	“	“	Ground	“	350
“	“	“	Front seat	“	400

1. Lux levels are measure with uncalibrated iOS® App from Velux on iPhone® 6.
2. Under similar lighting conditions, the device should automatically start up and begin transmitting sensor data. For more information on sensor operations, refer to section ‘Firmware Implementation’.

Firmware Implementation

Default Configuration

The development platform is loaded with a preconfigured operating setting where both on-board temperature sensors are polled once at a time alternatively, and temperature information Ioss is sent via a default Eddystone beacon format.

Using the RSL10 Sensor Beacon App

The RSL10 Sensor Beacon application enables monitoring of the telemetry data of various RSL10–based platforms and it is available on iOS and Google Play. When connected to the RSL10 Solar Cell Multi–Sensor Platform, the app allows you to view the following telemetry and environment parameters:

- Temperature (via two different inputs)
- Pressure
- Humidity
- Acceleration in 3–axis
- Advertised packages count
- Time since up–time

Aside from displaying the real–time values of these variables, the app plots as well the values over time in several charts. All the logged data is exportable in csv format.

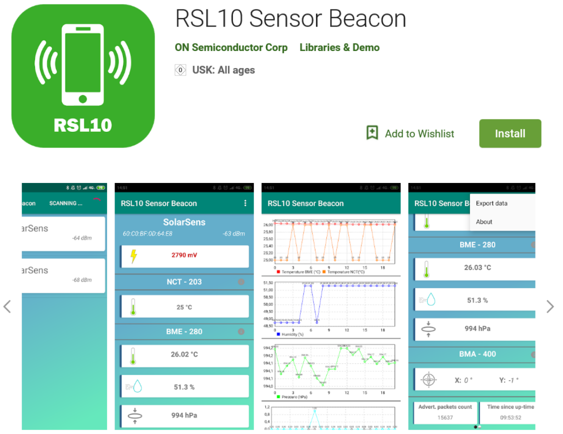


Figure 4. RSL10 Sensor Beacon Mobile App

Other freely–available beacon scanning apps such as BLE Scanner can also connect with RSL10 Solar Cell Multi–Sensor Platform to display the received packages, but will not support all the functionalities of RSL10 Sensor Beacon.

Customizing the Firmware

In order to customize the firmware, ensure you have downloaded the following from onsemi.com:

- RSL10 Software Development Kit (SDK), an Eclipse–based environment for software development for all RSL10–based platforms
- Bluetooth IoT Development Kit CMSIS Pack

The RSL10 SDK contains fully integrated development environment with a powerful editor, toolchain, documentation, a wide range of example code, and a CMSIS–Pack based software packages. For more information on the RSL10 SDK, refer to the RSL10 SDK Getting Started Guide (AND9697).

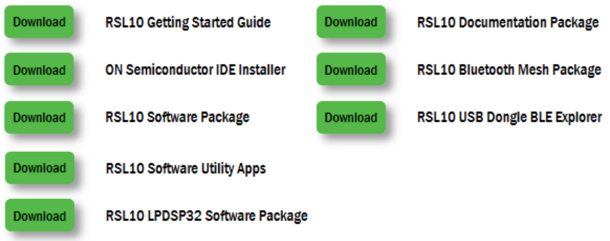
Installing the RSL10 Software Development Kit and CMSIS–Packs

The B–IDK software allows for rapid development of various use cases. For This section details the prerequisites and provides detailed steps for downloading the firmware onto the RSL10–SOLARSENS–GEVK.

Prerequisities

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



Download RSL10 Getting Started Guide

Download ON Semiconductor IDE Installer

Download RSL10 Software Package

Download RSL10 Software Utility Apps

Download RSL10 LPDSP32 Software Package

Download RSL10 Documentation Package

Download RSL10 Bluetooth Mesh Package

Download RSL10 USB Dongle BLE Explorer

Figure 5.

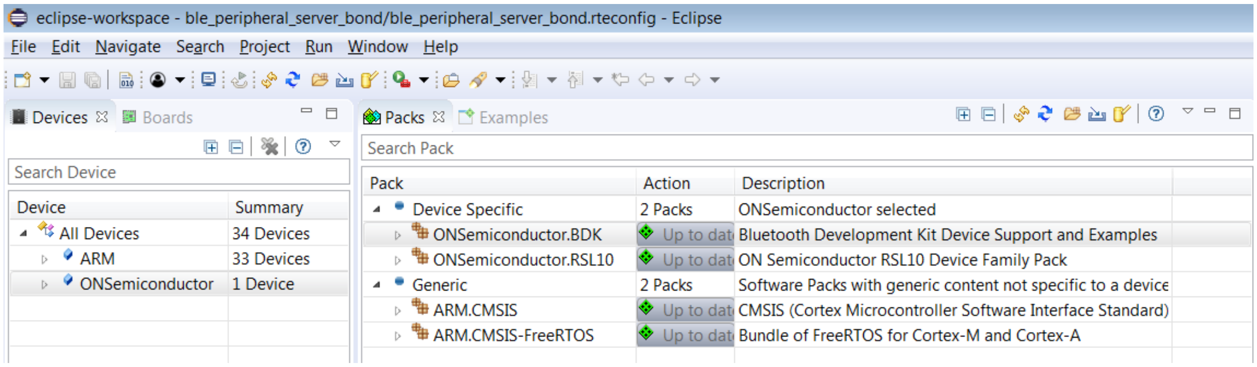
4. Download the B–IDK CMSIS pack from https://www.onsemi.com/B–IDK and save it in the same folder as the RSL10 CMSIS pack (see 3.a above)
5. CMSIS pack at item 4. is dependent on ARM CMSIS pack as well. Please install ARM CMSIS pack 5.5.1 or higher after download from: https://github.com/ARM–software/CMSIS_5/releases
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

1. Launch the RSL10 ON Semiconductor IDE

NOTE: Please import the RSL10 CMSIS pack first as the B−IDK CMSIS pack (step 4 in the Prerequisites section) depends on the RSL10 CMSIS pack (step 3a in the Prerequisites section).

2. Refer to Chapter 3 of RSL10 SDK Getting Started Guide (step 3a) for step−by−step instruction on importing the CMSIS packs.
3. Once all packs are successfully imported, they can be viewed in the CMSIS pack manager perspective as shown below.



eclipse-workspace - ble_peripheral_server_bond/ble_peripheral_server_bond.rteconfig - Eclipse

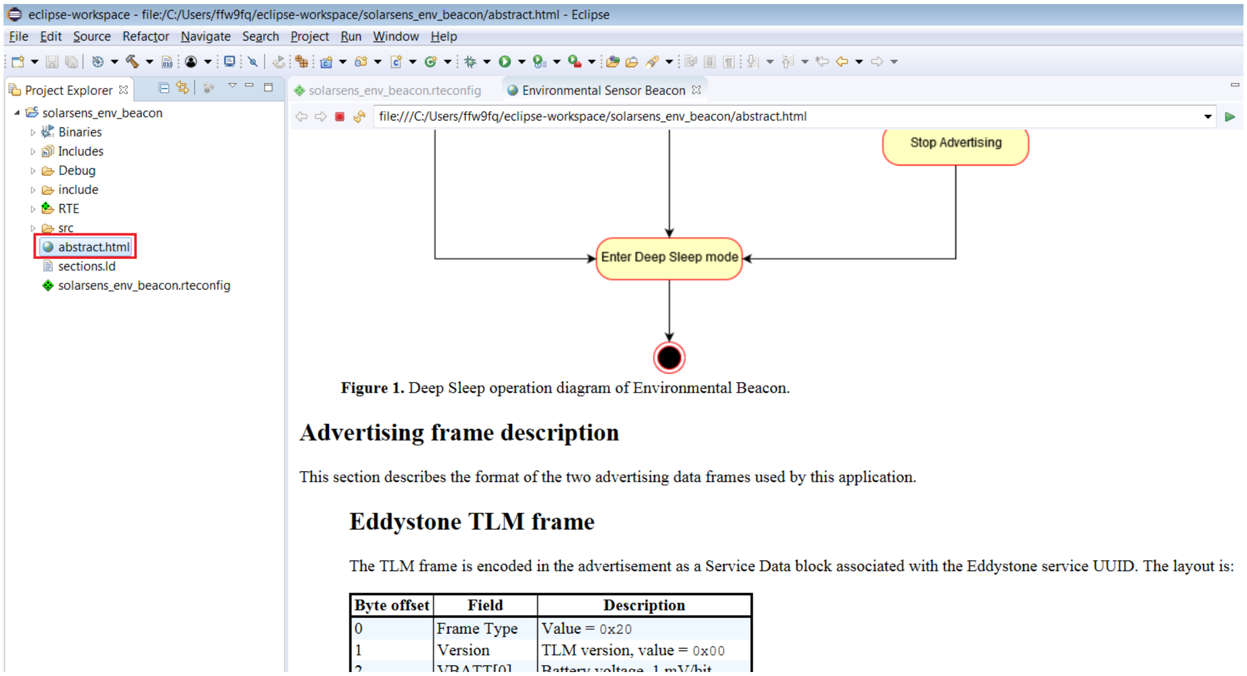
Search Device

Device	Summary
All Devices	34 Devices
ARM	33 Devices
ONSemiconductor	1 Device

Search Pack

Pack	Action	Description
Device Specific	2 Packs	ONSemiconductor selected
ONSemiconductor.BDK	Up to dat	Bluetooth Development Kit Device Support and Examples
ONSemiconductor.RSL10	Up to dat	ON Semiconductor RSL10 Device Family Pack
Generic	2 Packs	Software Packs with generic content not specific to a device
ARM.CMSIS	Up to dat	CMSIS (Cortex Microcontroller Software Interface Standard)
ARM.CMSIS-FreeRTOS	Up to dat	Bundle of FreeRTOS for Cortex-M and Cortex-A

Figure 6. CMSIS Pack Manager Perspective



eclipse-workspace - file:/C:/Users/ffw9fq/eclipse-workspace/solarsens_env_beacon/abstract.html - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
solarsens_env_beacon
Binaries
Includes
Debug
include
RTE
src
abstract.html
sections.ld
solarsens_env_beacon.rteconfig
solarsens_env_beacon.rteconfig
Environmental Sensor Beacon
file:///C:/Users/ffw9fq/eclipse-workspace/solarsens_env_beacon/abstract.html
Stop Advertising
Enter Deep Sleep mode

Figure 1. Deep Sleep operation diagram of Environmental Beacon.

Advertising frame description

This section describes the format of the two advertising data frames used by this application.

Eddystone TLM frame

The TLM frame is encoded in the advertisement as a Service Data block associated with the Eddystone service UUID. The layout is:

Byte offset	Field	Description
0	Frame Type	Value = 0x20
1	Version	TLM version, value = 0x00
2	VBATT[0]	Battery voltage, 1 mV/bit

Figure 7. Advertising Frame Description

Since the board does not provide a debugging probe, a compatible standalone debugging probe is required. This can be any SEGGER J−Jink debug probe with 10−pin Cortex® Debug connector adapter.

The board comes with a Ribes Tech solar cell attached. **For the purposes of debugging and re−flashing of the board**, it is required to disconnect the solar cell from the board. External power source needs to be connected to either of the VIN connectors. Power source can be alternatively connected to VBAT header for the purpose of detailed power consumption analysis of RSL10 and associated sensors.

Example setup for debugging is shown in Figure 9 where the board is powered by 3.3 V from an UART to USB convertor and also uses the `PROG` header as UART TX line. For power measurements the converter should be replaced by appropriate power consumption meter and both debug probe and UART should be disconnected.

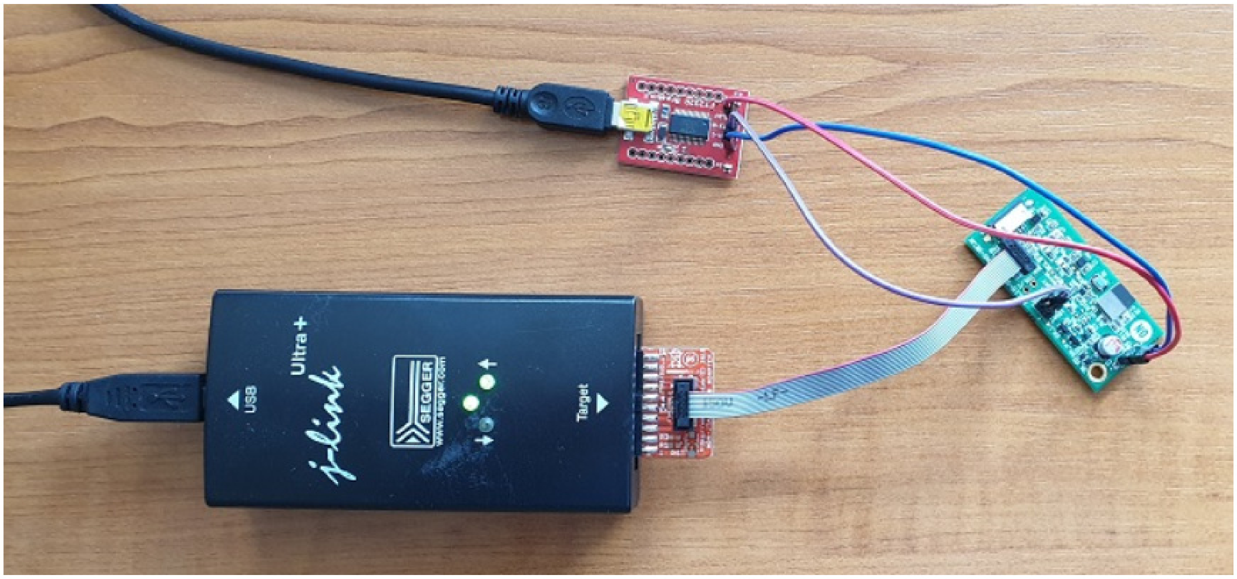


Figure 8. Sample Debugging Set−up

Build Configurations

The project provides two build configurations that can be selected and build using the **Build Selector** in the Toolbar.

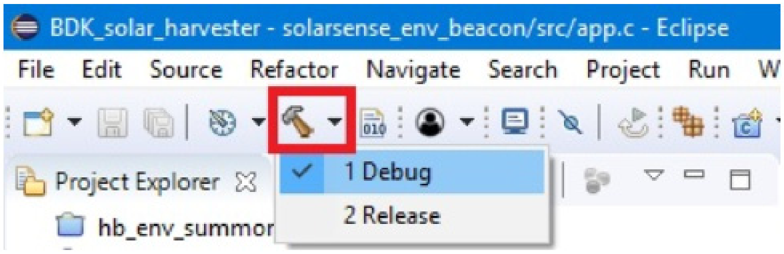


Figure 9. Build Configuration Selection Button in the RSL10 SDK

Debug

This configuration should be used for debugging purposes only.

- Binary with debugging symbols (`-O0 -g3`).
- Trace messages printed over UART peripheral using PROG DIO pad (Header J8, DIO12). Configuration for this port is: `230400 bps, 8N1, no flow control`

Sending of trace messages over UART slows down the execution of the program which might impact performance in some cases.

Release

This configuration should be used for power consumption measurements and production builds.

- Optimized for speed, no debug symbols (`-O2`).
- Trace messages disabled.

CMSIS Configuration Header

The project provides configuration header `app_config.h` located in the `include` folder of project. This header can be opened by using CMSIS Configuration wizard editor as shown in Figure 10. The Configuration Wizard allows some predefined program parameters to be changed without changing the code directly. All options provide short descriptions and check for valid setting value range.

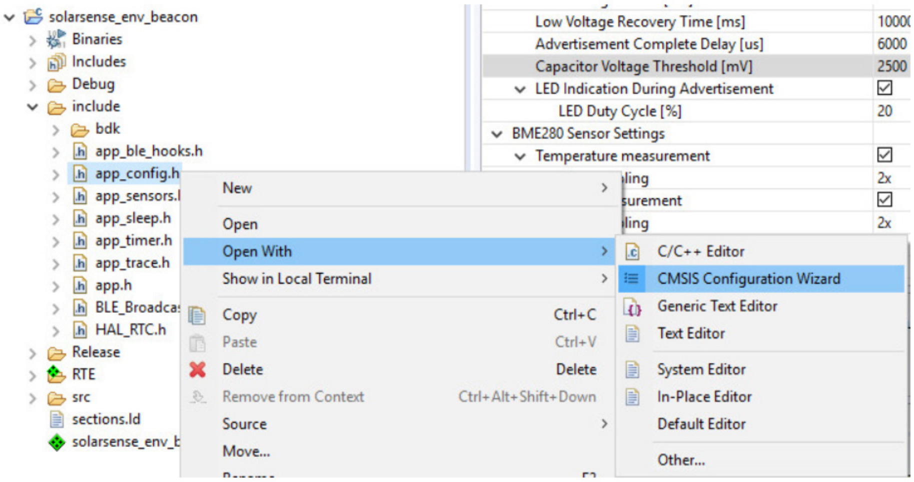
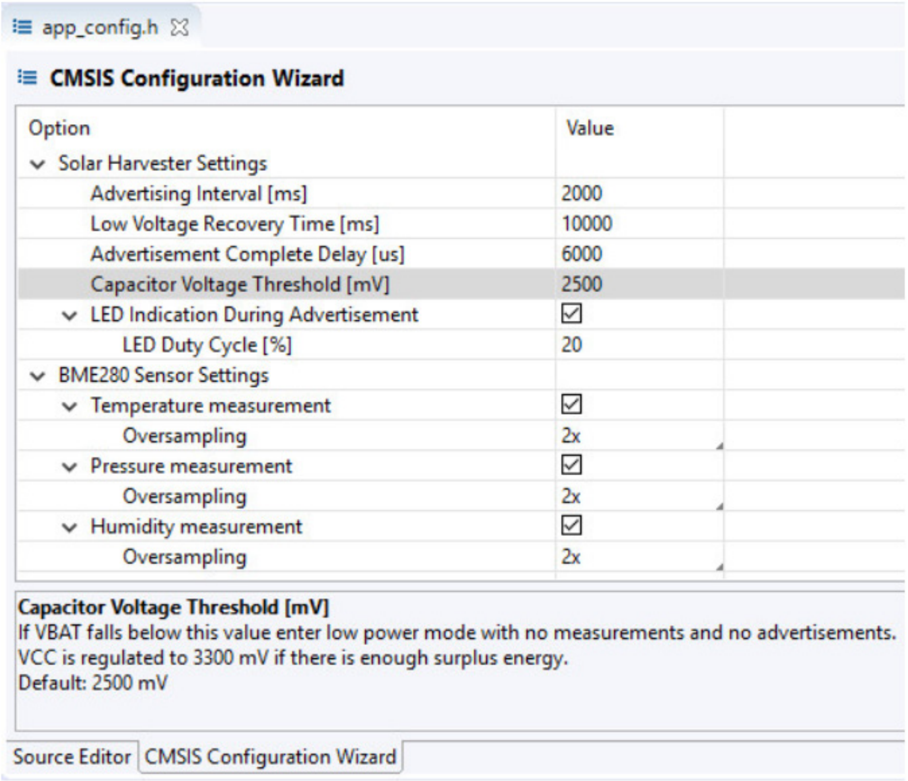


Figure 10. Selecting CMSIS Configuration Wizard as the Default Editor for *app_config.h* File



Option	Value
Solar Harvester Settings	
Advertising Interval [ms]	2000
Low Voltage Recovery Time [ms]	10000
Advertisement Complete Delay [us]	6000
Capacitor Voltage Threshold [mV]	2500
LED Indication During Advertisement	☑
LED Duty Cycle [%]	20
BME280 Sensor Settings	
Temperature measurement	☑
Oversampling	2x
Pressure measurement	☑
Oversampling	2x
Humidity measurement	☑
Oversampling	2x

Capacitor Voltage Threshold [mV]
If VBAT falls below this value enter low power mode with no measurements and no advertisements.
VCC is regulated to 3300 mV if there is enough surplus energy.
Default: 2500 mV

Figure 11. CMSIS Configuration Wizard with Available Program Settings

Figure 12 shows the current consumption of the board during a sensor measurement event, followed by advertisement of measured data. During this event, a total of 60 μJ of energy was used to both measure sensor data and advertise the results. If sensor measurement is not scheduled and the board only advertises, the energy consumption is reduced to 20 μJ.

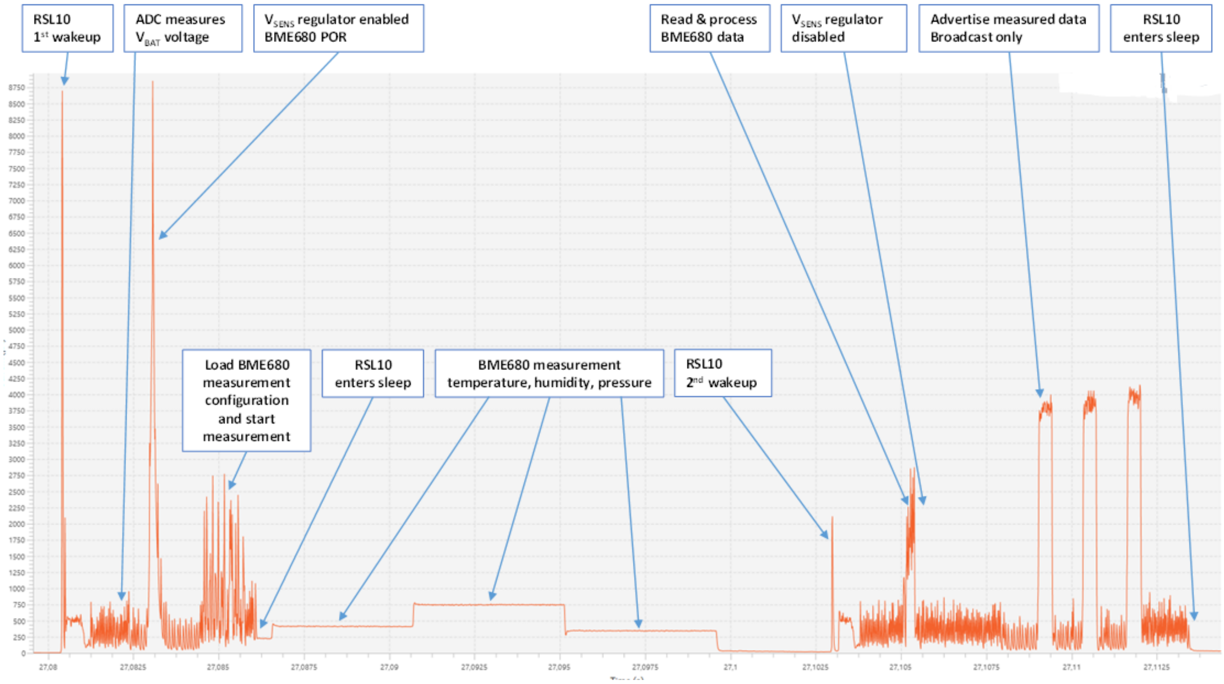


Figure 12. Typical Operation Cycle with Sensor Measurement and Advertising (3 V power supply, advertising interval set to 1 s, and sensor measurement during every advertising interval)

Debugging/Flashing

Refer to the RSL10 SDK Getting Started Guide Section 4.4 for instructions on how to create debugging configurations and flash the program onto RSL10.