

RSL10-SENSE-GEVK (and RSL10-SENSE-DB-GEVK) User Guide

Introduction

The RSL10-SENSE-GEVK (and RSL10-SENSE-DB-GEVK) is a comprehensive, compact, node-to-cloud IoT sensor platform that allows development of various Bluetooth Low Energy based use cases. Along with the hardware and software, the RSL10-SENSE-GEVK includes a mobile app to interact with sensors and actuators. The board features RSL10, Industry's lowest power Bluetooth® 5 SoC and several sensors from ON Semiconductor and Bosch. By combining motion, environmental, ambient light sensing with the ultra-low power of the Bluetooth 5 Certified RSL10 and will enable customers to realize a new class of battery powered static, mobile and wearable smart sensors targeting consumer and industrial applications in the IoT.

The overall deep sleep consumption of 20 μ A results in a battery life of over 1 year. For further increase in battery life, software configuration wizard allows flexible timing setup as discussed in the following sections.

Variants

There are two SKUs of the RSL10 Sensor kit. Both variants are pre-loaded with an ultra-low power firmware and include a 3 V CR2032 coin cell and a flexible NFC antenna.

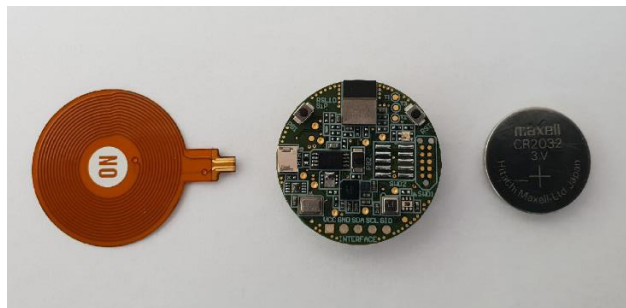


Figure 1.

RSL10-SENSE-GEVK: Firmware can be flashed via 10-pin needle adapter (e.g. TC2050 from Tag-Connect) (not included). The 10-pin header for debugger is not populated on the board.

RSL10-SENSE-DB-GEVK: The “debug” (-DB) version of the board also includes a low cost Segger debugger J-Link LITE CortexM and a USB cable. Users can directly debug/communicate/flash the firmware over the populated-pin header.



ON Semiconductor®

www.onsemi.com

EVAL BOARD USER'S MANUAL

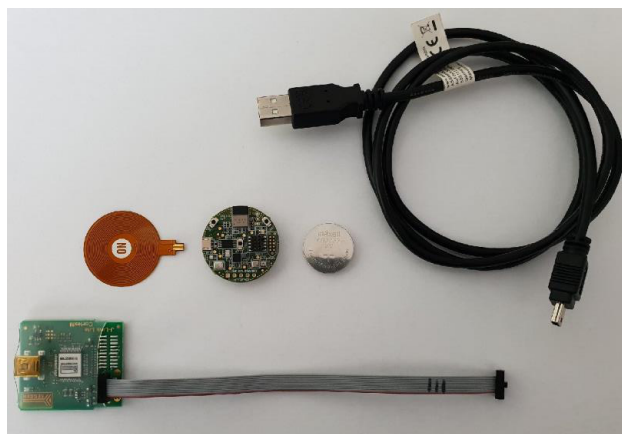


Figure 2.

Scope

The board starts functioning as soon as the coin cell is in place. This document covers the setup, software architecture, documentation and provides instructions on downloading firmware to the board. The details regarding the mobile app and cloud connectivity are not covered in this document.

Default Configuration

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

- NOA1305, ambient Light sensor
- N24RF64, NFC EEPROM
- BME680, environmental sensor (temperature, humidity, pressure, air quality)
- BHI160 + BMM150, 3-axis accelerometer, gyroscope, magnetometer. Together return absolute orientation supported in software
- INMP522 -> ultra-low power microphone for audio applications
- User can insert NFC flexible antenna into dedicated connector and bend underneath the battery holder for custom packaging / cases.

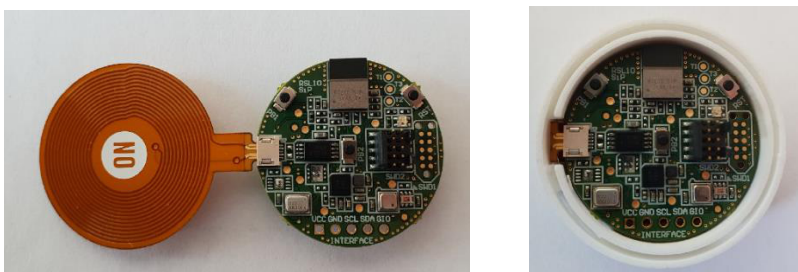


Figure 3.

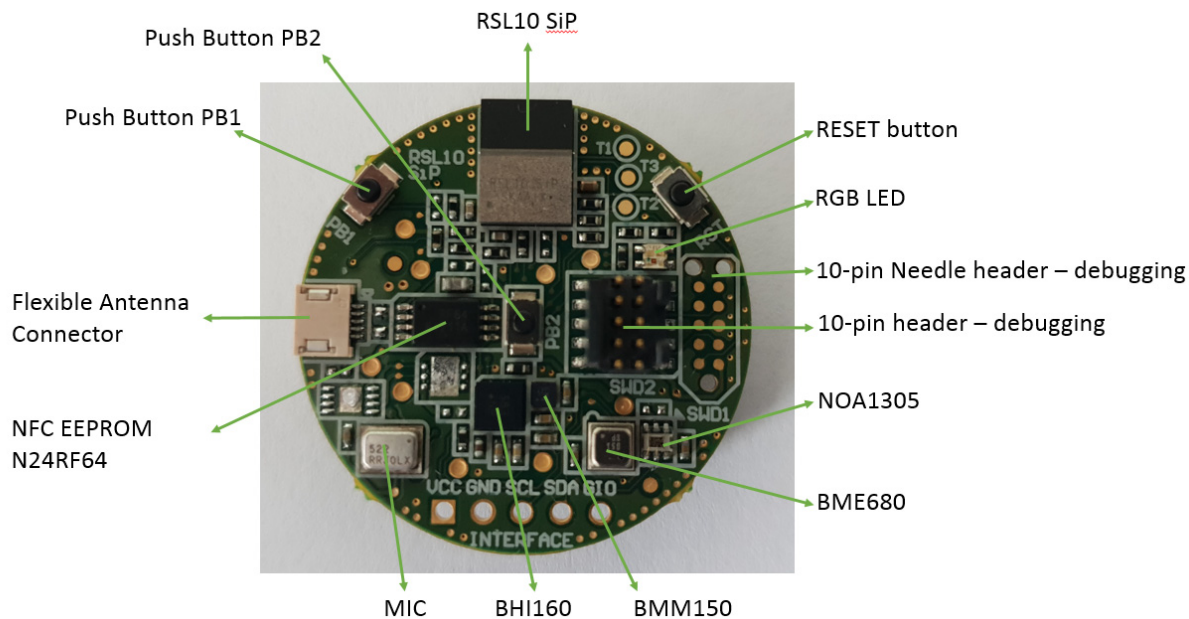


Figure 4.

Both kits are shipped with the ultra-low power firmware pre-loaded into the boards.

Powering the Board

To power RSL10-SENSE-GEVK, one has to insert CR2032 (3 V) battery into battery holder located on the bottom side of the board.

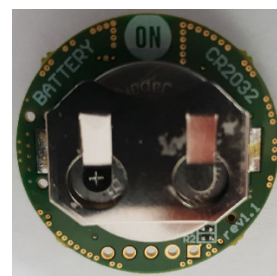


Figure 5.

SOFTWARE

The RSL10-SENSE-GEVK boards are, by default, configured with the ultra-low power firmware. For users that want to download different firmware versions, this section details the involved steps.

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



Figure 6.

4. Download the B-ID 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>

The next section provides details on importing the downloaded CMSIS packs into the SDK.

Importing CMSIS Packages

1. Launch the RSL10 ON Semiconductor IDE

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

2. Refer to Chapter 3 of RSL10 SDK Getting Started Guide (step 3.a) for step-by-step instructions 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 (Figure 7)

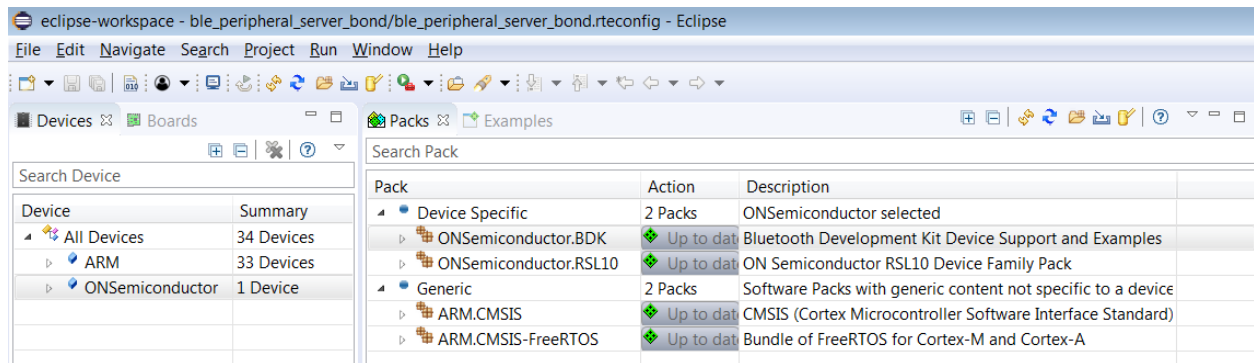


Figure 7.

Compiling and Flashing of Ultra Low Power Firmware

4. Examples related to RSL10-SENSE-GEVK are highlighted in brackets. Choose the example *Custom Service Firmware with Deep Sleep (RSL10-SENSE-GEVK)*

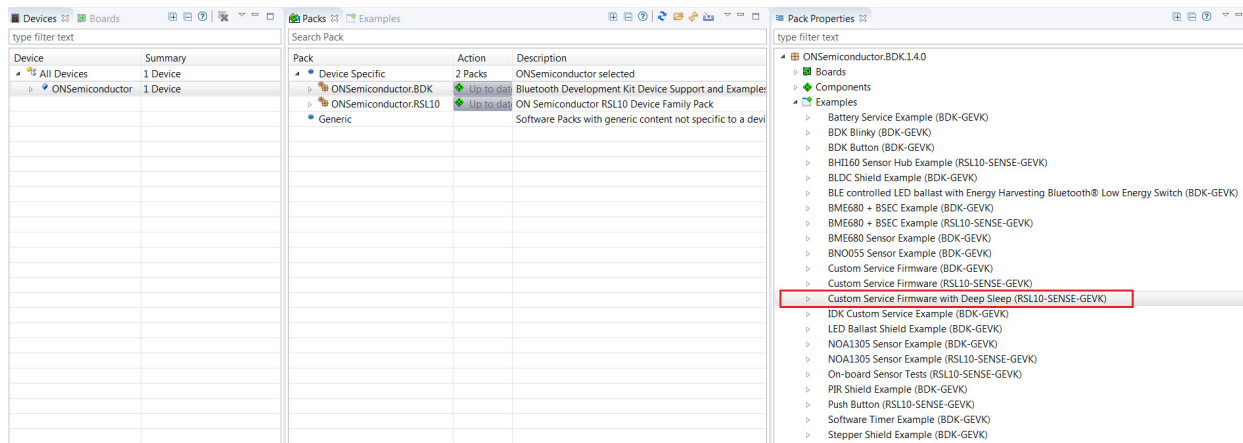


Figure 8.

5. Right click and copy the project into workspace

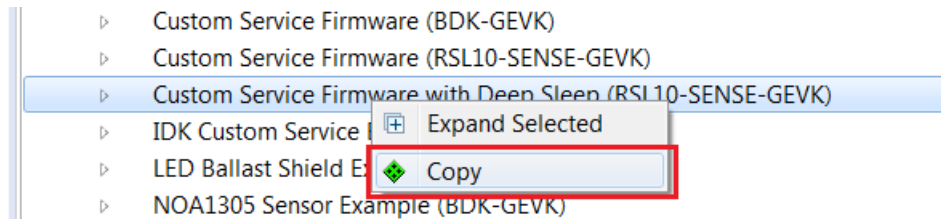


Figure 9.

NOTE: Once the example is copied, it can be viewed under Project Explorer. All source files including main are located in the src folder.

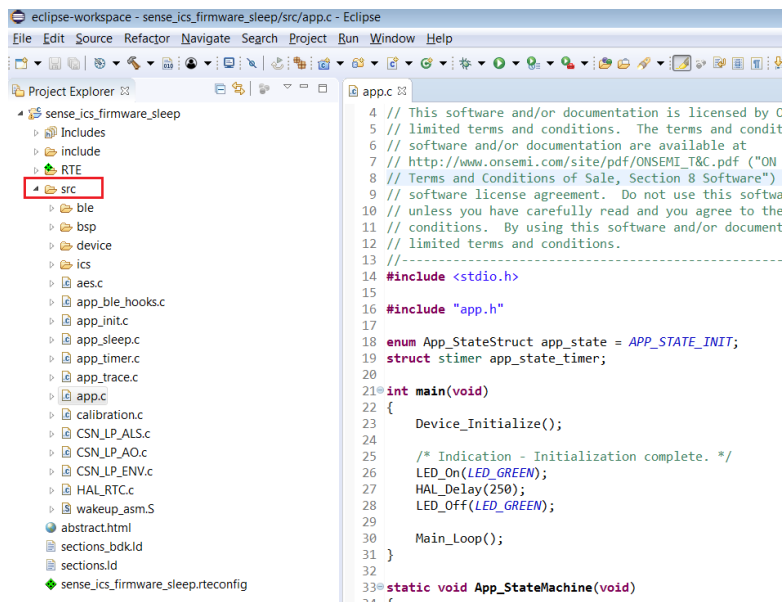


Figure 10.

6. Now user has to build the project as this creates binaries to be flashed to RSL10-SENSE-GEVK. For the sensor board, there are two options:
 - a.) 1 Debug
 - b.) 2 Release – go to hammer icon inside IDE and click Release. Project is automatically build

Debug mode enables user to debug application over serial terminal connected to GPIO pin on expansion connector. It's the option how to fine tune the sleep mode code. RSL10-SENSE-GEVK natively doesn't support serial communication, only RTT over JTAG.

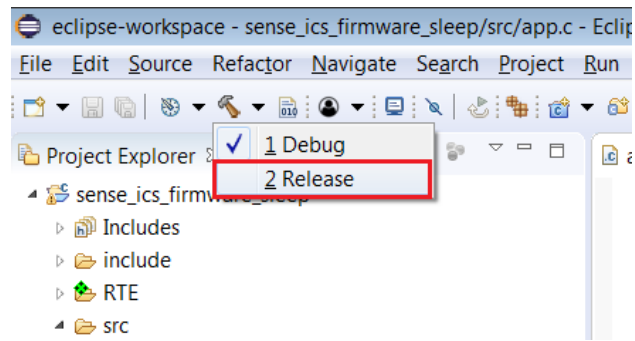


Figure 11.

NOTE: If the binaries are not seen, press F5 (refresh)

Alternatively you can build the project: right click on project under Project Explorer -> Build Configurations -> Set Active -> 2 Release

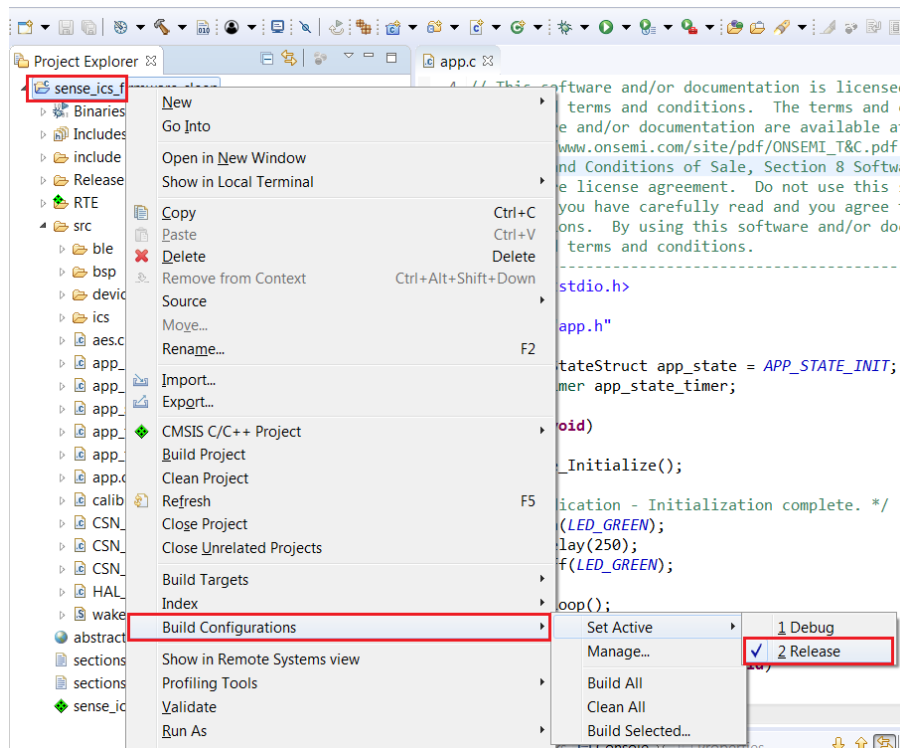


Figure 12.

7. Once the build is done, the code is ready to be flashed to the RSL10-SENSE-GEVK.
 - a.) Insert the battery into the board. Mandatory step as it creates the voltage reference for SWD logic signals.
 - b.) Connect the low cost Debugger (RSL10-SENSE-DB-GEVK version) / 10-pin needle adapter with J-LINK (RSL10-SENSE-GEVK version)



Figure 13.

8. Select the project (sense_ics_firmware_sleep), and go to debug configurations as shown below.

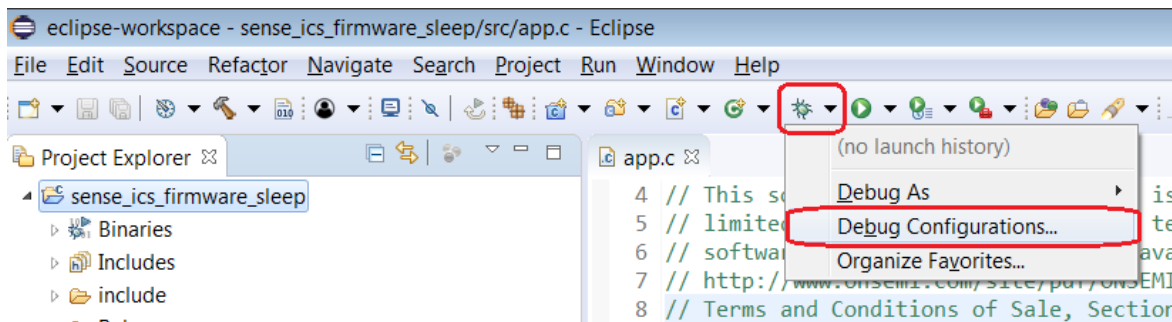


Figure 14.

- a.) Double click GDB Segger J-Link Debugging to create the debug configuration for the selected example.

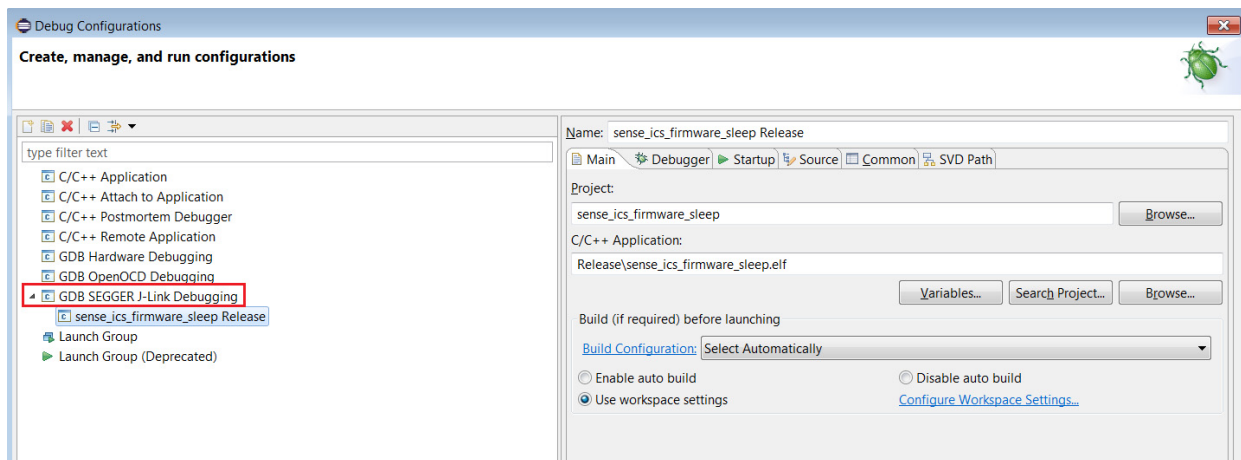


Figure 15.

NOTE: The debug configuration for the selected example is automatically saved and there's no need to re-create it. Make sure you have the Release version of binary (.elf). Click on Search Project and Qualifier returns *Release* in the path. For debugging purposes you can build and switch Debug version as discussed in step 12 a/b.

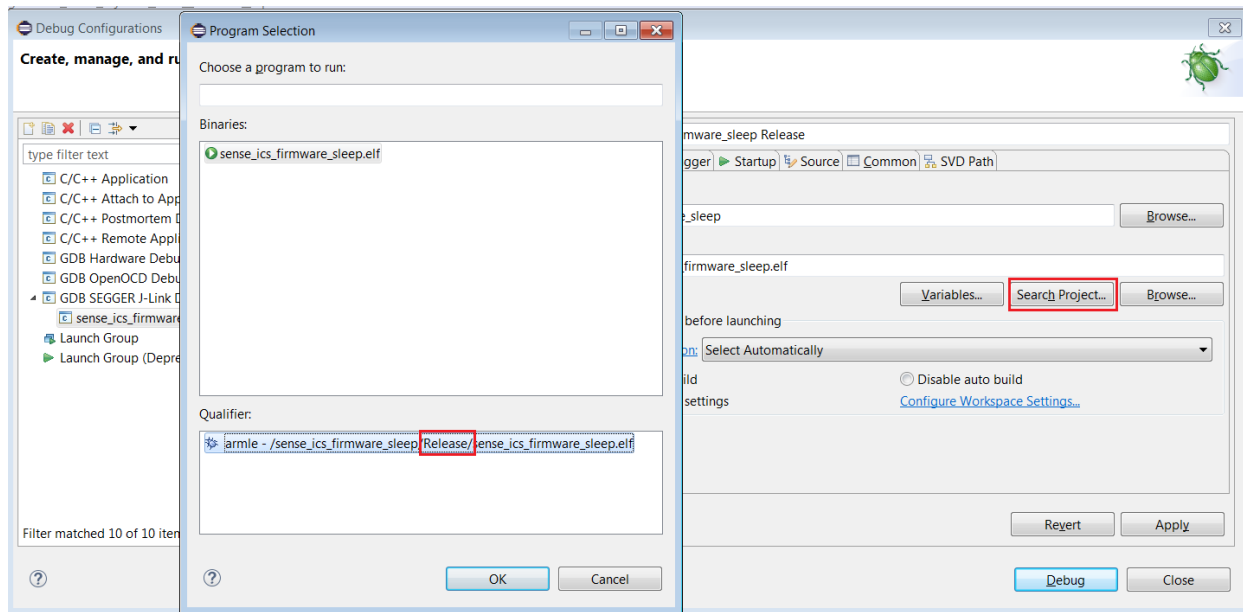


Figure 16.

- b.) On the Debugger tab, set RSL10 as the device name.
Click Debug.

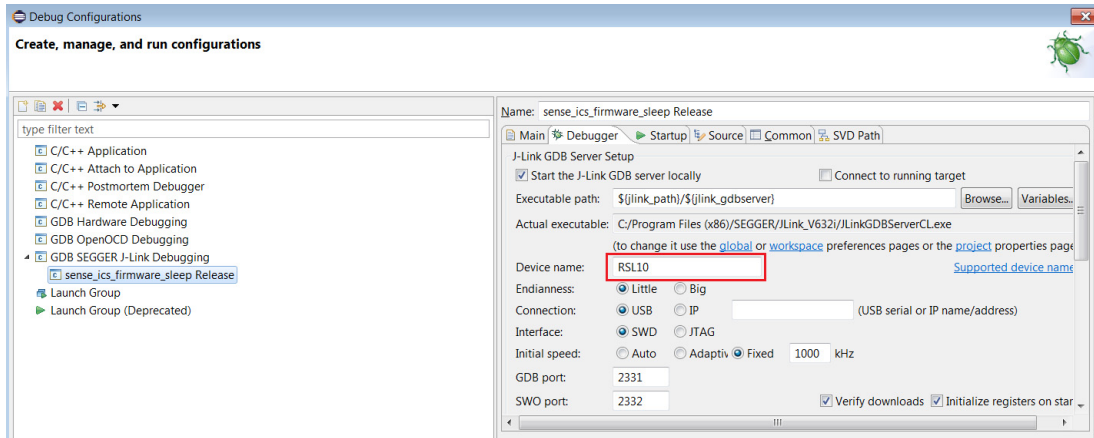


Figure 17.

9. For application debugging, confirm perspective switch by clicking Yes.

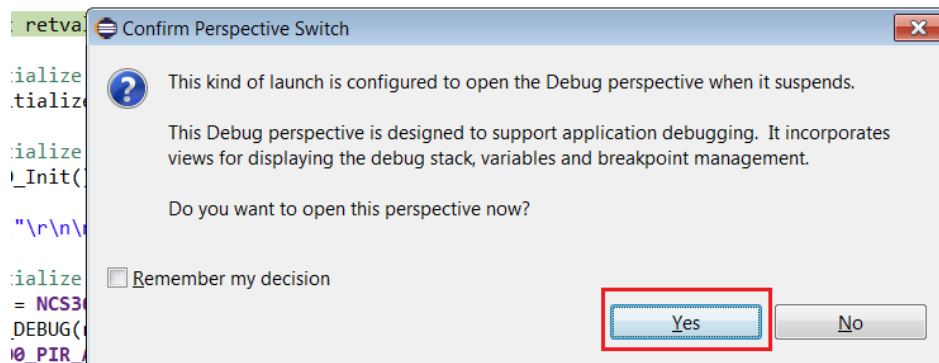


Figure 18.

10. The debug session is now launched. Click Resume (F8) to start the target CPU. Green LED briefly flashes. By default, in Release version is no Logging option and terminal doesn't return useful data. By terminating the session, user closes connection with DBG server.

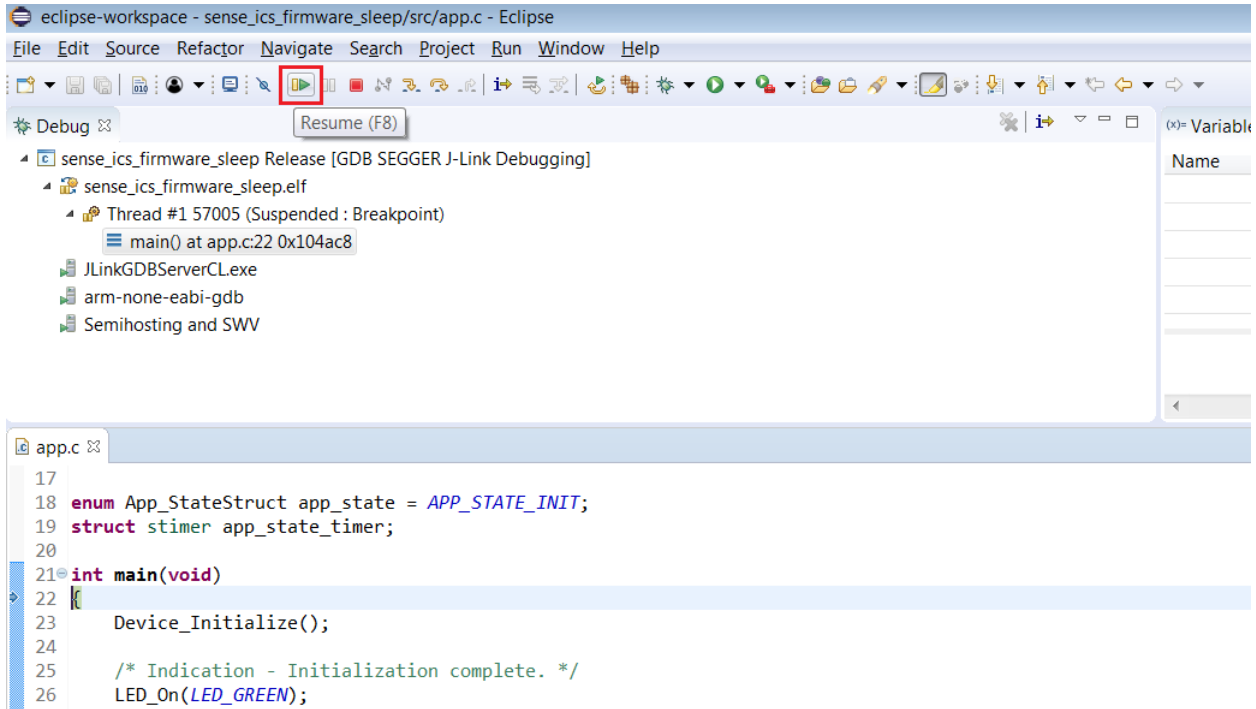


Figure 19.

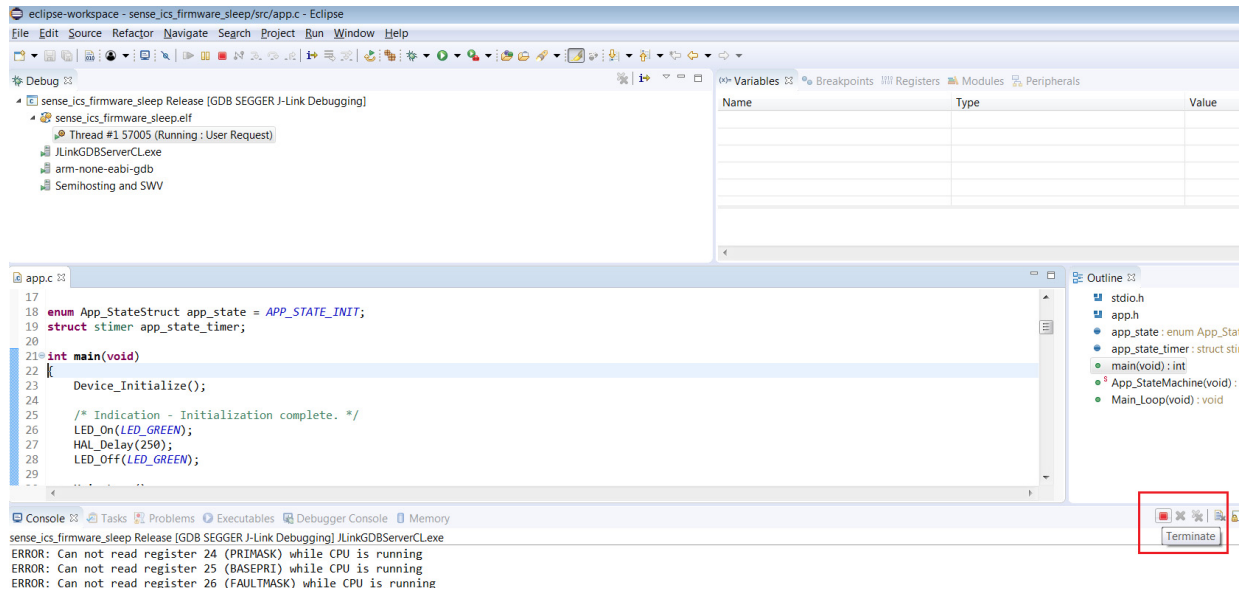


Figure 20.

11. Disconnect debugger, download and open the mobile app available under store (Android and iOS). App name is *RSL10 Sense and Control* (www.onsemi.com/b~idk)

IMPORTANT NOTE:

When the board is flashed, Green LED shortly blinks. Board starts BLE advertising only and is visible on the mobile app. When connection with mobile app is not made for next 60s (by default), blue LED blinks and RSL10-SENSE-GEVK goes into deep sleep mode. You can resume operation by holding button PB1 for >1s. Green LED blinks and process repeats. See below the state diagram.

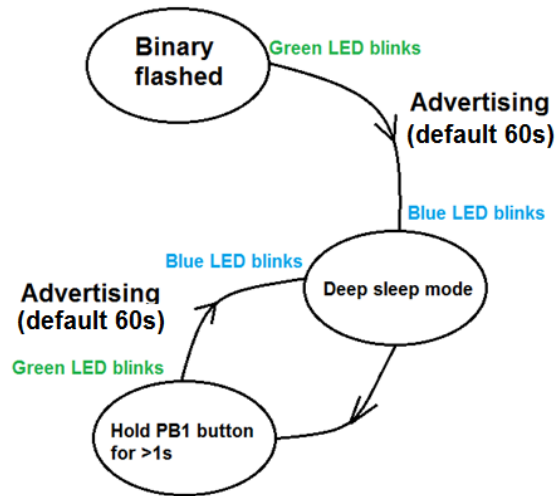


Figure 21.

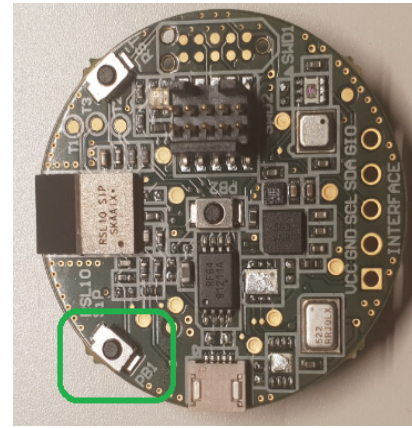


Figure 22.

12. User is exposed to set various parameters that have impact on battery longevity. Three main parameters can be configured in CMSIS:
- a.) BLE Advertising Interval (Default 1000 ms)
 - b.) Advertising stop Timeout (Default 60 s)
 - c.) Wake-up Button Check Interval (Default 1500 ms)

To get into the CMSIS Configuration Wizard, right click on RTE_app_config.h and open CMSIS Configuration Wizard. Change parameters, save the project and build it starting from step 12.

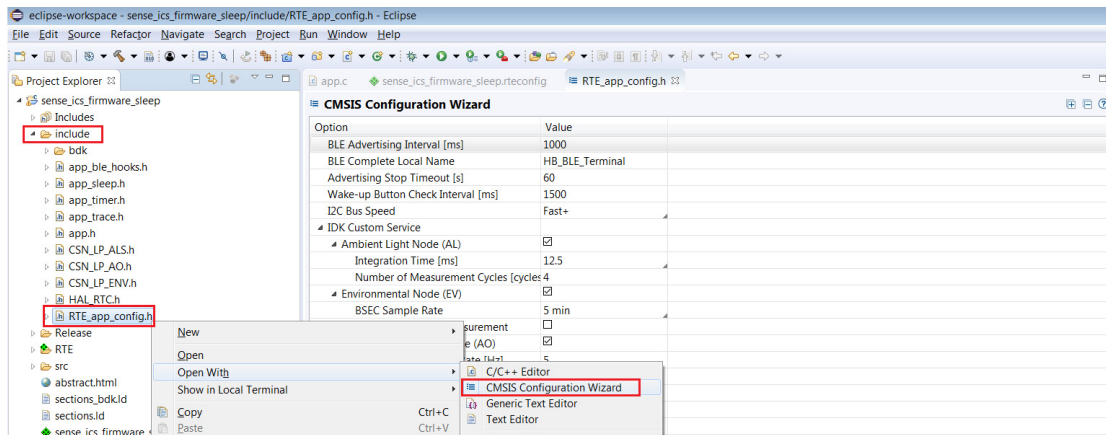


Figure 23.

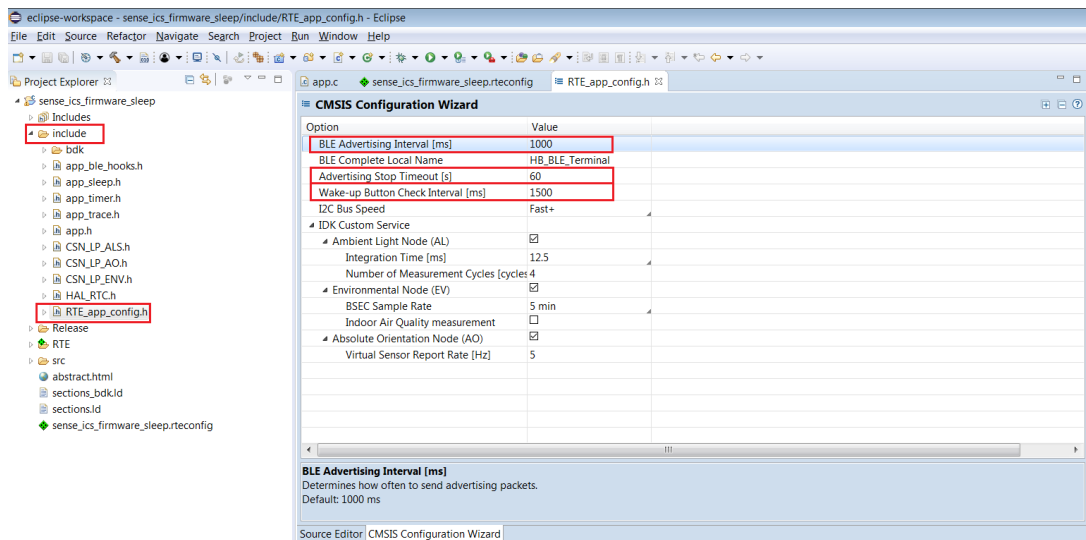


Figure 24.

Mobile App Usage

13. Within the *Advertising Stop Timeout* interval, board is visible on the app screen. RSL10–SENSE–GEVK is advertising only over BLE. When multiple boards (sensor nodes) are present, each has unique MAC address and user selects the desired one → HB_BLE_Terminal.

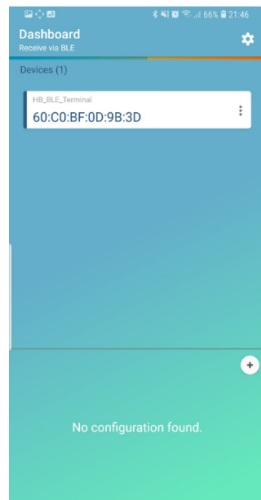


Figure 25.

14. When the appropriate board is selected, one can choose what sensor data to observe. Below are depicted all supported sensors and quantities taken. More simultaneous sensors in place equals more power required.

RSL10–SENSE–GEVK supports also cloud connectivity via the same mobile app that functions as a gateway.

NOTE: Air quality is not supported in this low power mode example due to heating element and consequent need for higher power consumption. However it's available under *BME680 + BSEC example* or *Custom Service Firmware* in CMSIS.

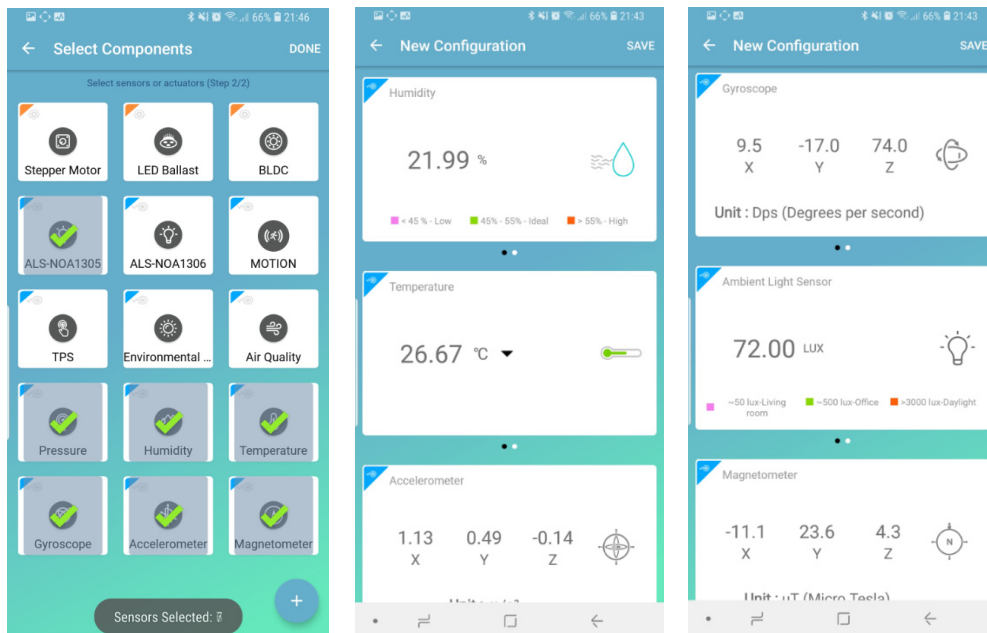


Figure 26.

Ultra-low Power Firmware Modes

15. The following are the low power features of above described firmware:

- BDK libraries adapted for use with deep sleep mode of RSL10.
- HAL library for RTC and RTC based low power timer.
- Low power IDK Custom Service nodes for:
 - a.) ALS (NOA1305)
 - b.) Environmental sensing (BME680 + BSEC software)
 - c.) Absolute Orientation (BHI160 + BMM150)
- Automatic on demand sensor activation.
- Automatically stops BLE advertising if no connection is made.
- BLE advertising can be restarted by holding push button (PB1).
- Configurable using RTE configuration header.

16. Environmental Sensing Node (BME680 + BSEC software):

- Provides two sample rates (every 3 seconds or every 5 minutes). By default, due to reduced power consumption, environmental sensors updates each 5 minutes.
- Option to disable IAQ measurement to save power.
- Long term average power consumption:
 - a.) Outputs: Indoor Air Quality, Compensated Temperature, Compensated Humidity, Pressure
 - i. Low Power mode – 900 μ A (3 s sample rate),
 - ii. Ultra-low power mode – 90 μ A (5 min sample rate)
 - b.) Outputs: Compensated Temperature, Compensated Humidity, Pressure
 - i. < 5.2 μ A (1 s sample rate)

- Gas sensor uses too much power and is not suitable for CR2032 battery powered systems. By default, this feature is disabled in ultra-low power firmware

17. Ambient Light Node (NOA1305):

- Sensor is activated only when ambient light value is requested by peer device.
- Power consumption depends on number of requests received from peer device.
 - a.) ~80 μ A current draw when sensor is active
- Sensor remains active for 4 measurement cycles (integration times) to stabilize sensor output.
- Integration time and number of cycles are configurable from RTE header.

18. BLE Connection Interval – Possible Power savings:

- BLE allows devices to negotiate connection parameter, most notably **Slave Connection Interval**
 - a.) BLE communication always occurs at every connection interval even if the devices do not have anything to exchange (just send empty packets).
 - b.) Configurable from 1.25 ms up to 4000 ms.
 - c.) Bigger Slave Connection Interval → Less energy consumed by HB.
 - d.) Master (phone) has complete control over used connection interval.
- Android allows only 3 specific interval configurations:
 - i. High (11.25 – 15 ms), Balanced (30 – 50 ms), Low Power (100 – 125 ms)
- All Android apps do not allow to set connection interval and always force balanced mode.

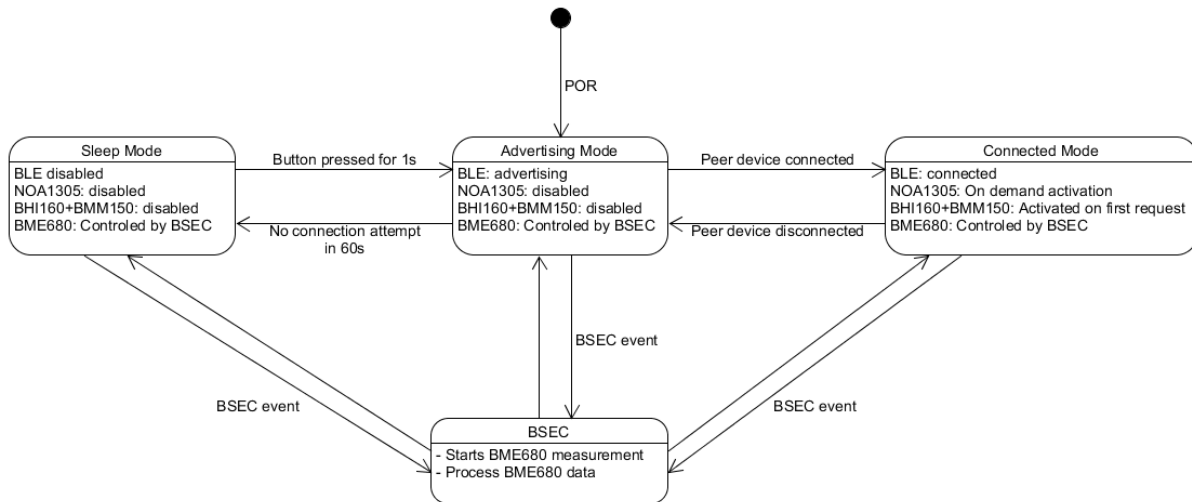


Figure 27.

19. Short term power consumption (100s interval):

- Deep Sleep mode:
 - a.) 1.5 s periodic button check
 - b.) Consumption: 18.7 uA @ 3 V
- Advertising mode:
 - c.) 1 Hz BLE advertising interval
 - d.) Consumption: 24.5 uA @ 3 V
- Connected mode:
 - e.) Full operation connected to RSL10 Sense & Control:
 - i. ALS: 1 s measurement & report rate
 - ii. Absolute Orientation: 12.5 Hz measure rate, 1 s report rate
 - iii. Environmental data (IAQ disabled): 5 min. measure rate, 3 s report rate
 - f.) Consumption: ~3000 uA @ 3 V

20. Low power firmware – block diagram. The diagram in Figure 27 depicts detailed high level operation of ultra-low power firmware.

Compiling and Flashing of the Rest Examples Attached to RSL10-SENSE

In this section user is guided on how to flash software for all remaining examples in the CMSIS pack. The procedure is similar as for Ultra-Low power FW. Let's pick up *On-board Sensor Tests* that enables microphone functionality and returns sensors data into console.

21. Right click and copy the project into workspace

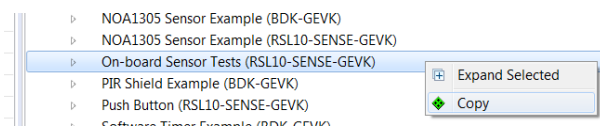


Figure 28.

22. Right click on the project and build it

NOTE: If the binaries are not seen, press F5 (refresh)

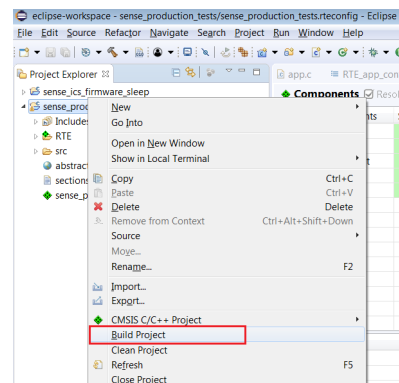


Figure 29.

23. Once the project is build, go to Debug configurations, double click on GDB SEGGER J-Link Debugging that automatically creates Session and import binaries ready to be flashed. Click on Debug button.

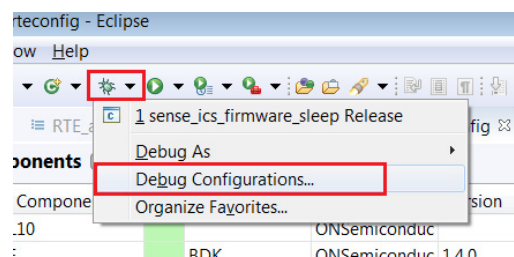


Figure 30.

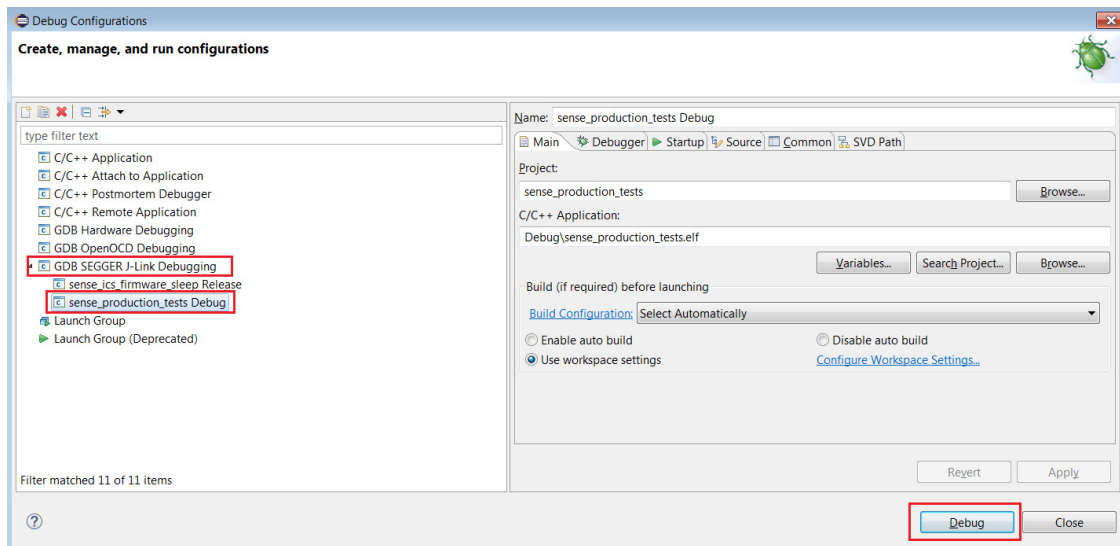


Figure 31.

24. When the debug session is launched, click on Resume button (F8).

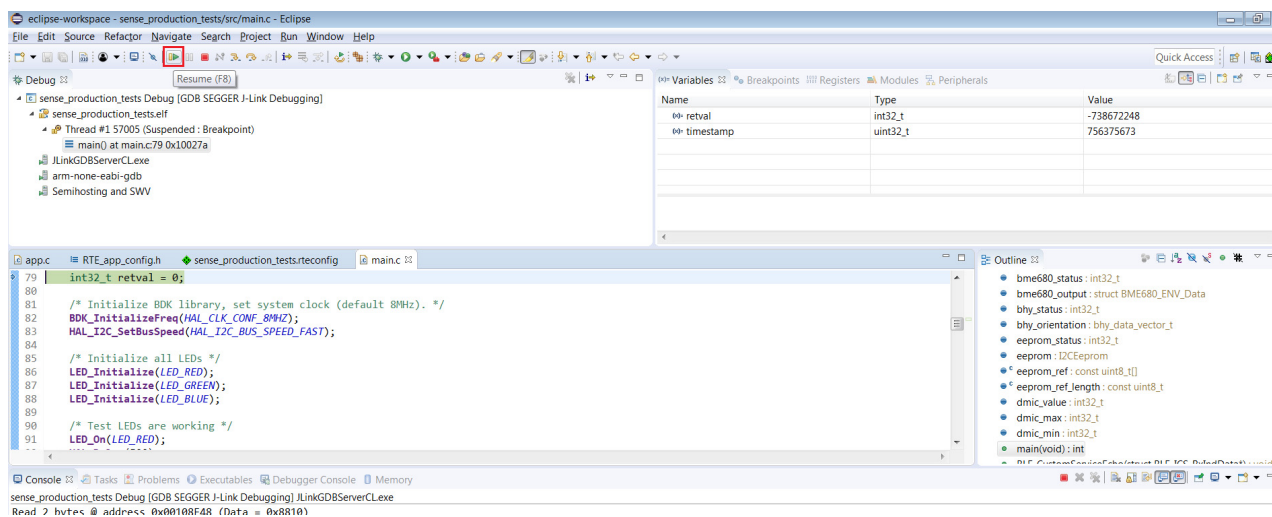


Figure 32.

Logging/Debugging

This is the next step after launching the session. For logging/debugging the downloaded Firmware, either J-Link RTT or Eclipse Console may be used. This section provides instructions for both.

Using J-Link RTT

25. After step 30 is done, open J-Link RTT viewer 6.32i (should be installed when J-Link software package was installed per Step 2)
26. Select USB / Existing session and click OK. As the debugger is in operation, easiest way is to utilize Existing Session.

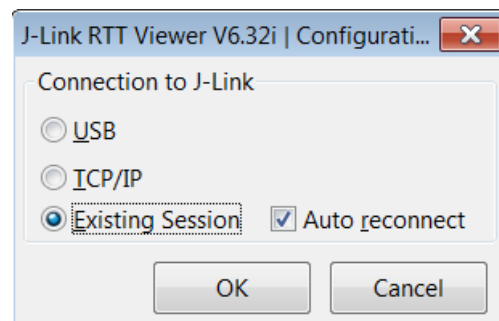


Figure 33.

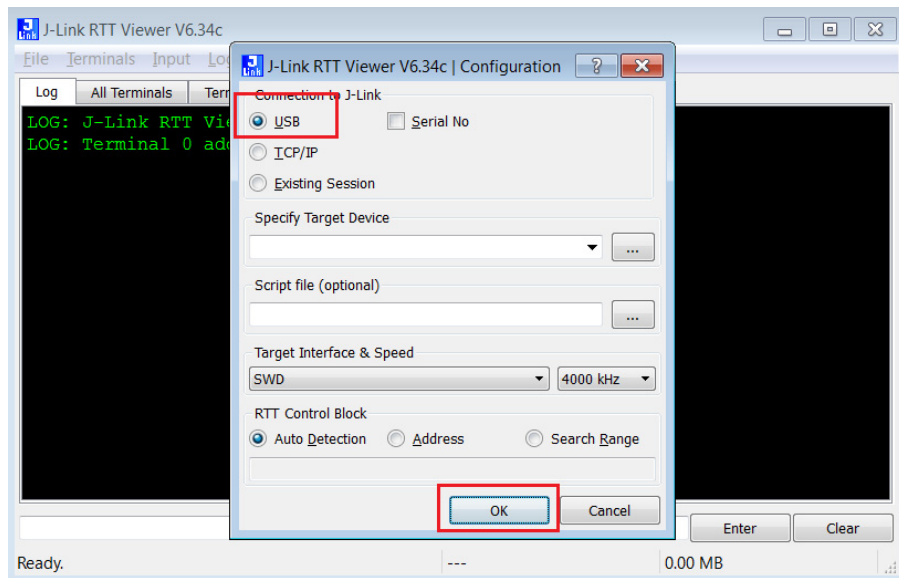


Figure 34.

27. RTT prompts you to select the appropriate microcontroller. Select RSL10 and click OK. The serial terminal is ready to use and the events from RSL10 can be observed by clicking the All Terminals Window.

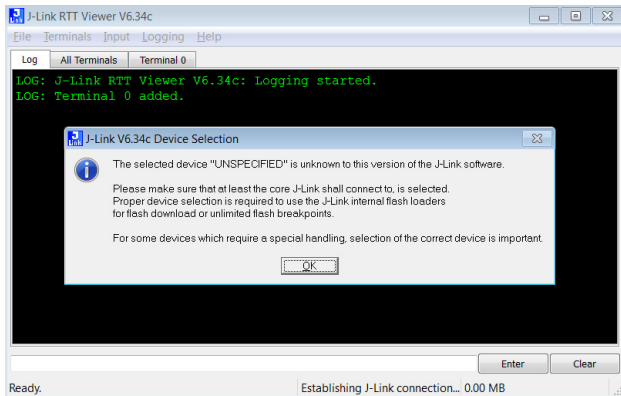


Figure 35.

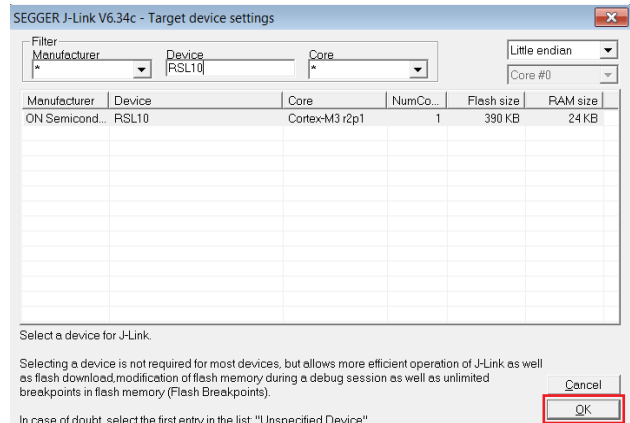


Figure 36.

28. Console returns the actual values from all sensors assembled on the board

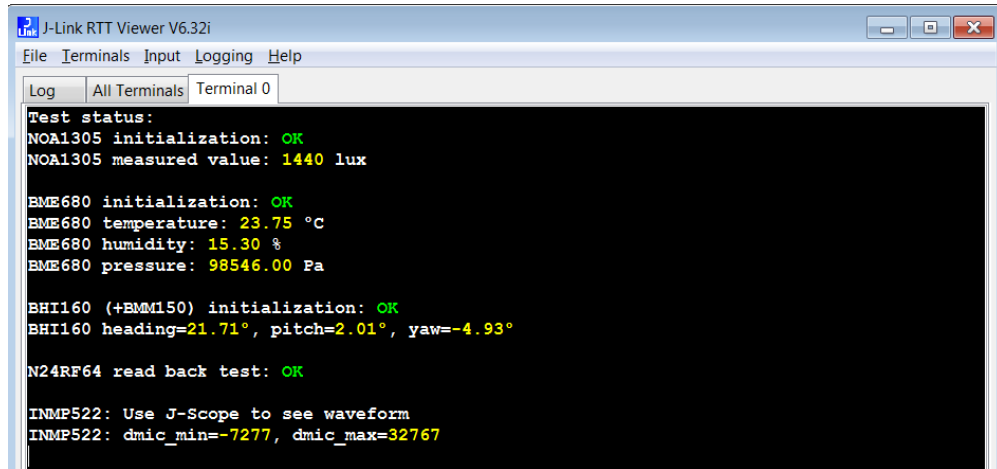


Figure 37.

Using Eclipse RTT Console

29. Click the Open a Terminal Icon

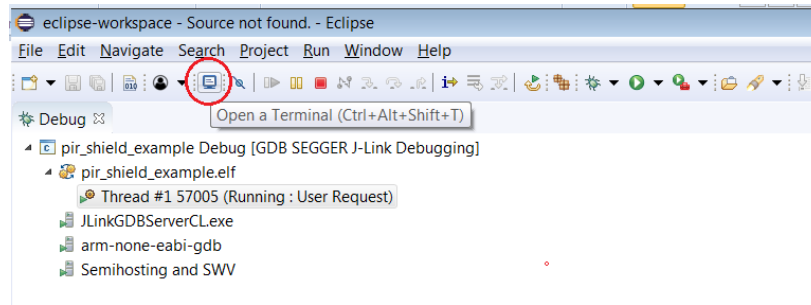


Figure 38.

30. Enter the values shown below and launch the session. The incoming events are printed on the terminal window.

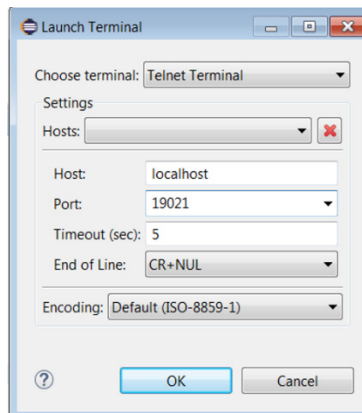


Figure 39.

31. Terminal window returns the same data format as using the J-LINK RTT viewer.

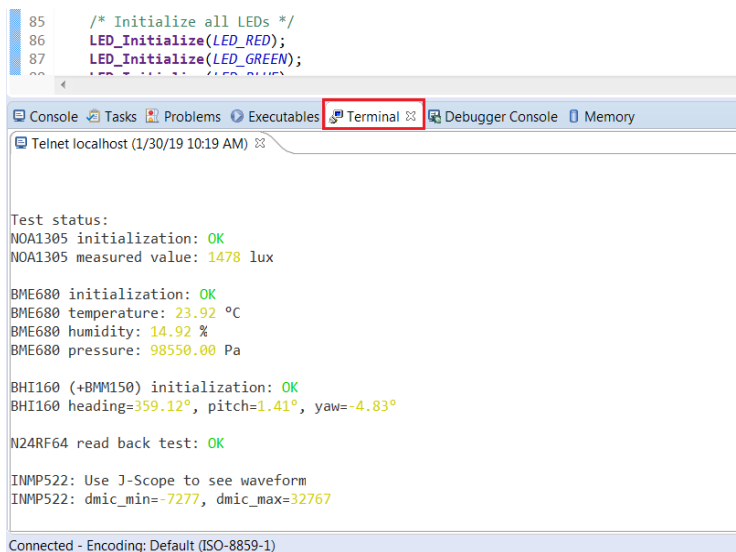


Figure 40.

Using J-scope for MIC data visualization

32. Launch Segger J-Scope and click on New project

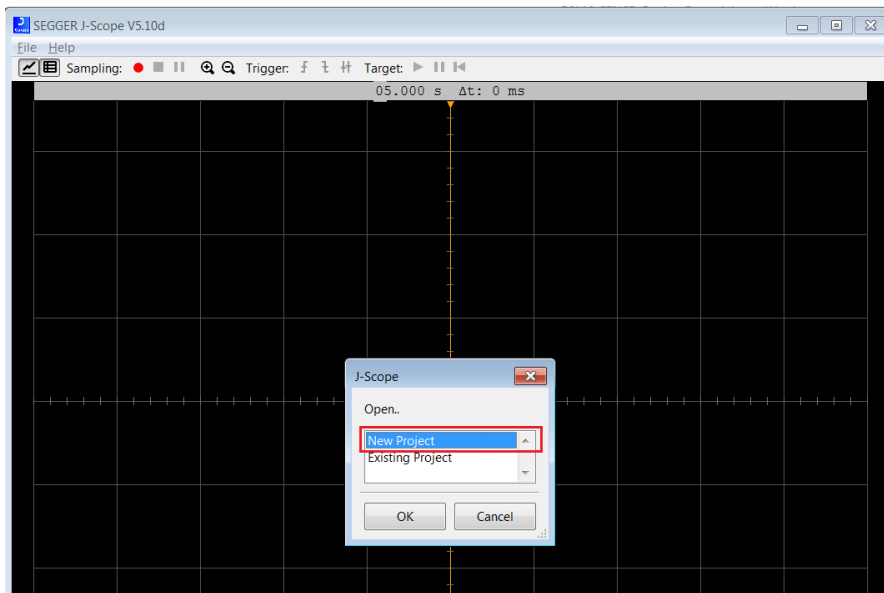


Figure 41.

33. When your Eclipse debug session is launched, use existing Session (alternatively you use USB), set Sample rate every 10us and load elf. file (binary) that is located under Eclipse-workspace and Debug folder.

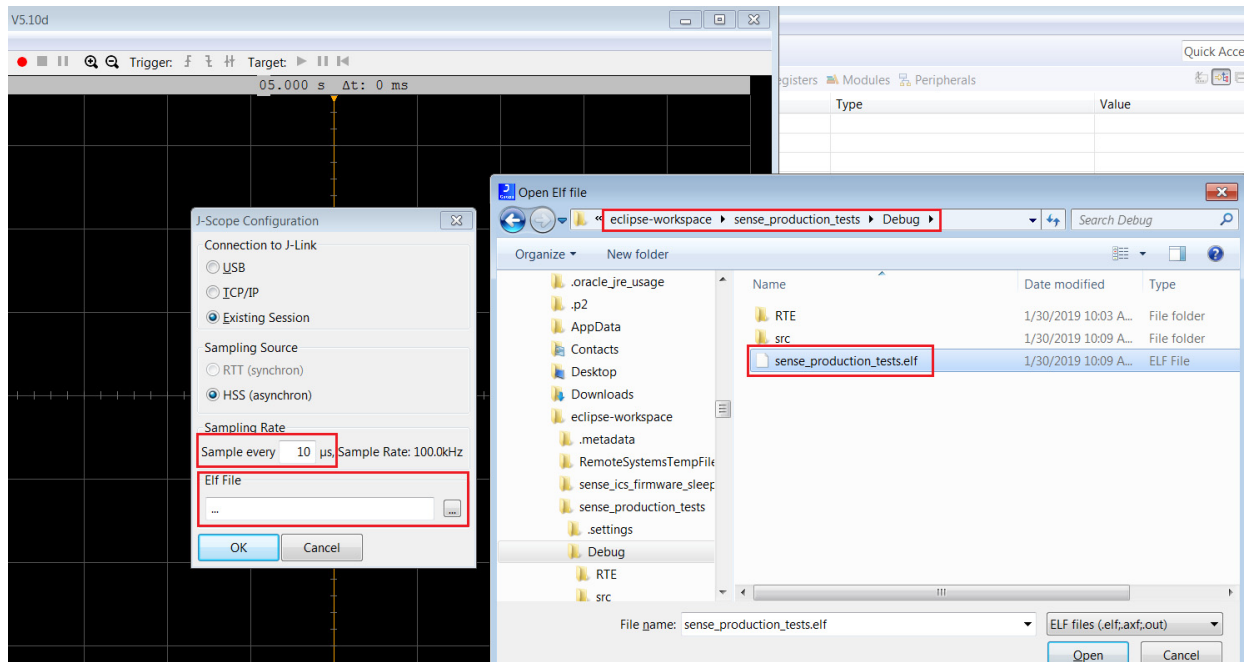


Figure 42.

34. J-Scope symbol section opens. Check `dmic_value` box and hit OK button.

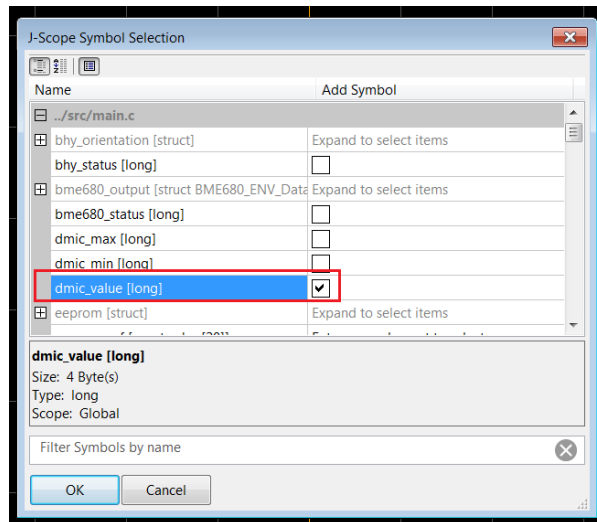


Figure 43.

35. Visualization of the audio is started when Red Sampling button is pushed (or F5.)



Figure 44.

Configuration Setup

System settings can be configured directly from within the CMSIS pack. Each example is equipped with basic system configuration that covers three main categories. These are accessible in the RTE/BDK folder within the project. Each system configuration starts with “RTE_”. As shown below,

opening the RTE_... header files using the CMSIS configuration wizard (right click on the header file), displays the configuration table. Various application specific parameters can be set. This allows pre-configuration of RSL10 without the need for explicit programming.

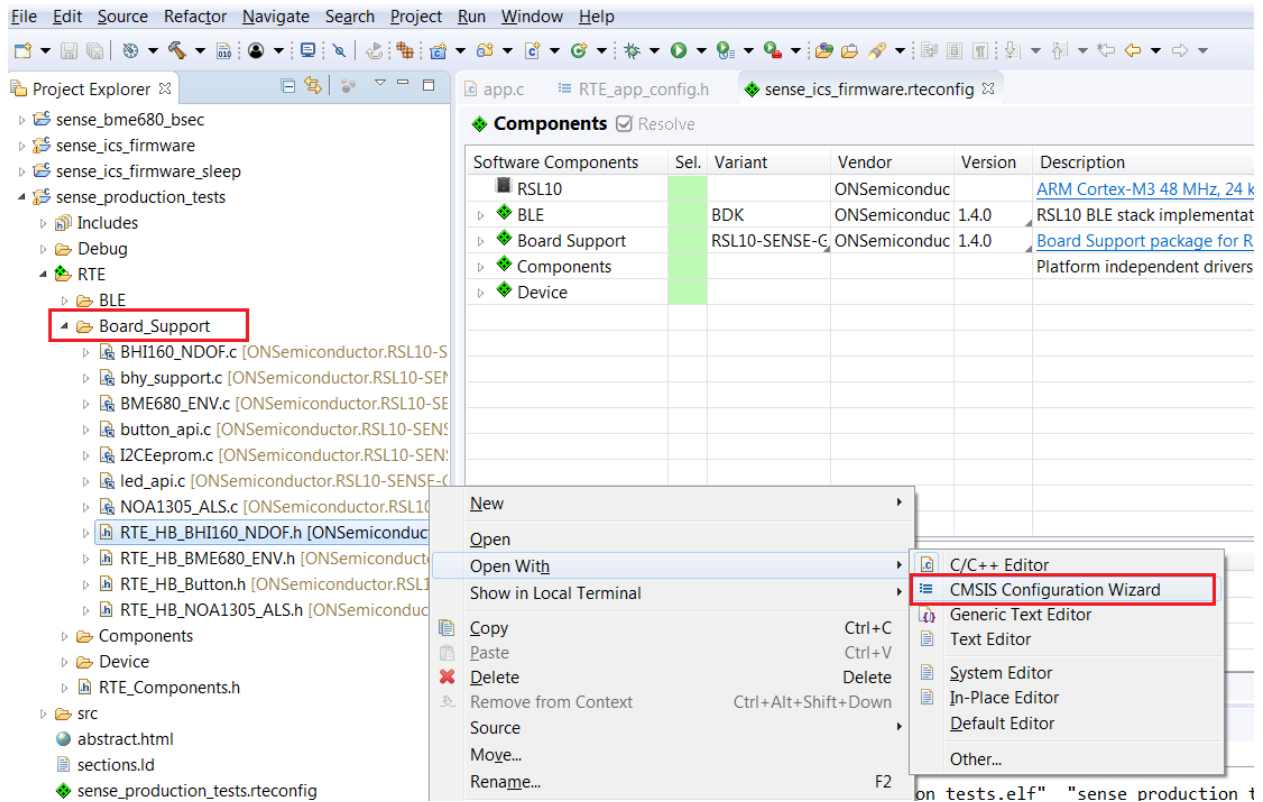


Figure 45.

A brief description on the header files is given in the wizard for various sensors.

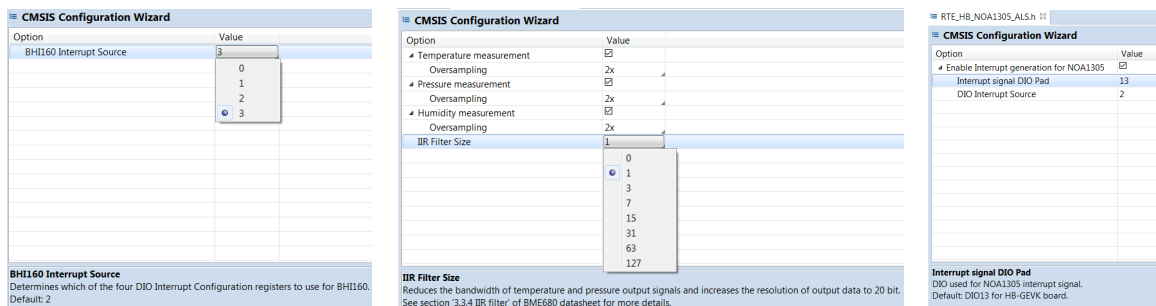


Figure 46.

DOCUMENTATION

Detailed documentation of all functions, code, APIs, HALs is part of the CMSIS package. Every use case (for a particular daughter card, service, etc.) copied into the workspace has its own manual with key description in the abstract.html page. URL Information and orderable part numbers are also provided as shown below.

*.rteconfig

The *.rteconfig file lists the software components within the CMSIS pack. To access the components, double click *.rteconfig file. Extensive help is provided under the description tab.

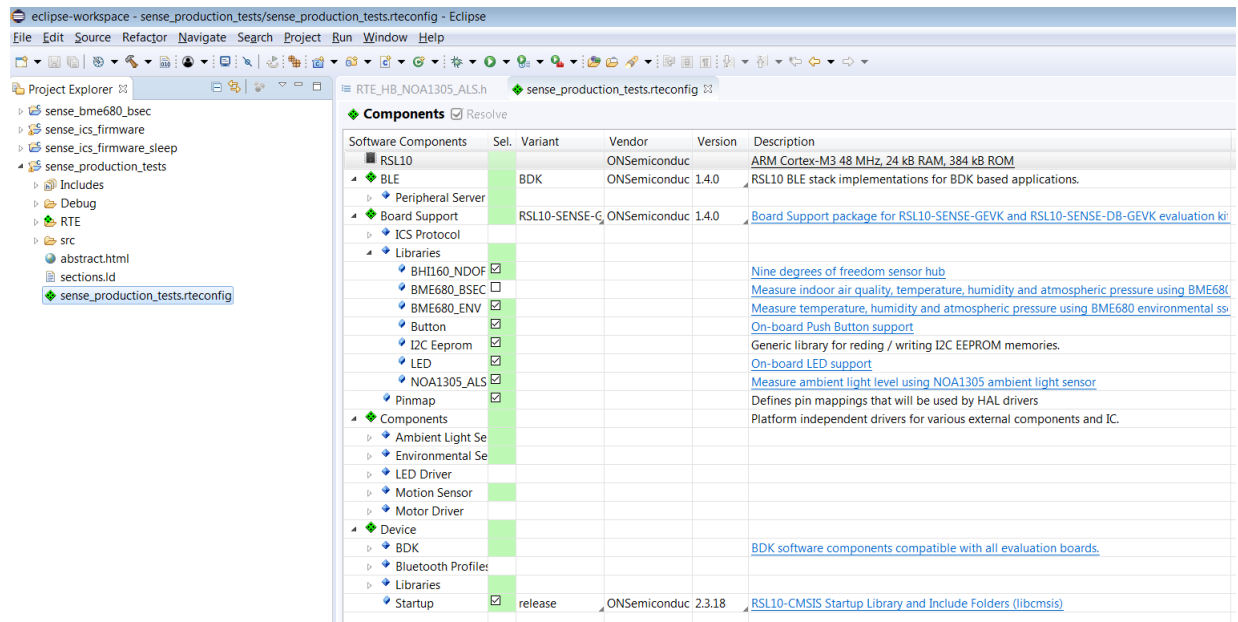


Figure 47.

Main Help Page

The main help page is accessible via Device/BDK, visible for all use cases in *.rteconfig file. It's further divided into various modules as shown below.

Software Components	Sel.	Variant	Vendor	Version	Description
RSL10			ON Semiconductor		ARM Cortex-M3 48 MHz, 32 kB RAM, 384 kB ROM
BLE		BDK	ON Semiconductor	1.0.0	RSL10 BLE stack implementations for BDK based applications.
Board Support		BDK-GEVK	ON Semiconductor	1.0.0	Board Support package for BDK-GEVK evaluation board.
Components					Platform independent drivers for various external components and IC.
Device					
BDK					BDK software components compatible with all evaluation boards.
AES	☐		ON Semiconductor	1.0.0	AES module from mbedTLS
Event Callback	☒		ON Semiconductor	1.0.0	Library for assigning of multiple callbacks to events.
HAL	☒		ON Semiconductor	1.0.0	Peripheral HAL drivers and RSL10 configuration
Output Redirection	☒	SEGGER RTT	ON Semiconductor	1.0.0	Redirects standard output calls using SEGGER RTT
Scheduling	☒		ON Semiconductor	1.0.0	Management layer for Event Kernel Application Task
Software Timer	☒		ON Semiconductor	1.0.0	Allows to create multiple timer events while using only single hardware timer.
Bluetooth Profiles					
Libraries					
Startup	☒	release	ON Semiconductor	2.1.10	RSL10-CMSIS Startup Library and Include Folders (libcmsis)

Figure 48.

BDK
Abstraction layers for RSL10 Bluetooth Development Kit based applications. More...
Modules
COMPONENTS
TASK_APP Management Application Task management & custom event scheduling.
Event Callback Library for attaching multiple callback functions (listeners) to single event source.
HAL Peripheral Hardware Abstraction Layer for RSL10.
Software Timer Allows creation of unlimited number of software timers with Ticker, Timeout and Timer functionality.
ANSI Terminal Color support Bring color to your terminal screen.
Target Evaluation board specific definitions.
API
Bluetooth Low Energy Library for handling of BLE functionality and libraries of supported BLE profiles.

Figure 49.

Sub-sections may be expanded for further information
(Ex: HAL interfaces shown below)

HAL BDK
Peripheral Hardware Abstraction Layer for RSL10. More...
Modules
Clock Configurations Defines possible clock configurations for proper operation of BDK.
I2C I2C interface for communication with connected shields.
SPI SPI interface for communication with connected shields.
UART UART interface for communication with connected shields.
Macros
#define HAL_TIME_RESOLUTION_US (1000)
#define HAL_TIME_ELAPSED_SINCE (start_timestamp) (HAL_Time() - start_timestamp)
#define HAL_OK (0)

Figure 50.

CMSIS also provides software timers and applications task manager abstraction layers to enable management of specific tasks and timing within the event kernel.

ON Semiconductor® ON BDK v1.0.0 Bluetooth LE Development Kit for RSL10
BDK
Abstraction layers for RSL10 Bluetooth Development Kit based applications. More...
Modules
COMPONENTS
TASK_APP Management Application Task management & custom event scheduling.
Event Callback Library for attaching multiple callback functions (listeners) to single event source.
HAL Peripheral Hardware Abstraction Layer for RSL10.
Software Timer Allows creation of unlimited number of software timers with Ticker, Timeout and Timer functionality.
ANSI Terminal Color support Bring color to your terminal screen.
Target Evaluation board specific definitions.
API
Bluetooth Low Energy Library for handling of BLE functionality and libraries of supported BLE profiles.

Figure 51.

Every example attached to the RSL10-SENSE-GEVK is equipped with addl. help under *abstract.html*

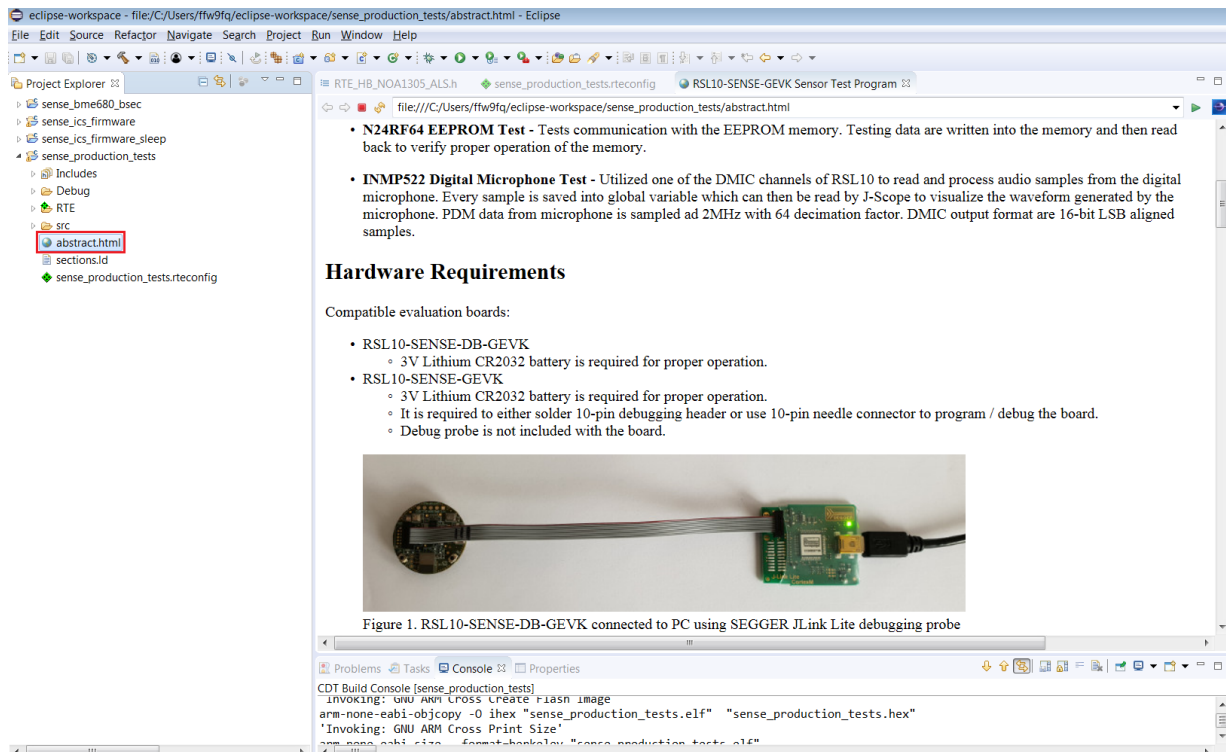


Figure 52.

Bluetooth is a registered trademark of Bluetooth SIG.

onsemi, **onsemi**, and other names, marks, and brands are registered and/or common law trademarks of Semiconductor Components Industries, LLC dba "**onsemi**" or its affiliates and/or subsidiaries in the United States and/or other countries. **onsemi** owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of **onsemi**'s product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marking.pdf. **onsemi** is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

The evaluation board/kit (research and development board/kit) (hereinafter the "board") is not a finished product and is not available for sale to consumers. The board is only intended for research, development, demonstration and evaluation purposes and will only be used in laboratory/development areas by persons with an engineering/technical training and familiar with the risks associated with handling electrical/mechanical components, systems and subsystems. This person assumes full responsibility/liability for proper and safe handling. Any other use, resale or redistribution for any other purpose is strictly prohibited.

THE BOARD IS PROVIDED BY ONSEMI TO YOU "AS IS" AND WITHOUT ANY REPRESENTATIONS OR WARRANTIES WHATSOEVER. WITHOUT LIMITING THE FOREGOING, ONSEMI (AND ITS LICENSORS/SUPPLIERS) HEREBY DISCLAIMS ANY AND ALL REPRESENTATIONS AND WARRANTIES IN RELATION TO THE BOARD, ANY MODIFICATIONS, OR THIS AGREEMENT, WHETHER EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING WITHOUT LIMITATION ANY AND ALL REPRESENTATIONS AND WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, NON-INFRINGEMENT, AND THOSE ARISING FROM A COURSE OF DEALING, TRADE USAGE, TRADE CUSTOM OR TRADE PRACTICE.

onsemi reserves the right to make changes without further notice to any board.

You are responsible for determining whether the board will be suitable for your intended use or application or will achieve your intended results. Prior to using or distributing any systems that have been evaluated, designed or tested using the board, you agree to test and validate your design to confirm the functionality for your application. Any technical, applications or design information or advice, quality characterization, reliability data or other services provided by **onsemi** shall not constitute any representation or warranty by **onsemi**, and no additional obligations or liabilities shall arise from **onsemi** having provided such information or services.

onsemi products including the boards are not designed, intended, or authorized for use in life support systems, or any FDA Class 3 medical devices or medical devices with a similar or equivalent classification in a foreign jurisdiction, or any devices intended for implantation in the human body. You agree to indemnify, defend and hold harmless **onsemi**, its directors, officers, employees, representatives, agents, subsidiaries, affiliates, distributors, and assigns, against any and all liabilities, losses, costs, damages, judgments, and expenses, arising out of any claim, demand, investigation, lawsuit, regulatory action or cause of action arising out of or associated with any unauthorized use, even if such claim alleges that **onsemi** was negligent regarding the design or manufacture of any products and/or the board.

This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and may not meet the technical requirements of these or other related directives.

FCC WARNING – This evaluation board/kit is intended for use for engineering development, demonstration, or evaluation purposes only and is not considered by **onsemi** to be a finished end product fit for general consumer use. It may generate, use, or radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment may cause interference with radio communications, in which case the user shall be responsible, at its expense, to take whatever measures may be required to correct this interference.

onsemi does not convey any license under its patent rights nor the rights of others.

LIMITATIONS OF LIABILITY: **onsemi** shall not be liable for any special, consequential, incidental, indirect or punitive damages, including, but not limited to the costs of requalification, delay, loss of profits or goodwill, arising out of or in connection with the board, even if **onsemi** is advised of the possibility of such damages. In no event shall **onsemi**'s aggregate liability from any obligation arising out of or in connection with the board, under any theory of liability, exceed the purchase price paid for the board, if any.

The board is provided to you subject to the license and other terms per **onsemi**'s standard terms and conditions of sale. For more information and documentation, please visit www.onsemi.com.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:

Email Requests to: orderlit@onsemi.com

onsemi Website: www.onsemi.com

TECHNICAL SUPPORT

North American Technical Support:

Voice Mail: 1 800-282-9855 Toll Free USA/Canada

Phone: 011 421 33 790 2910

Europe, Middle East and Africa Technical Support:

Phone: 00421 33 790 2910

For additional information, please contact your local Sales Representative