

EVBUM2589/D

Bluetooth® Low Energy IoT Development Kit (B-IDK) Getting Started Guide

INTRODUCTION

This document helps you get started with the Bluetooth Low Energy IoT Development Kit (B-IDK). The B-IDK is a comprehensive node-to-cloud and a modular IoT platform that allows development of various BLE based use cases. Along with the hardware and software, the B-IDK includes a mobile app to interact with sensors and actuators.

The B-IDK features RSL10, Industry's lowest power Bluetooth 5 SoC and comprises of a baseboard (BDK-GEVK) and several sensor and actuator daughter cards. For a complete listing of available daughter cards, please visit <https://www.onsemi.com/B-IDK>. The daughter cards connect to the baseboard, via the two PMOD connectors and/or the Arduino connector to enable various use cases.

Scope

This document covers the hardware setup, software architecture, B-IDK 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.

HARDWARE

- BDK-GEVK – B-IDK Baseboard
- Daughter Cards – Optional
- BDK-DCDC-GEVB – Power Shield For Use With Higher Power Daughter Cards – Optional

Default Configuration

The BDK-GEVK is shipped with the following jumper configuration. As the board supports OBD, there is no need for an external debugger. In case an external debugger is used, connect it to SWD header, J6.

Powering the Board

Multiple options are available to power the BDK-GEVK.

- USB
- Coin Cell (CR2032)
- External AC/DC Adapter plus power shield (BDK-DCDC-GEVB)
- External Supply

When higher power daughter cards (listed below) are attached to the baseboard, external supply either using the power shield or direct is required.

Higher Power Daughter Cards

- D-LED-B-GEVK Dual LED Ballast
- D-STPR-GEVK Dual Stepper Motor Driver
- BLDC-GEVK BLDC Motor Driver



ON Semiconductor®

www.onsemi.com

EVAL BOARD USER'S MANUAL

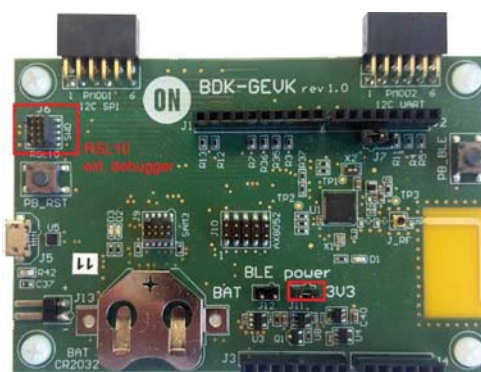


Figure 1. Board Photo

EVBUM2589/D

USB

The B-IDK can be powered via the USB port when the use case doesn't need any higher power daughter cards. An example configuration with the baseboard and a couple of sensor boards is shown below.



Coin Cell

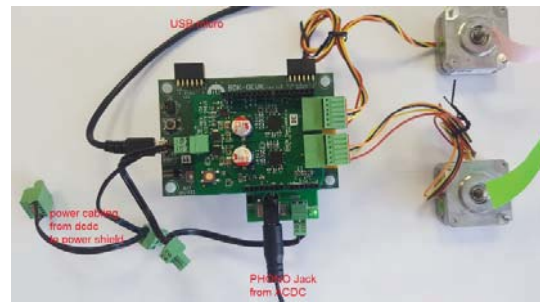
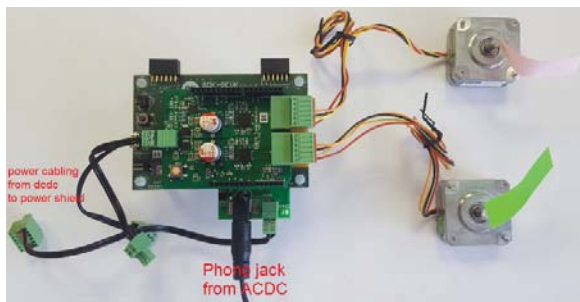
Once the firmware is flashed onto the baseboard, a coin cell (CR2032) may be used to power the system. Similar to USB based power supply, this method of powering is for use cases that don't utilize the higher power daughter cards. The jumper configuration must match the below table to allow for various power modes.

Table 1. JUMPERS

J11	J12	Usage
IN	X	Programming and Power over USB
X	IN	After programming. Only RSL10 is powered.
IN	IN	After programming. Both RSL 10 and OBD Microcontroller are powered

External AC/DC Adapter Plus Power Shield (BDK-DCDC-GEVB)

For use cases that utilize higher power daughter cards, an external AC/DC power supply (Ex: SMI24-12-V-P6) plus the power shield (BDK-DCDC-GEVB) are needed to power the system. While the 3.3 V supply to the baseboard is provided by the power shield via the Arduino connector, power cables (Green connector) are required between BDK-DCDC-GEVB and the higher power daughter card. For firmware flashing and debugging, the USB cable may be plugged in simultaneously with this mode as shown below.



External Supply

The B-IDK can be powered by an external supply via J13. In this mode, the battery cannot be installed. Jumpers J11 and J12 must be installed.

EVBUM2589/D

SOFTWARE

The B-IDK software allows for rapid development of various use cases. This section details the prerequisites and detailed steps in downloading firmware onto the baseboard.

Prerequisites

1. Install 64-bit version of Java from <https://www.java.com/en/download/>
2. Install J-Link Version 6.20f or later from <https://www.segger.com/downloads/jlink> (select J-Link software and documentation pack)
3. Download and “Install RSL10 Software Development Kit (SDK) Installer” from <http://www.onsemi.com/PowerSolutions/supportDoc.do?type=software&rpn=RSL10>
 - a. Download the RSL10 SDK Getting Started Guide and RSL10 SDK Oxygen Eclipse CMSIS pack 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

Document Title	Document ID/Size
RSL10 Bluetooth Mesh Getting Started Guide <small>NEW</small>	RSL10 Mesh Getting Started Guide (734kB)
RSL10 Bluetooth Mesh Package <small>NEW</small>	RSL10 Mesh Package (8386kB)
RSL10 Bluetooth Mesh Release Notes <small>NEW</small>	RSL10 Mesh Release Notes (7kB)
RSL10 SDK Getting Started Guide <small>NEW</small>	RSL10 SDK Getting Started Guide (1779kB)
RSL10 SDK LPD5P32 Package <small>NEW</small>	RSL10 LPD5P32 Package (9325kB)
RSL10 SDK Oxygen Eclipse CMSIS Pack <small>NEW</small>	ON Semiconductor.RSL10.2.1.10 (29697kB)
RSL10 SDK Release Notes <small>NEW</small>	RSL10 SDK Release Notes (22kB)
RSL10 Software Development Kit (SDK) Installer 2.1 <small>NEW</small>	RSL10 Development Tools (435874kB)
RSL10 Software Documentation Package <small>NEW</small>	RSL10 SDK Documentation Package (35959kB)
RSL10 Software Release Notes History <small>NEW</small>	RSL10 SDK Release Notes History (43kB)
RSL10 Software Signature Files <small>NEW</small>	RSL10 SDK Signature Files (1kB)
RSL10 Software Utility Apps <small>NEW</small>	RSL10 SDK Utility Apps (7649kB)

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)

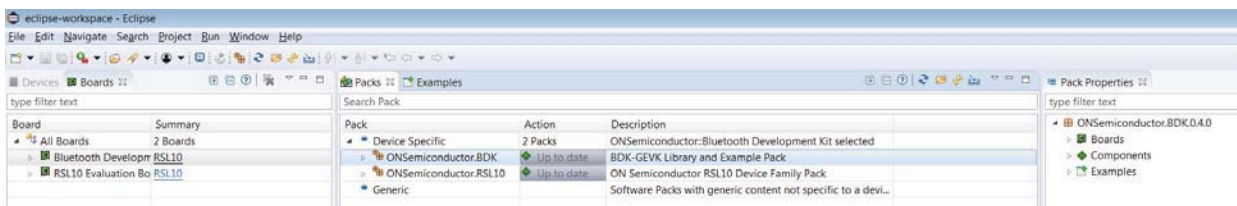
The next section provides details on importing the downloaded CMSIS packs into the SDK.

Importing CMSIS Packages

5. Launch the RSL10 SDK

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

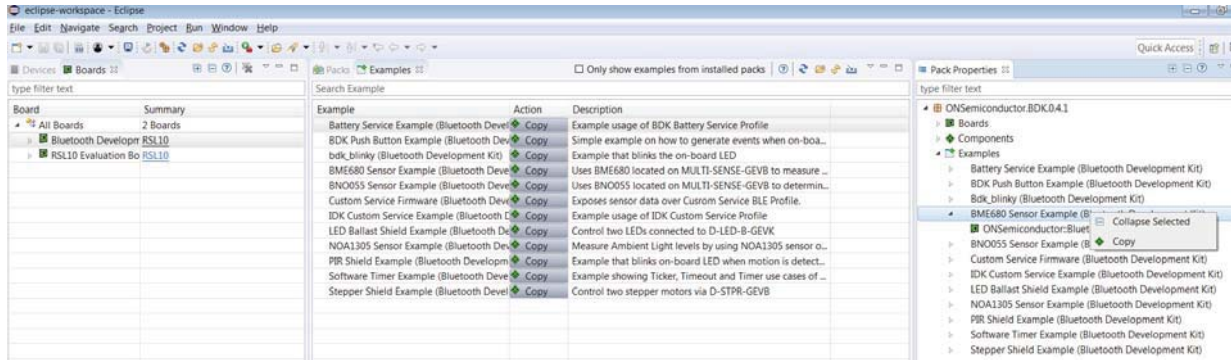
6. Refer to Chapter 3 of RSL10 SDK Getting Started Guide (step 3.a) for step-by-step instructions on importing the CMSIS packs.
7. Once the two packs are successfully imported, they can be viewed in the CMSIS pack manager perspective as shown below



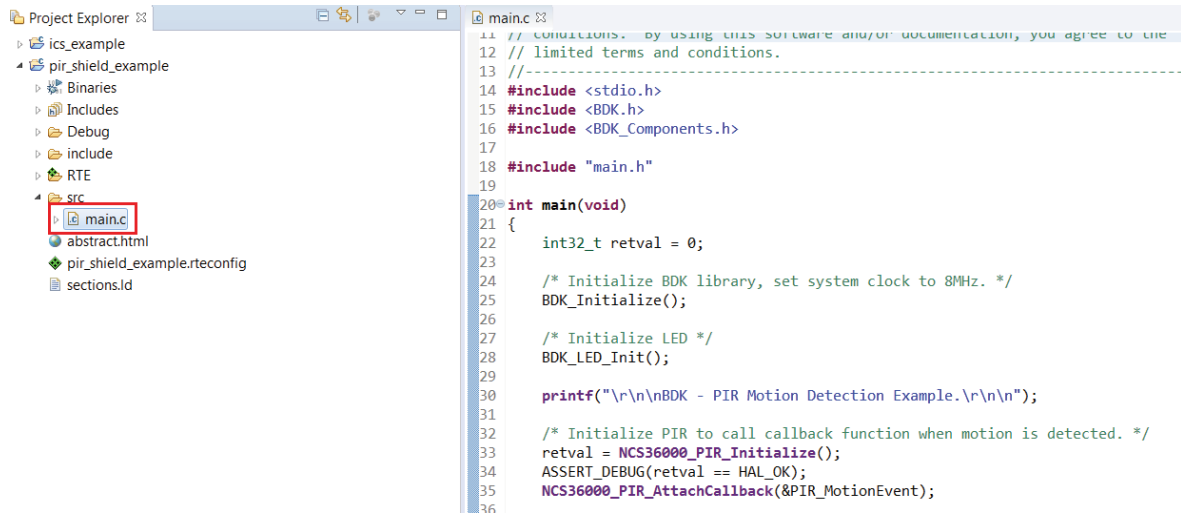
EVBUM2589/D

Compiling and Flashing

8. Choose an example (for example, `pr_shield_example`) to flash by copying it to the workspace.

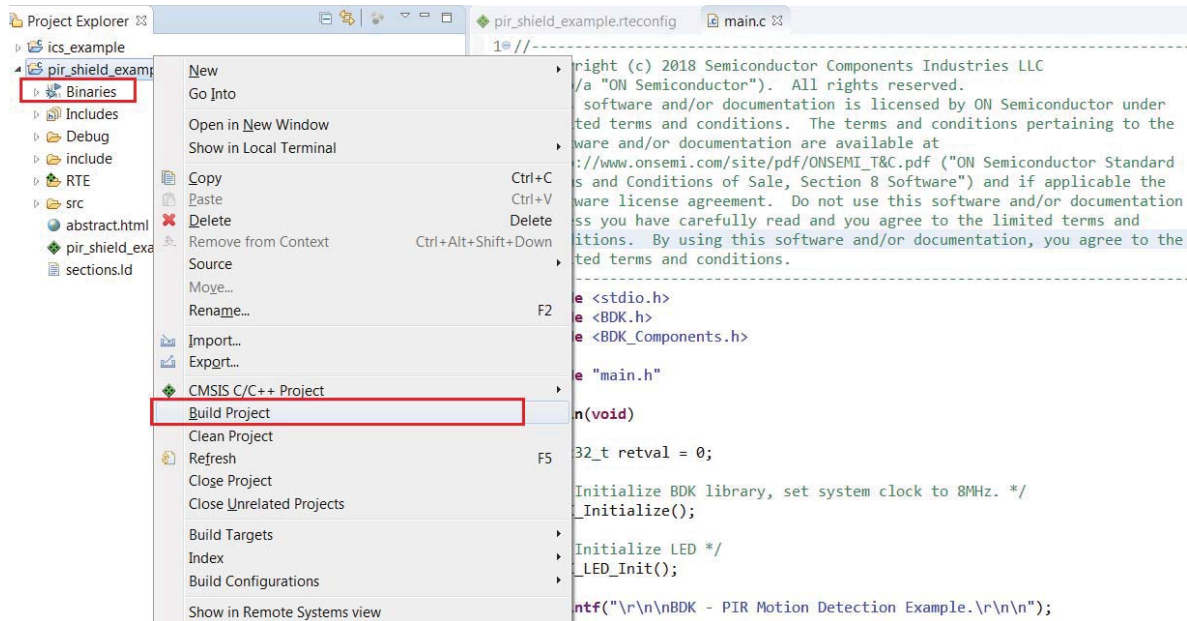


NOTE: Once the example is copied, it can be viewed under Project Explorer. All source files including main are located in the src folder.



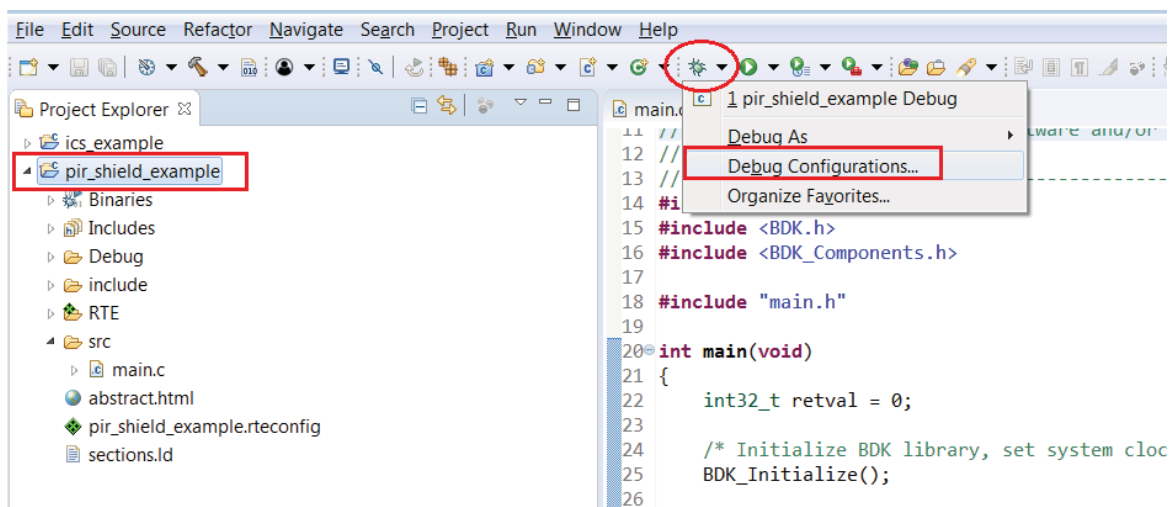
EVBUM2589/D

9. Right click and build the project. This creates binaries to be flashed to BDK–GEVK.



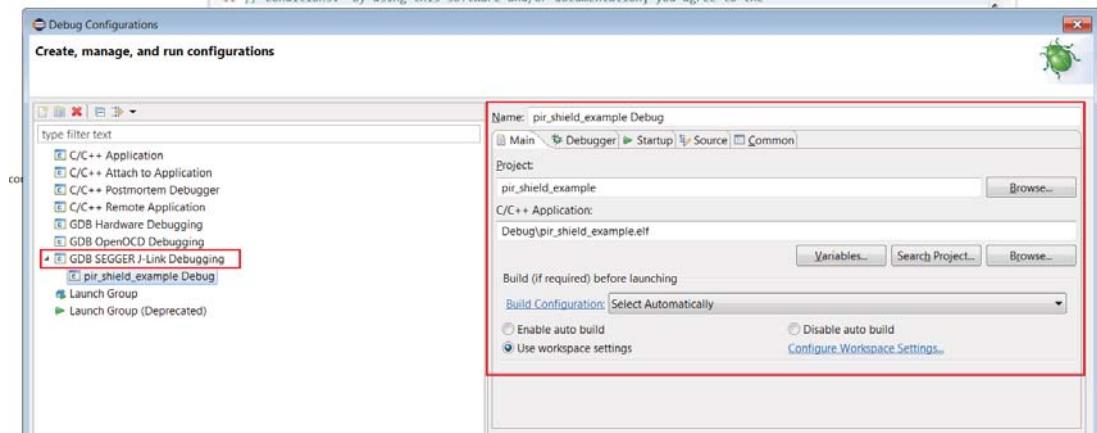
NOTE: If the binaries are not seen, press F5 (refresh)

10. Once the build is done, the code is ready to be flashed to the BDK–GEVK. Select the project (pir_shield_example), and go to debug configurations as shown below.



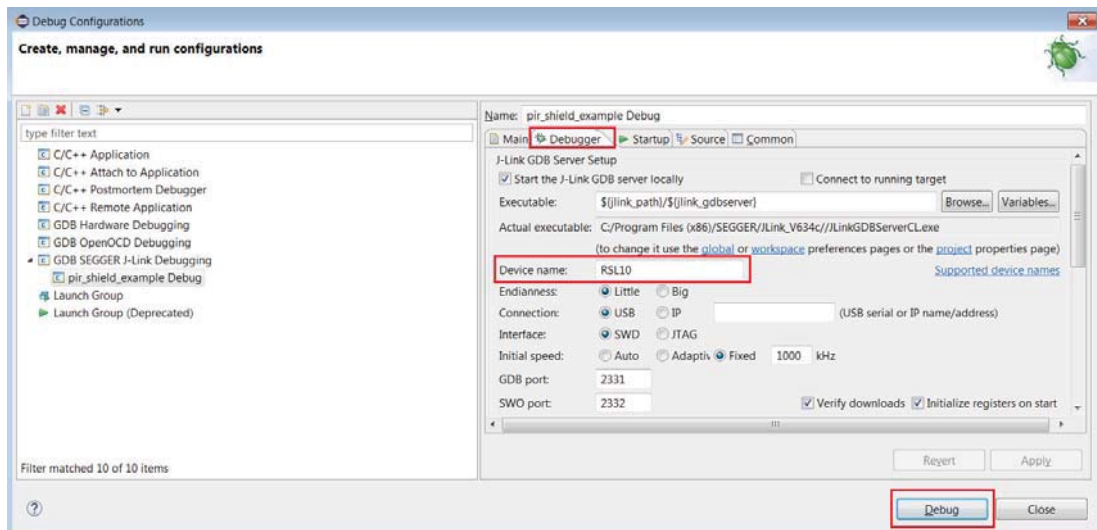
EVBUM2589/D

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

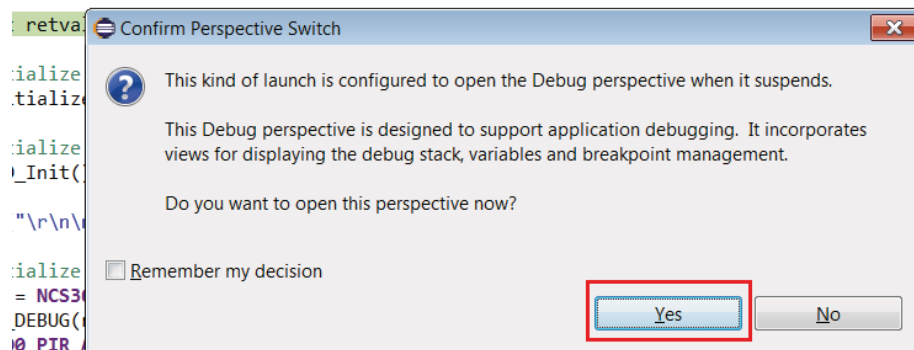


12. The debug configuration for the selected example is automatically saved and there's no need to re-create it

13. On the Debugger tab, set RSL10 as the device name. Click Debug to launch the code.

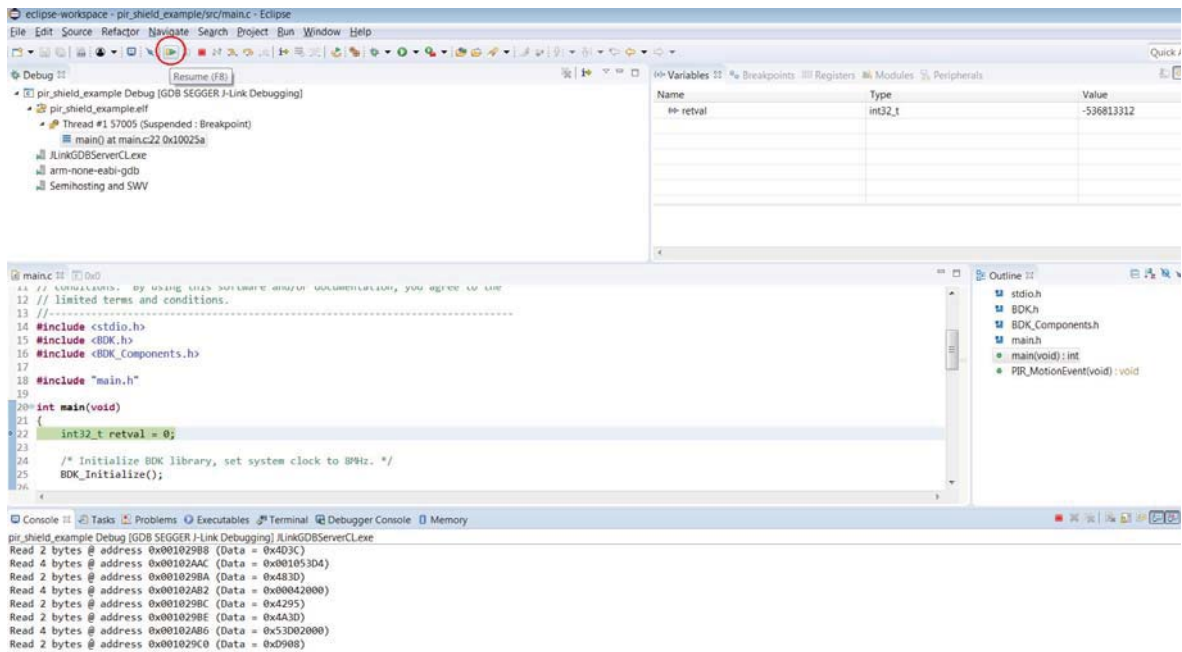


14. For application debugging, confirm perspective switch by clicking Yes.



EVBUM2589/D

15. The debug session is now launched. Click Resume (F8) to start the target CPU



Logging/Debugging

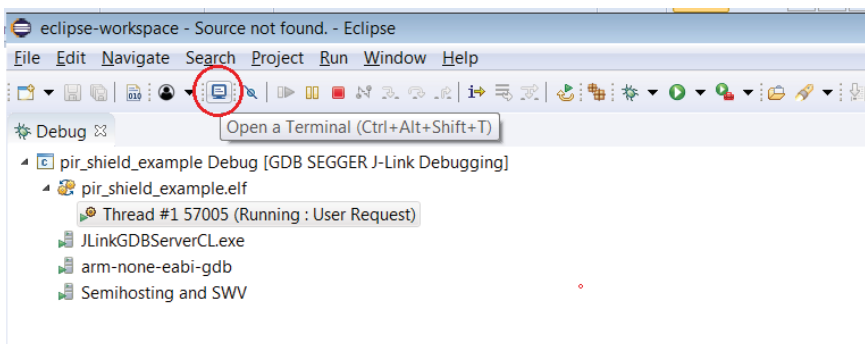
The following options are available to log/debug the downloaded firmware:

- Eclipse
- J-Link RTT
- AX8052F100 UART-SPI bridge

This section provides instructions for each of the above options.

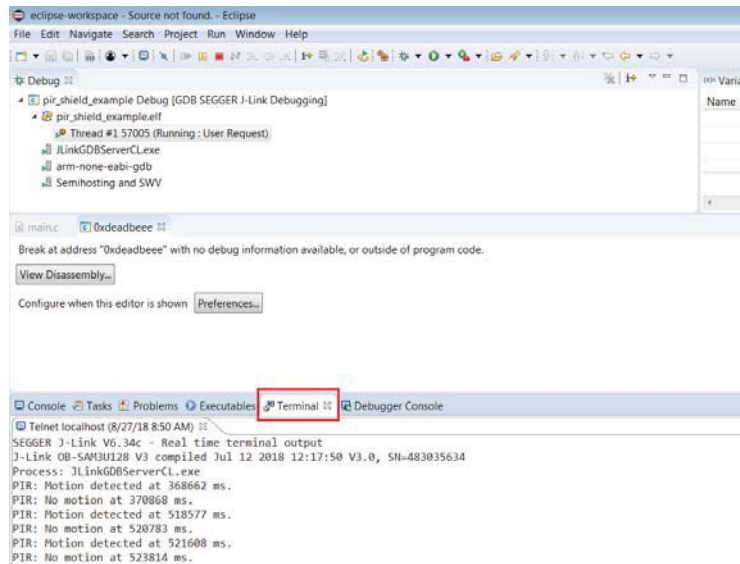
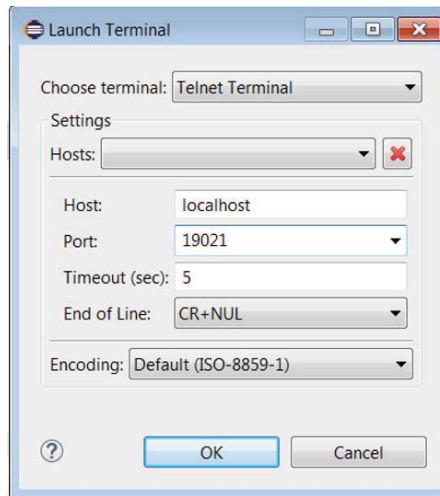
Using Eclipse

16. Click the Open a Terminal Icon



EVBUM2589/D

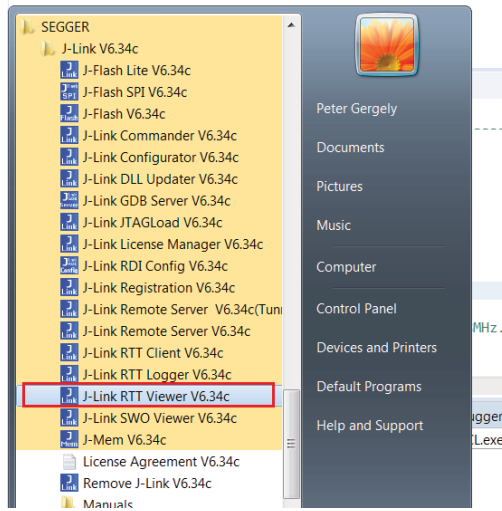
17. Enter the values shown below and launch the session. The incoming events are printed on the terminal window.



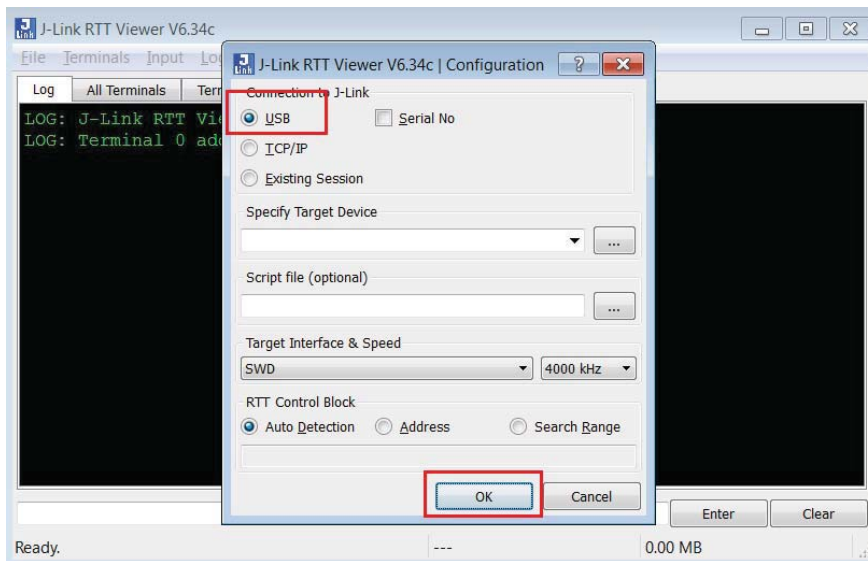
EVBUM2589/D

Using J-Link RTT

18. After step 14 is done, open J-Link RTT viewer (should be installed when J-Link software package was installed per Step 2)

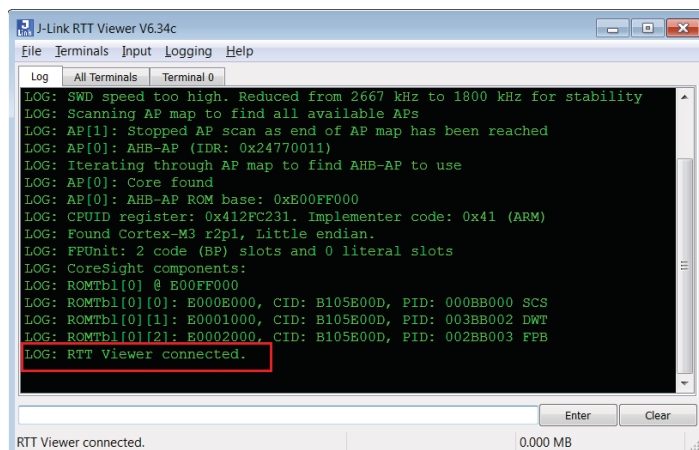
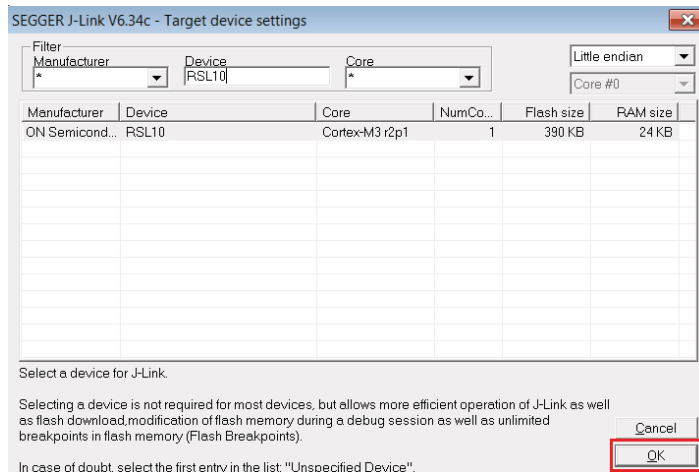
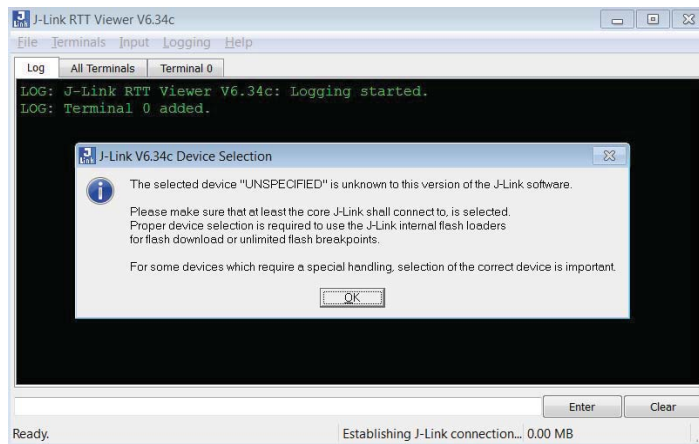


19. Select USB and click OK

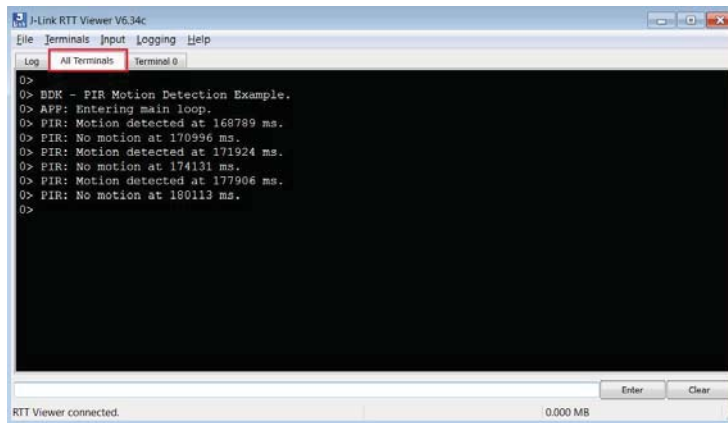


EVBUM2589/D

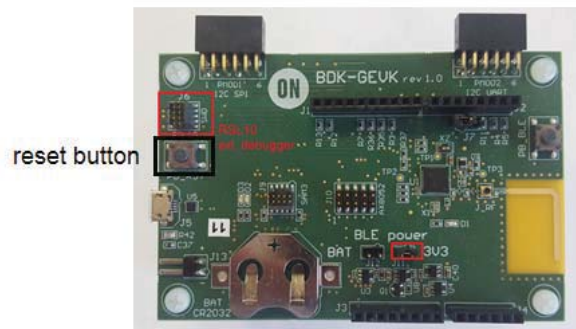
20. RTT prompts you to select the appropriate microcontroller. Select RSL10 and click OK. The serial terminal is ready to use and the events from RSL10 can be observed by clicking the All Terminals Window.



EVBUM2589/D



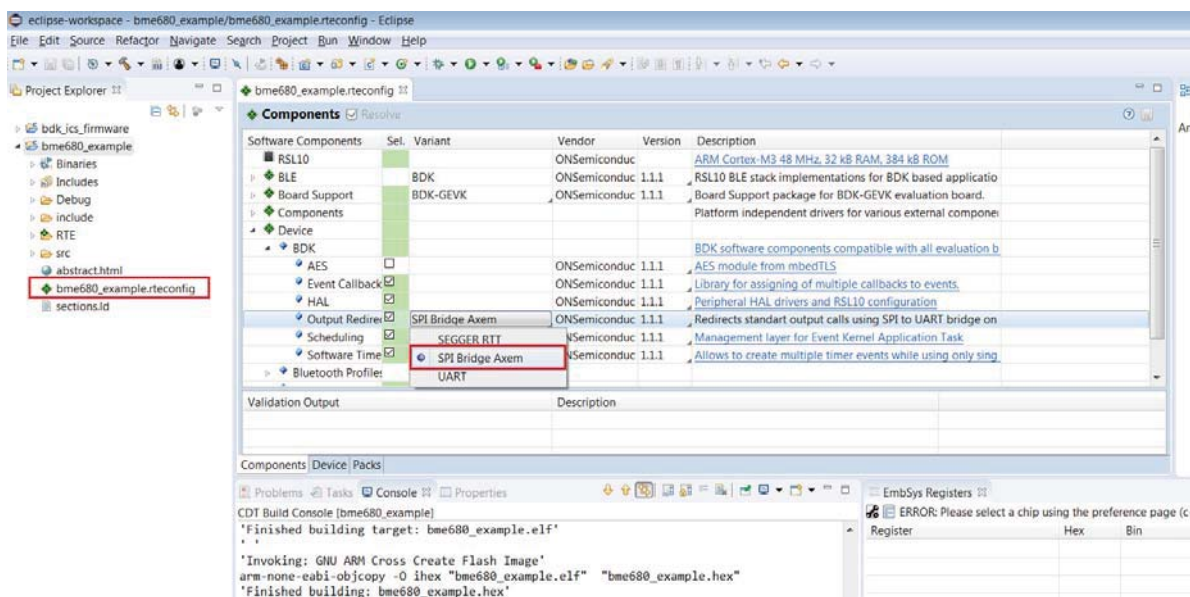
NOTE: You may reset (PB_RST) the BDK-GEVK (shown below) to launch RTT terminal without needing to launch Eclipse.



Using Eclipse Serial Console via UART-SPI Bridge

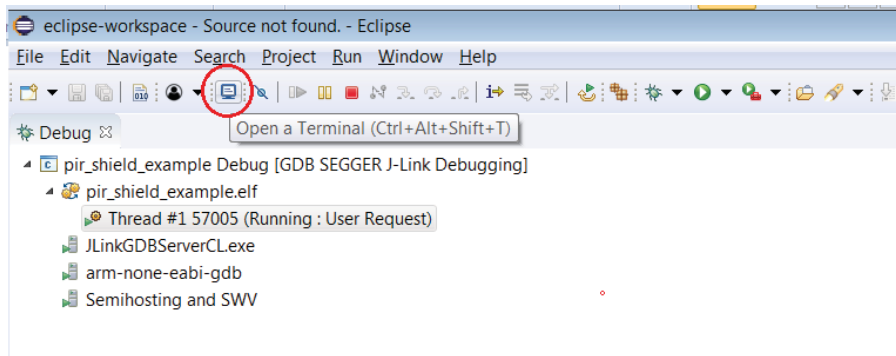
The BDK-GEVK board is equipped with UART-SPI microcontroller AX8052F100 flashed with special firmware, to enable serial communication with values returned to Terminal.

21. Click on example's rteconfig file and choose "SPI Bridge AXEM" under *Device/BDK/Output redirection*. Save, compile and flash the whole project.

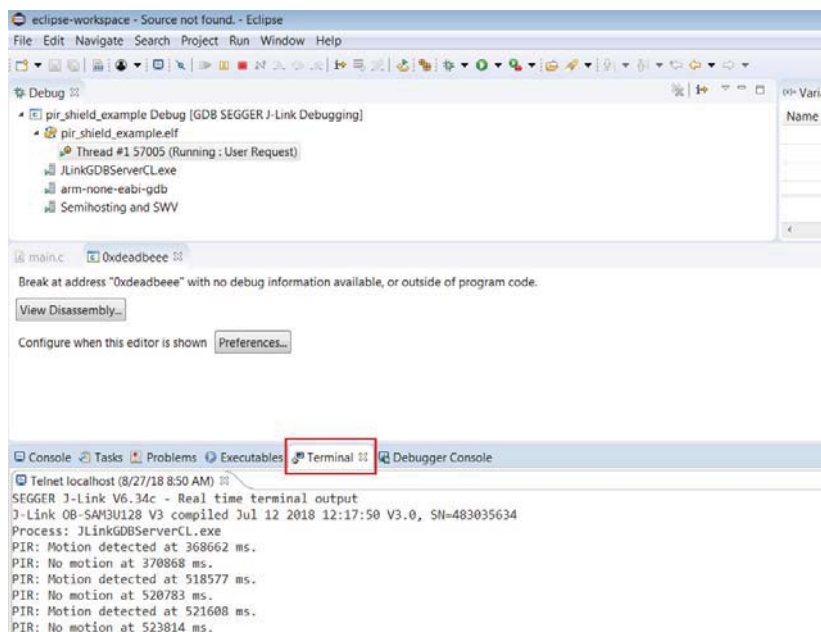
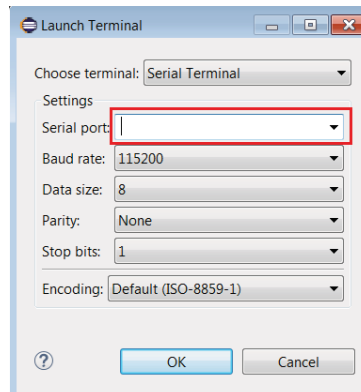


EVBUM2589/D

22. When the project runs, Click the Open a Terminal Icon.



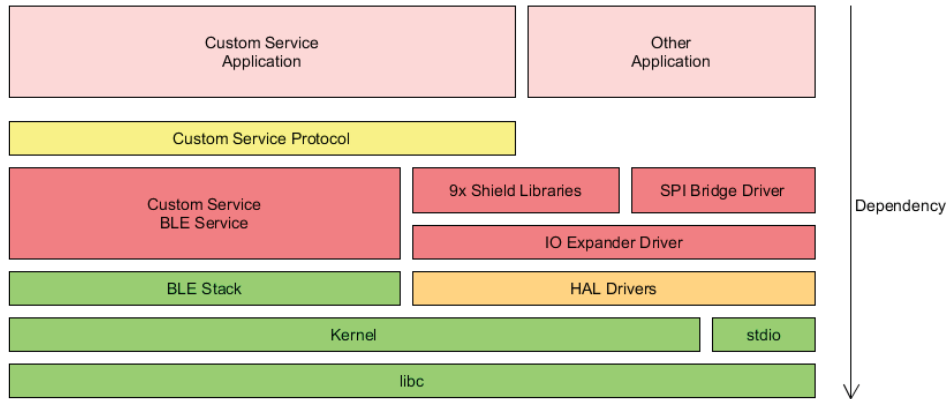
23. Enter the appropriate COM port as shown below and launch the session. The incoming events are printed on the terminal window.



EVBUM2589/D

SOFTWARE ORGANIZATION

For users modifying the example code and building new projects, the following sections detail the B-IDK software organization. The stack overview is shown below.



B-IDK CMSIS Software Organization

CMSIS pack and the associated software components handle multiple evaluation boards as different bundles of the standardized Board Support Cclass.

- This bundle shows only components supported by ON Semiconductor for a given board
- No confusing component variants

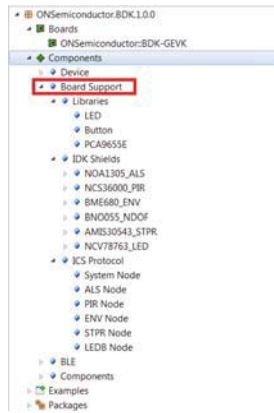
Common libraries and HAL are in a separate group within the Device class

Cbundle	Cclass	Cgroup	Csub	Cvariant	Description
BDK-GEVK	Board Support	Libraries	LED		Board support package for BDK-GEVK evaluation board
			Button		Board specific libraries
			PCA9655E		On-board LED support
		IDK Shields	PIR-GEVB		On-board push button support
			ALS-GEVB		16-bit I2C IO Expander library
			MULTI-SENSE-GEVB	rev2.1	Support for Arduino / PMOD extension boards
			BLDC-GEVK		PIR Motion detection using NCS36000
			D-LED-B-GEVK		Measure Ambient light levels using NOA1305 ambient light sensor
		ICS Protocol	D-STPR-GEVK		Combines 3 sensors: BME680, BNO055, NOA1305
			System Node		Libraries that allow connected BLE devices to take control over sensors / actuators using ICS Service.
			PIR Node		Protocol implementation and system node used by other sensor / actuator nodes.
			ALS Node		Exposes motion data provided by NCS36000 from PIR-GEVB
			ENV Node		Exposes ambient light levels measured by NOA1305 from ALS-GEVB
			AO Node		Exposes environmental data measured by BME680 from MULTI-SENSE-GEVB
			STPR Node		Exposes absolute orientation measured by BNO055 from MULTI-SENSE-GEVB
	Components	LEDB Node		Allows remote control of two stepper motors connected to D-STPR-GEVB.	
		BLDC Node		Allows remote control of two power LEDs connected to D-LED-B-GEVK	
				Allows to remote control BLDC motor connected to BLDC-GEVB.	
		LED Driver	NCV78763	Platform independent software drivers for controlling of various external IC.	
		Ambient Light Sensor	NOA1305	Dual LED Driver and Power Ballast, for Automotive Front Lighting, 1.6 A, 2nd Generation	
Motor Driver		AMIS-3054	Ambient Light Sensor with I2C Interface and DarkCurrent Compensation		
Environmental Sensor		LV907UW	Micro-stepping stepper motor driver with SPI interface for bipolar stepper motors		
Motion sensor		bme680	Sensor-less Three-phaseBrushless DC MotorController, with GateDrivers, for Automotive		
Device	BDK	Touch Sensor	LC717A00AR	Low power gas, pressure, temperature & humidity sensor	
		HAL		Intelligent 9-axis absolute orientation sensor	
		Scheduling		Capacitance-Digital-Converter for Electrostatic Capacitive Touch Sensors	
		Software Timer			
		Event Callback		RSL10 Peripheral abstraction layers for BDK applications.	
		Output Redirection		Event Kernel wrapper for BDK applications.	
				Allows to create multiple timer events while using only single hardware timer.	
				Library for executing multiple event handlers when an event occurs.	
				Redirects standard library output calls (printf, ...) to specified channel	
				Output is transmitted using UART peripheral	
BDK	BLE	Peripheral Server	Battery Service		Output is transmitted over SWD using the on-board or external J-LINK debug probe
			ICS Service		
			Peripheral Server		Exposes current battery level to connected client and application. IDK Custom Service used to transmit sensor data using ICS Protocol library. BLE Peripheral Server implementation for BDK applications.

EVBUM2589/D

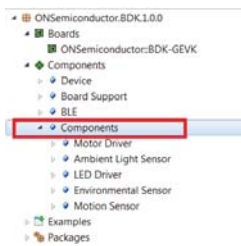
Board Support

- Libraries to support BDK–GEVK, GPIO Expander, Various daughter cards and custom protocol (required for the mobile app)



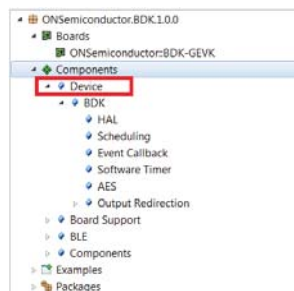
Components

- Libraries attached to board support



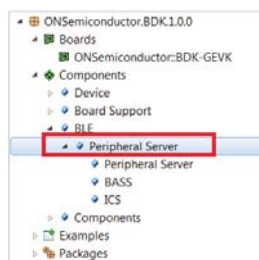
Device

- Abstraction layers for interfaces, timers, AES, serial re-direction, etc.



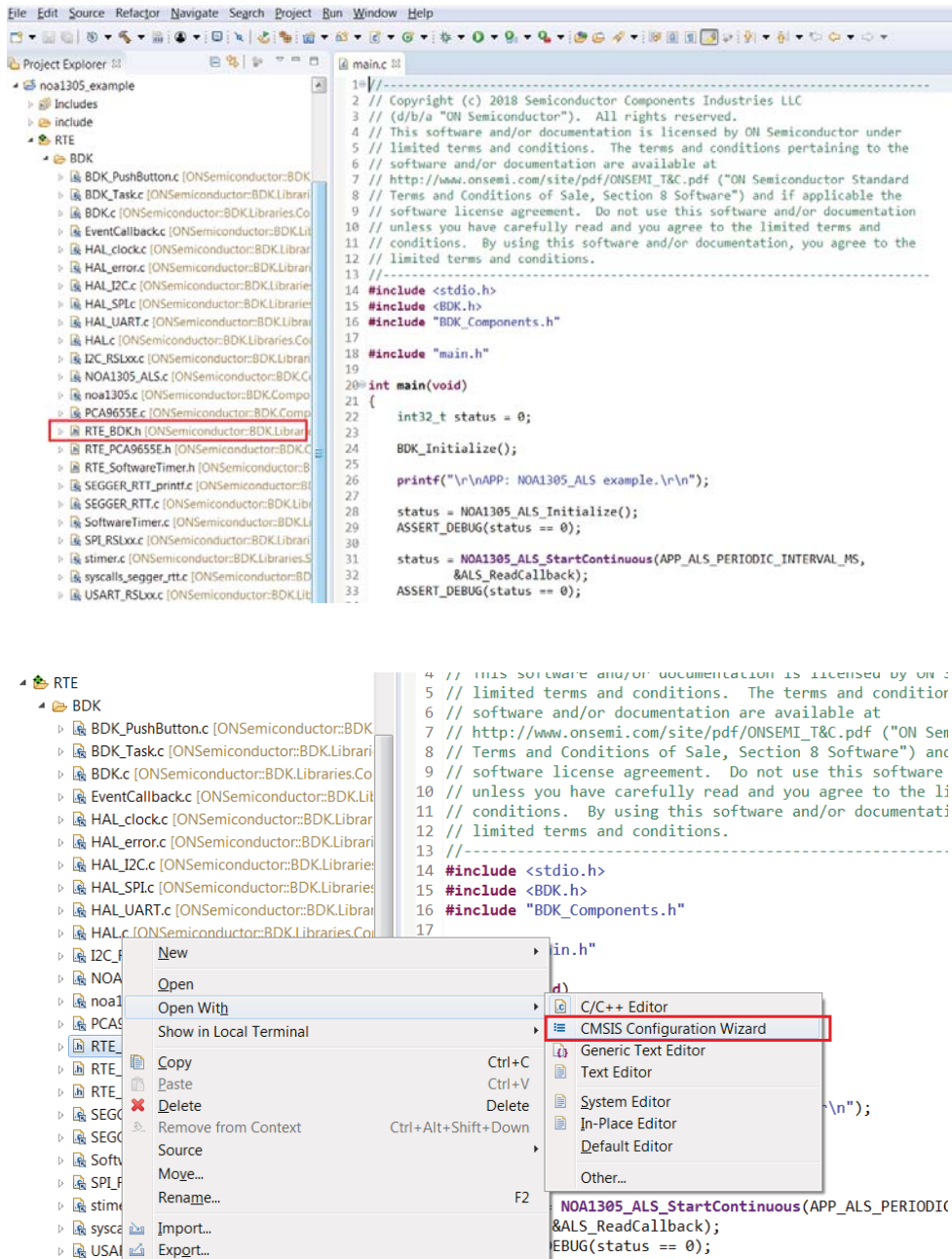
BLE

- Peripheral Server Support



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.

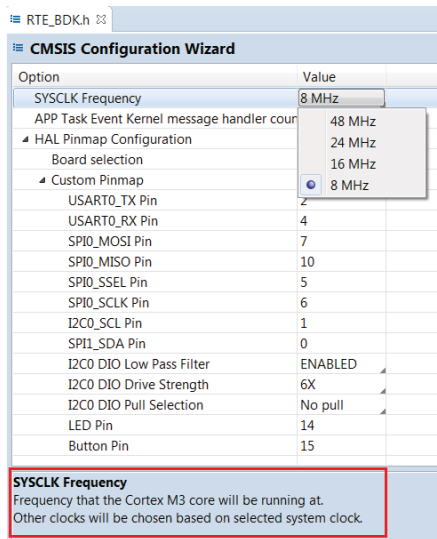


A brief description on the header files is given below.

EVBUM2589/D

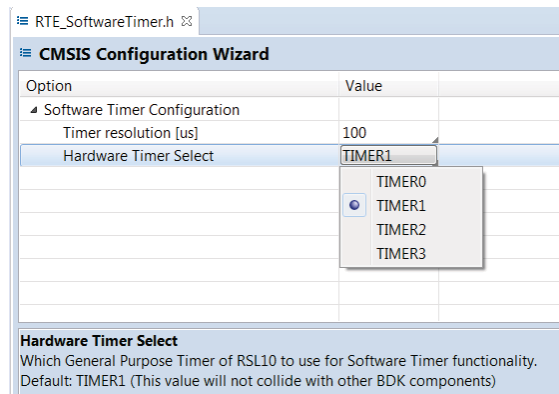
RTE_BDK.h

Parameters such as system clock frequency and the board that feature RSL10 (default set to BDK-GEVK), etc. can be set. Descriptions of each of these parameters are also provided.



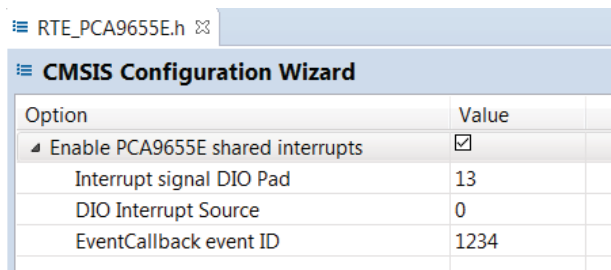
RTE_Software_Timer.h

Various timers (4) supported by RSL10 can be configured by invoking the CMSIS configuration wizard on this header file. Timer 1 is used for B-IDK components.



RTE_PCA9655.h

PCA9655 is the GPIO expander chip assembled on most daughter cards to expand interface functionality. Parameters related to this chip can be set here.



EVBUM2589/D

RTE_x.h

In addition to configuring system settings, all the supported daughter cards' parameters can be configured directly using the configuration wizard, without the need for programming. Once the parameters are changed per the application requirements, saving, rebuilding and flashing the project will let the new parameters take effect. Examples for the stepper and LED ballast daughter cards are shown below. Other daughter cards can be configured in a similar fashion.

RTE_AMIS30543_STPR.h

CMSIS Configuration Wizard

Option	Value
▲ Stepper Shield Left Channel	
Step Mode	1 / 4 Micro - Step
Coil Peak Current	245 mA
Direction Of Rotation	CW motion
NXT Edge Trigger	Rising Edge
Turn On / Off Slopes of Motor Driver	Very Fast
Speed Load Angle Transparency Bit	SLA is not transparent
Speed Load Angle Gain	0.5
Enables doubling of the PWM frequency	<input type="checkbox"/>
Enables jittery PWM	<input type="checkbox"/>
Steps Per Revolution	200
▲ Stepper Shield Right Channel	
Step Mode	1 / 4 Micro - Step
Coil Peak Current	1 / 32 Micro - Step
Direction Of Rotation	1 / 128 Micro - Step
NXT Edge Trigger	1 / 64 Micro - Step
Turn On / Off Slopes of Motor Driver	Compensated Full Step, 2 phase on
Speed Load Angle Transparency Bit	Compensated Full Step, 1 phase on
Speed Load Angle Gain	1 / 16 Micro - Step
Enables doubling of the PWM frequency	1 / 8 Micro - Step
Enables jittery PWM	<input checked="" type="radio"/> 1 / 4 Micro - Step
Steps Per Revolution	Compensated Half Step
	Uncompensated Half Step
	Uncompensated Full Step

Step Mode
 Default: 1 / 4 Micro - Step (for motors provided with Stepper shield)

RTE_NCV78763_LED.h

CMSIS Configuration Wizard

Option	Value
▲ Enable Booster	<input checked="" type="checkbox"/>
Booster PWM generation	Internal
Booster PWM Frequency	242 kHz
Booster Clock Inversion	<input type="checkbox"/>
Booster Slope Compensation	10 mV / us
Booster Error Amplifier Gain [Siemens]	30 uS
Booster Overvoltage Shutdown	5.8 V
Booster Overvoltage Reactivation	-1 V
Booster Gate Voltage Threshold	0.4 V
Booster Minimum Off Time	115 ns
Booster Minimum On Time	150 ns
Booster Regulation Setpoint Voltage	45.0 V
Booster Current Limitation Peak Value	100 mV
Activate VBOOST_AUX_SUPPLY	<input type="checkbox"/>
Booster Skip Clock Cycles	Disabled
▲ Enable Buck Regulator Channel 1	<input checked="" type="checkbox"/>
D-LED-B-GEVK Channel 1 Peak current [m 252	
D-LED-B-GEVK Channel 1 Average current 140	
Enables the offset compensation for buck	<input type="checkbox"/>
Comparator Threshold Voltage	0
Tunes the Toff x VLED value for channel 1	0
> Overcurrent Settings	
> Enable Buck Regulator Channel 2	<input checked="" type="checkbox"/>
▲ General Settings	
Thermal warning threshold	0
LED sampling duration selection	88

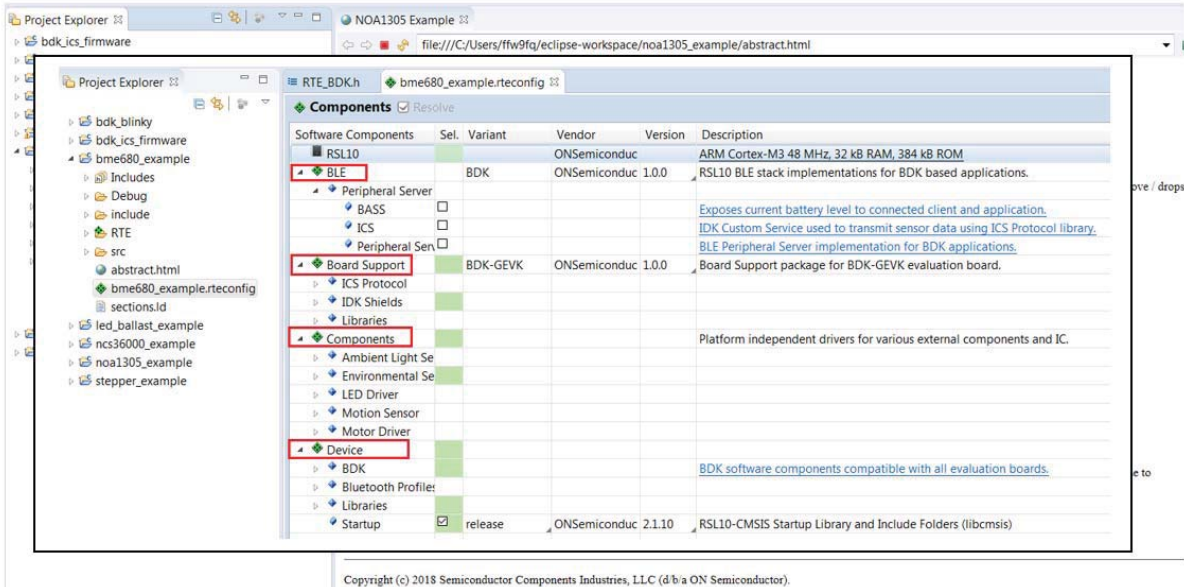
Booster Overvoltage Reactivation
 Defines the hysteresis for the reactivation once the overvoltage shutdown is triggered.
 Default: -1 V for D-LED-B-GEVK

Source Editor CMSIS Configuration Wizard

EVBUM2589/D

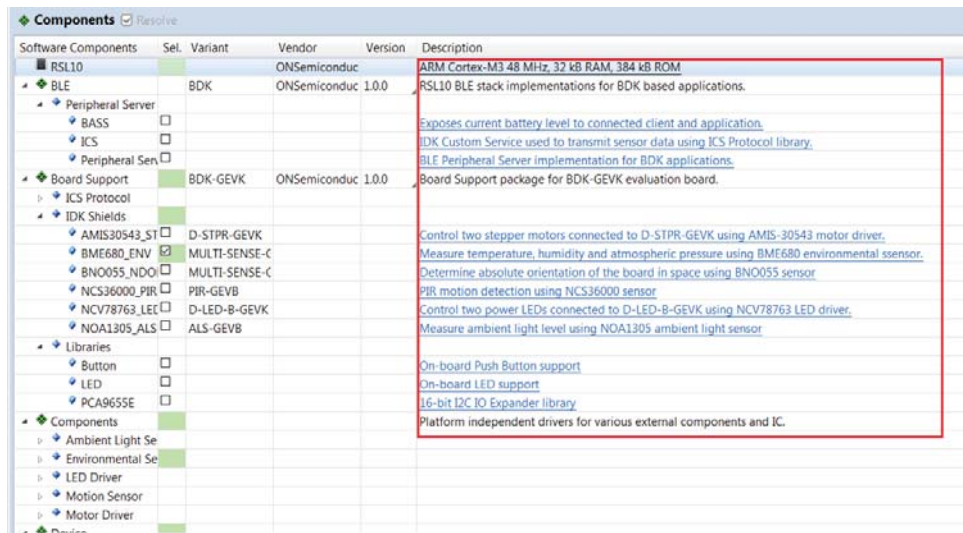
DOCUMENTATION

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 as described in the B_IDK CMSIS Software Organization section. To access the components, double click *.rteconfig file. Extensive help is provided under the description tab.



EVBUM2589/D

ON Semiconductor  **BDK** v1.0.0
Bluetooth LE Development Kit for RSL10

BN0055 Absolute Orientation Sensor
Part # 93081-1000010

Absolute orientation sensor library (accelerometer, gyroscope, magnetometer). More...

Data Structures

```

struct BN0055_NDOF_CalStatus
  BN0055 calibration status structure. More...
struct BN0055_NDOF_Resources
  
```

MACROS

```

#define BN0055_NDOF_IOEXP_ADDRESSES (0x48 << 1)
  I2C address of IO expander on Multisensor anted.
#define BN0055_NDOF_IOEXP_PORT (1)
  IO expander port containing BN0055 related signals.
#define BN0055_NDOF_IOEXP_RST_PIN (0)
  IO expander pin number for BN0055 (reset) signal.
#define BN0055_NDOF_IOEXP_RST_PIN_MASK (1 << BN0055_NDOF_IOEXP_RST_PIN)
#define BN0055_NDOF_IOEXP_INT_PIN (1)
  IO expander pin number for BN0055 Interrupt signal.
#define BN0055_NDOF_IOEXP_INT_PIN_MASK (1 << BN0055_NDOF_IOEXP_INT_PIN)
  
```

Enumerations

```

enum BN0055_NDOF_PowerMode { BN0055_NDOF_POWER_MODE_NORMAL = 0, BN0055_NDOF_POWER_MODE_LOW_POWER = 1, BN0055_NDOF_POWER_MODE_SUSPEND = 2 }
  Available power modes of BN0055. More...
  
```

Functions

```

int32_t BN0055_NDOF_initialize(void)
  Initializes the BN0055 and sets it into Nine Degrees of Freedom (NDOF) operation mode. More...
int32_t BN0055_NDOF_setPowerMode(enum BN0055_NDOF_PowerMode mode)
  Allows to set chips power mode to reduce current consumption or disable sensors. More...
int32_t BN0055_NDOF_GetCalibrationStatus(struct BN0055_NDOF_CalStatus *status)
  Reads calibration status of BN0055 sensors. More...
int32_t BN0055_NDOF_ReadLinearAccel(struct bn0055_accel_float_t *ps)
  Reads latest linear acceleration vector in MSQ from device. More...
int32_t BN0055_NDOF_ReadGravity(struct bn0055_gravity_float_t *ps)
  Reads latest gravity vector in MSQ from device. More...
int32_t BN0055_NDOF_ReadAngRotation(struct bn0055_gyro_float_t *ps)
  Reads latest angular rotation vector in DPS from device. More...
int32_t BN0055_NDOF_ReadAbsOrientation(struct bn0055_euler_float_t *ps)
  Reads latest absolute orientation vector in degrees from device. More...
  
```

Run Time Environment Configuration

These parameters are part of the RTE_BN0055_NDOF RTE configuration file and can be used to adjust library behavior. This file is copied into the Eclipse project when the BN0055_NDOF component is selected and can be edited by using the CMSIS Configuration Wizard editor.

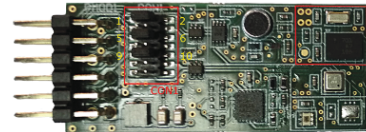
```

#define RTE_BN0055_NDOF_EXT_CLK_SRC 1
  
```

Detailed Description

Absolute orientation sensor library (accelerometer, gyroscope, magnetometer).

The BN0055 is a System in Package integrating a triaxial accelerometer, a triaxial gyroscope, a triaxial geomagnetic sensor and 32 bit microcontroller.



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

EVBUM2589/D



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.

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)
```

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

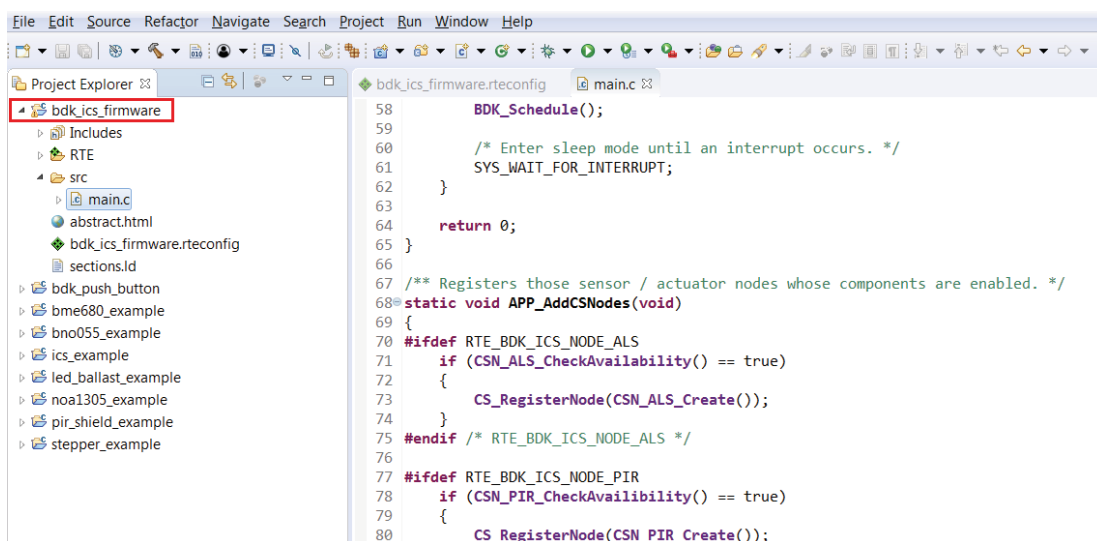
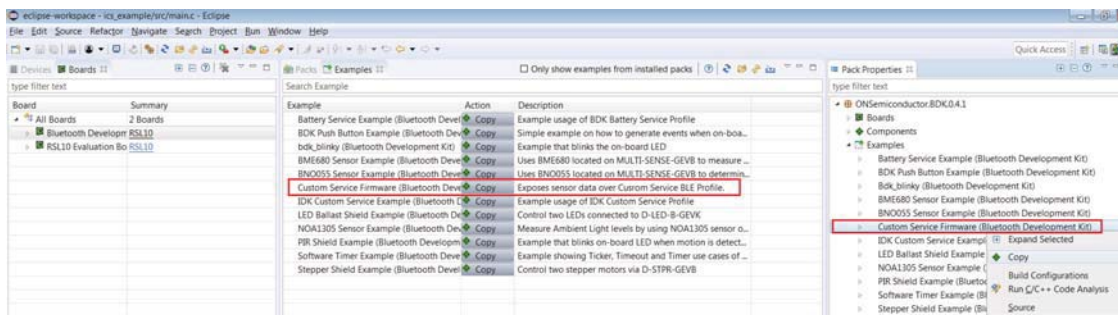
EVBUM2589/D



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.

Custom Service Firmware

In order to read sensor data and control actuators connected to the BDK-GEVK from the RSL10 Sense and Control mobile app, the Custom Service Firmware must be downloaded onto the BDK-GEVK. This firmware can be found as Custom Service Firmware under examples in the CMSIS pack.



EVBUM2589/D

Bluetooth is a registered trademark of Bluetooth SIG

ON Semiconductor and the ON Semiconductor logo are trademarks of Semiconductor Components Industries, LLC dba ON Semiconductor or its subsidiaries in the United States and/or other countries. ON Semiconductor owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of ON Semiconductor's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marking.pdf. ON Semiconductor 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 as such not available for sale to consumers. The board is only intended for research, development, demonstration and evaluation purposes and should be used only 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 delivered "AS IS" and without warranty of any kind including, but not limited to, that the board is production-worthy, that the functions contained in the board will meet your requirements, or that the operation of the board will be uninterrupted or error free. ON Semiconductor expressly disclaims all warranties, express, implied or otherwise, including without limitation, warranties of fitness for a particular purpose and non-infringement of intellectual property rights.

ON Semiconductor 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 ON Semiconductor shall not constitute any representation or warranty by ON Semiconductor, and no additional obligations or liabilities shall arise from ON Semiconductor having provided such information or services.

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. Should you purchase or use the board for any such unintended or unauthorized application, you shall indemnify and hold ON Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that ON Semiconductor was negligent regarding the design or manufacture of 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 ON Semiconductor 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.

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

LIMITATIONS OF LIABILITY: ON Semiconductor 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 ON Semiconductor is advised of the possibility of such damages. In no event shall ON Semiconductor'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.

For more information and documentation, please visit www.onsemi.com.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:

Literature Distribution Center for ON Semiconductor
19521 E. 32nd Pkwy, Aurora, Colorado 80011 USA
Phone: 303-675-2175 or 800-344-3860 Toll Free USA/Canada
Fax: 303-675-2176 or 800-344-3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800-282-9855 Toll Free
USA/Canada
Europe, Middle East and Africa Technical Support:
Phone: 421 33 790 2910

ON Semiconductor Website: www.onsemi.com

Order Literature: <http://www.onsemi.com/orderlit>

For additional information, please contact your local Sales Representative