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

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	Х	Programming and Power over USB
Х	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.

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 6.32f 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: <u>https://github.com/ARM-software/CMSIS_5/releases</u>
- 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: <u>https://github.com/ARM-software/CMSIS-FreeRTOS/releases</u>

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

Importing CMSIS Packages

1. Launch the RSL10 SDK 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.
 - 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.



Compiling and Flashing

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

eclipse-workspace - Eclip	se				e é
<u>File Edit Navigate Search</u>	n <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp				
🖻 🕶 🗟 🕼 🗎 🖨 🕶 🗖	े 🐂 २ 🐸 २ 🖬 💁 🗸 🥯	• [金 • ◎ • ◎ • ◎ •			Quick Access
E Devices Boards 🛙	🗄 🖻 🕐 🗖 🗖	触 Packs 📑 Examples 😫		🗆 Only show examples from installed packs 🕲 🍣 🐸 🖑 😐 🗢 🗉	🖩 Pack Properties 😫 🕢 🐨 🐨
type filter text		Search Example			type filter text
Board	Summary	Example	Action	Description	 BONSemiconductor.BDK.0.4.1
🔺 🍄 All Boards	2 Boards	Battery Service Example (Bluetooth Devel	Copy	Example usage of BDK Battery Service Profile	Boards
Bluetooth Develops	rr <u>RSL10</u>	BDK Push Button Example (Bluetooth Dev	💠 Copy	Simple example on how to generate events when on-boa	Components
RSL10 Evaluation B	o RSL10	bdk_blinky (Bluetooth Development Kit)	Copy	Example that blinks the on-board LED	 Examples
		BME680 Sensor Example (Bluetooth Deve	💠 Сору	Uses BME680 located on MULTI-SENSE-GEVB to measure	 Battery Service Example (Bluetooth Development Kit)
		BNO055 Sensor Example (Bluetooth Deve	Copy	Uses BNO055 located on MULTI-SENSE-GEVB to determin	 BDK Push Button Example (Bluetooth Development Kit)
		Custom Service Firmware (Bluetooth Deve	Copy	Exposes sensor data over Cusrom Service BLE Profile.	 Bdk_blinky (Bluetooth Development Kit)
		IDK Custom Service Example (Bluetooth E	Copy	Example usage of IDK Custom Service Profile	 BME680 Sensor Example (B Collapse Solosted
		LED Ballast Shield Example (Bluetooth De	Copy	Control two LEDs connected to D-LED-B-GEVK	ONSemiconductor::Bluet
		NOA1305 Sensor Example (Bluetooth Dev	Copy	Measure Ambient Light levels by using NOA1305 sensor o	BNO055 Sensor Example (B Copy
		PIR Shield Example (Bluetooth Developm	🕈 Copy	Example that blinks on-board LED when motion is detect	 Custom Service Firmware (Bluetooth Development Kit)
		Software Timer Example (Bluetooth Deve	Copy	Example showing Ticker, Timeout and Timer use cases of	 IDK Custom Service Example (Bluetooth Development Kit)
		Stepper Shield Example (Bluetooth Devel	Copy	Control two stepper motors via D-STPR-GEVB	 LED Ballast Shield Example (Bluetooth Development Kit)
					 NOA1305 Sensor Example (Bluetooth Development Kit)
					 PIR Shield Example (Bluetooth Development Kit)
					 Software Timer Example (Bluetooth Development Kit)
					 Stepper Shield Example (Bluetooth Development Kit)

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



2. 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).

3. 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 the debug configurations as shown below.



4. Double click **GDB Segger J–Link Debugging** to create the debug configuration for the selected example.

Debug Configurations	
Create, manage, and run configurations	
Image: Second Secon	Name: pir_shield_example Debug Main Debugger Project: pir_shield_example Debugypir_shield_example Browse C/C++ Application: Debugypir_shield_example.elf Debugypir_shield_example.elf Variables Build (if required) before launching Browse Build Configuration Select Automatically © Enable auto build O Disable auto build @ Use workspace settings Configure Workspace Settings_

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

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

Debug Configurations	
Create, manage, and run configurations	ta de la companya de
Image: Second Sec	Name: pir_shield_example Debug Main © Debugger) Startup) Source Common J-Link GDB Sever Setup Start the J-Link GDB server locally Connect to running target Executable: S(jlink_path)/S(jlink_gdbserver) Browse Variables Actual executable: C/Program Files (x86)/SEGGER/JLink_V634c//JLinkGDBServerCLexe (to change it use the global or workspace preferences pages or the project properties page) Device name: RSL10 Supported device names Endianness: © Little Big Connection: @ USB P (USB serial or IP name/address) Interface: © SWD JTAG Initial speed: Auto SWO port: 2331 Verify downloads Initialize registers on start
Filter matched 10 of 10 items	Re <u>v</u> ert Apply
0	Qebug Close

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



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

Compiling and Flashing

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

eclipse-workspace - Eclipse			- Ö
<u>File Edit Navigate Search Project Run Window H</u> elp			
🖆 🕶 🔛 🐘 🕲 🕶 🖼 😓 🦂 🖮 🥔 🕶 💁 🕶 🚱 🖉	• () • () • () • () • () • ()		Quick Access 🛛 😰 🛛 🗄
🖩 Devices 🗃 Boards 🕴 🖷 🖻 🕲 🙀 😤 🖶 🗖	🛳 Packs 📑 Examples 😫	🗆 Only show examples from installed packs 🕐 ಿ 🐸 🖑 🔤 🌣 🖻	🗏 Pack Properties 🛿 🕢 🐨 🐨
type filter text	Search Example		type filter text
Board Summary	Example Action	Description	ONSemiconductor.BDK.0.4.1
All Boards 2 Boards	Battery Service Example (Bluetooth Devel 🔷 Copy	Example usage of BDK Battery Service Profile	Boards
Bluetooth Developm <u>RSL10</u>	BDK Push Button Example (Bluetooth Dev 🗢 Copy	Simple example on how to generate events when on-boa	A Components
RSL10 Evaluation Bo RSL10	bdk_blinky (Bluetooth Development Kit) 🔅 Copy	Example that blinks the on-board LED	Examples
	BME680 Sensor Example (Bluetooth Deve 💠 Copy	Uses BME680 located on MULTI-SENSE-GEVB to measure	 Battery Service Example (Bluetooth Development Kit)
	BNO055 Sensor Example (Bluetooth Deve Scopy	Uses BNO055 located on MULTI-SENSE-GEVB to determin	 BDK Push Button Example (Bluetooth Development Kit)
	Custom Service Firmware (Bluetooth Deve Copy	Exposes sensor data over Cusrom Service BLE Profile.	 Bdk_blinky (Bluetooth Development Kit)
	IDK Custom Service Example (Bluetooth E 🍫 Copy	Example usage of IDK Custom Service Profile	BME680 Sensor Example (B Collapse Selected
	LED Ballast Shield Example (Bluetooth De 🔶 Copy	Control two LEDs connected to D-LED-B-GEVK	ONSemiconductor:Bluet
	NOA1305 Sensor Example (Bluetooth Dev 🕏 Copy	Measure Ambient Light levels by using NOA1305 sensor o	BNO055 Sensor Example (B Copy
	PIR Shield Example (Bluetooth Developm 🕏 Copy	Example that blinks on-board LED when motion is detect	 Custom Service Firmware (Bluetooth Development Kit)
	Software Timer Example (Bluetooth Deve 💠 Copy	Example showing Ticker, Timeout and Timer use cases of	 IDK Custom Service Example (Bluetooth Development Kit)
	Stepper Shield Example (Bluetooth Devel 🕸 Copy	Control two stepper motors via D-STPR-GEVB	 LED Ballast Shield Example (Bluetooth Development Kit)
			 NOA1305 Sensor Example (Bluetooth Development Kit)
			 PIR Shield Example (Bluetooth Development Kit)
			 Software Timer Example (Bluetooth Development Kit)
			 Stepper Shield Example (Bluetooth Development Kit)

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



2. Right click and build the project. This creates binaries to be flashed to BDK-GEVK.

눱 Project Explorer 🛛		□ 😫 😜 🗸 🗆 🗖	<pre></pre>	example.rteconfig 🔒 main.c 😫
ics_example			19//	
 Epir_shield_example Enaries Enaries Debug Enclude Enclude	1	New Go Into Open in New Window Show in Local Terminal Copy Ctrl+C Paste Ctrl+V Delete Delete		right (c) 2018 Semiconductor Components Industries LLC /a "ON Semiconductor"). All rights reserved. software and/or documentation is licensed by ON Semiconductor under ted terms and conditions. The terms and conditions pertaining to the ware and/or documentation are available at ://www.onsemi.com/site/pdf/ONSEMI_T&C.pdf ("ON Semiconductor Standard s and Conditions of Sale, Section 8 Software") and if applicable the ware license agreement. Do not use this software and/or documentation ss you have carefully read and you agree to the limited terms and
	8			
	24 24 8	Mo <u>v</u> e Rena <u>m</u> e	F2	e <stdio.h> e <bdk.h></bdk.h></stdio.h>
		Import Exp <u>o</u> rt		<pre>e <buk_components.h> e "main.h"</buk_components.h></pre>
			CMSIS C/C++ Project Build Project	,
		Clean Project Refresh Clo <u>s</u> e Project Close <u>U</u> nrelated Projects	F5	32_t retval = 0; Initialize BDK library, set system clock to 8MHz. */ Initialize();
		Build Targets Index Build Configurations	> >	Initialize LED */ _LED_Init();
		Show in Remote Systems view		<pre>ntf("\r\n\nBDK - PIR Motion Detection Example.\r\n\n");</pre>

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

3. 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.



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

	*
Debug Configurations	
Create, manage, and run configurations	100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100 - 100
Image: Image	Name: pir_shield_example Debug Imain the Debugger Imain the Startup the Source Image Common Broject: pir_shield_example Debugypir_shield_example Browse C/C++ Application: Debugypir_shield_example.elf Build (if required) before launching Build Configuration: Build Configuration: Search Project Build Configuration: Select Automatically Image:

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

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

Cebug Configurations	
Create, manage, and run configurations	The second se
Image, and full configurations Image, and full configurations Image, and full configurations Image, and full configurations Image: Image configuration Image: Image: Image configuration Image: Image: Image configuration Image: I	Name: pir shield_example Debug Main * Debugger > Startup * Source Common J-Link GDB Server Setup Start the J-Link GDB server locally Connect to running target Executable: Siglink_gath/\$(jink_gdbserver) Recutable: Siglink_gath/\$(jink_gdbserver) Recutable: C/Program Files (x86)/SEGGER//Link_V634c//LinkGDBServerCLexe (to change it use the global or workspace preferences pages or the project properties page) Device name: RSL10 Supported device names Findianness: Uittle Big Connection: USB Interface: SWD OB port: 2331 SWO port: 2332 Werify downloads Initialize registers on start
Filter matched 10 of 10 items	Reyert Apply
0	Debug Close

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

,

retva.	Confirm Perspective Switch
ialize	This kind of launch is configured to open the Debug perspective when it suspends.
:ialize _Init()	This Debug perspective is designed to support application debugging. It incorporates views for displaying the debug stack, variables and breakpoint management.
["\r\n\I	Do you want to open this perspective now?
:ialize = NCS30 DEBUG(1 0_PIR_0	<u>Remember my decision</u> <u>Yes</u>

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

eclipse-workspace - pir_shield_example/src/main.c - Eclipse				
Eile Edit Source Refactor Navigate Search Project Run Window Help				
🗂 ▾ 💷 🐘 🏔 🔍 ▼ 😐 ¥ (💽)) 🔳 🕫 🌫 🦘 🕫 🕼 🖛 🗮 🏹 🕹 🏪 巻 ▼ 0 ▼ 9. ▼ 🥙 😂 🖋 ▼ 🥒 🕬	§ • § • ♥> � •	⇔ ▼		Quick #
₩ Debug 🛙 Resume (F8)	🍇 i 🕶 🗖	🕬 Variables 🖾 💁 Break	points 🕮 Registers 🛋 Modules 🐕 Periphe	erals 🐇 🖻
Ic pir_shield_example Debug [GDB SEGGER J-Link Debugging]		Name	Type	Value
4 🔐 pir_shield_example.elf		00• retval	int32 t	-536813312
 Phread #1 57005 (Suspended : Breakpoint) 				
main() at main.c:22 0x10025a				
JLinkGDBServerCL.exe				
🚚 arm-none-eabi-gdb				
📕 Semihosting and SWV				
		4		
le main.c 🖾 📧 0x0				🗄 Outline 🖾 📄 🖓 🔌
11 // conditions. By using this solutione and/or documentation, you agree to the			*	stdio.h
12 // Timited terms and conditions.				BDK.h
14 #include <stdio.h></stdio.h>				BDK_Components.h
15 #include <bdk.h></bdk.h>				main.h
16 #include <bdk_components.h></bdk_components.h>			=	main(void) : int
17				PIR_MotionEvent(void) : void
18 #include "main.h"				
200 int main(void)				
21 {				
<pre>22 int32_t retval = 0;</pre>				
23				
24 /* Initialize BDK library, set system clock to 8MHz. */				
25 BDK_Initialize();			-	
4			F.	
🖾 Console 💈 🔄 Tasks 🙎 Problems 🕡 Executables 🍠 Terminal 🗟 Debugger Console 📋 Memory				- • × 🔆 🗟 🖬 🖗 🗩 🗷
pir shield example Debug [GDB SEGGER J-Link Debugging] JLinkGDBServerCL.exe				
Read 2 bytes @ address 0x001029B8 (Data = 0x4D3C)				
Read 4 bytes @ address 0x00102AAC (Data = 0x001053D4)				
Read 2 bytes @ address 0x001029BA (Data = 0x483D)				
Kead 4 bytes @ address 0x00102AB2 (Data = 0x00042000)				
Read 2 bytes @ address 0x001029DL (Data = 0x4295)				
Read 4 bytes @ address 0x00102AB6 (Data = 0x53D02000)				
Read 2 bytes @ address 0x001029C0 (Data = 0xD908)				

Logging/Debugging

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

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

This section provides instructions for each of the above options.

Using Eclipse Console

1. Click the Open a Terminal Icon

eclipse-workspace - Source not found Eclipse	
<u>File Edit Navigate Search Project Run Window H</u> elp	
i 🗂 ➡ 📰 🐚 📾 i 👁 📲 💷 > 💷 > 2. ⊙ .e. i⇒ =	≅ ∞ 🕹 🐂 🛠 ▾ 🔿 ▾ 🂁 ▾ 🗀 🖋 ▾ 🕅
★ Debug ⊠ Open a Terminal (Ctrl+Alt+Shift+T)	
Dir_shield_example Debug [GDB SEGGER J-Link Debugging]
🔺 🥐 pir_shield_example.elf	
Thread #1 57005 (Running : User Request)	
JLinkGDBServerCL.exe	
📕 arm-none-eabi-gdb	
🔎 Semihosting and SWV	•

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

Caunch Terminal		
Choose terminal:	Telnet Terminal 🔹	
Settings		
Hosts:	▼ 🗶	
Host:	localhost	
Port:	19021 -	
Timeout (sec):	5	
End of Line:	CR+NUL 🔻	
Encoding: Default (ISO-8859-1)		
?	OK Cancel	

C eclipse-workspace - Source not found Eclipse	
File Edit Navigate Search Project Run Window Help	
* ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	
☆ Debug 🛛 🙀 😾 マ 🗖 🗖	(x)= Varial
 Image: Image: Ima	Name
』》 LinkGDSServerCLexe 』 arm-none-eabi-gdb 』 SemiNusting and SWV	
	4
Imain. © Oxdeadbeee 33 Break at address "Oxdeadbeee" with no debug information available, or outside of program code. View Disassembly Configure when this editor is shown Preferences	
□ Console ④ Tasks 🖹 Problems ④ Executables 🖉 Terminal 🛱 🔂 Debugger Console	
□ Tenetiocahost(%27/18.850 AM) ⊗ SEGGER J-Link V6.34c - Real time terminal output J-Link OB-SAM3U128 V3 compiled Jul 12 2018 12:17:50 V3.0, SN=483035634 Process: JLinkGDBserverCL.exe PIR: Motion detected at 36862 ms. PIR: No motion at 370868 ms. PIR: No motion at 518577 ms. PIR: No motion at 518573 ms.	

PIR: No motion at 520783 ms. PIR: Motion detected at 521608 ms. PIR: No motion at 523814 ms.

Using Eclipse Serial Console via UART-SPI Bridge.

When you do not want to use the Segger RTT viewer as serial console, the BDK–GEVK board is equipped with UART–SPI uC AX8052F100 flashed with special firmware, taking care of the entire serial communication with values returned on Terminal.

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

	2 × & ∰ @ ▼ @ ▼ @	• •	* * • • • • •	• 😕 🗁 🛷 🕶	BUE	[월 ▼ 禄 ▼ � � ▼ ↔ ▼				
Project Explorer 🛛 👘	bme680_example.rteco	onfig 8	3						- 0	85
🖻 😫 🕼	Components 🗹 R	♦ Components 🖸 Resolve							0	4.0
b bdk_ics_iirmware	Software Components	Sel	Variant	Vendor	Version	Description				An
 Diffeoso_example Pinaries 	RSL10			ONSemiconduc		ARM Cortex-M3 48 MHz, 32 kB F	AM, 384 kB ROM			
Includes	BLE		BDK	ONSemiconduc	1.1.1	RSL10 BLE stack implementation	s for BDK based applicatio			
Pebug	Board Support		BDK-GEVK	ONSemiconduc	1.1.1	Board Support package for BDK	-GEVK evaluation board.			
include	Components					Platform independent drivers for	r various external componei			
🖻 🍅 RTE	A Intervice									
Src	🔺 🎐 BDK					BDK software components comp	patible with all evaluation b		=	
abstract.html	AES			ONSemiconduc	1.1.1	AES module from mbedTLS				
bme680_example.rteconfig	Event Callba	ck⊠		ONSemiconduc	1.1.1	Library for assigning of multiple	callbacks to events.			
sections.ld	HAL			ONSemiconduc	1.1.1	Peripheral HAL drivers and RSL1	0 configuration			
	Output Redi	re 🗹	SPI Bridge Axem	ONSemiconduc	1.1.1	Redirects standart output calls u	sing SPI to UART bridge on			
	Scheduling		SEGGER RTT	VSemiconduc	1.1.1	Management layer for Event Ker	nel Application Task			
	Software Tir	ne 🖾	SPI Bridge Axem	NSemiconduc	1.1.1	Allows to create multiple timer e	events while using only sing			
	Bluetooth Profi	les	UART						-	
	Validation Output			Description						
	Components Device Pac Problems Tasks C CDT Build Console [bme6 'Finished building 'Invoking: GNU ARM	ks Conse B0_exa targe Cross	ble ⊠ □ Properties mple] •t: bme680_example.e : Create Flash Image	\$ î	5] I 6		EmbSys Registers 원 G 든 ERROR: Please select Register	a chip using the pre Hex	eference pag Bin	je (c+

4. When the project runs, Click the **Open a Terminal** Icon.

🗢 eclipse-workspace - Source not found Eclipse	
<u>File Edit Navigate Search Project Run Window</u>	w <u>H</u> elp
🖻 🗕 🖩 🕼 📾 - 🌘 🔍 🕨 💷 🛤 3. 🤊	≥? i→ ≂ ∞ & 🐂 ☆ ▾ O ▾ 🂁 ▾ 🗀 🛷 ▾ 🖗
☆ Debug 🛛 Open a Terminal (Ctrl+Alt	t+Shift+T)
pir_shield_example Debug [GDB SEGGER J-Link	Debugging]
# Ze pir_shield_example.elf	
Thread #1 57005 (Running : User Request)	
JLinkGDBServerCL.exe	
📕 arm-none-eabi-gdb	
Semihosting and SWV	0

5. Enter the appropriate COM port as shown below, and launch the session. The incoming events are entered on the terminal window.

🖨 Launch Terminal 📃 🗖 💌					
Choose term	inal: Serial Terminal 🔹				
Settings					
Serial port:					
Baud rate:	115200 🗸				
Data size:	8 🗸				
Parity:	None				
Stop bits:	1				
Encoding:	Default (ISO-8859-1)				
?	OK Cancel				

eclipse-workspace - Source not found Eclipse		
File Edit Navigate Search Project Run Window Help		
🖆 🕶 🔛 🔞 📾 🗣 🖳 🔍 🕨 💷 🖉 🌫 👁 💵 🕬 🗮 🖉 🔸 🖉 🔸 🖉 🔸	• [월 • 전 • ♥ ♥ • ♥ •	
拳 Debug ⊠	🍇 😥 🗢 🗖	(x)= Varia
 E pir_shield_example Debug [GDB SEGGER J-Link Debugging] Pir_shield_example.eff P Tread #1 57005 (Running : User Request) JInkGDBServer(Lexe 		Name
🗸 arm-none-eabi-gdb		
Semihosting and SWV		
🖻 main.c 📧 Oxdeadbeee 😂		
Streak at address 'toxeeabbeee' with no debug information available, or outside of program code. View Disassembly		
Console VI Tasks 🗄 Problems 🛈 Executables 🖉 Terminal 🛛 🗤 Debugger Console		
W Hemet localnost (6/2/1/b 620 MAN) & Second State Lement localnost (6/2/1/b 620 MAN) & SEGGER 3-1/hit V6.34c - Real time terminal output J-Link V6.34c - Real time terminal 0.12 2018 12:17:50 V3.0, SN=483035634 Process: JLinkGDBServerCL.exe PIR: Notion detected at 368662 ms. PIR: Notion detected at 3518577 ms. PIR: Notion detected at 518577 ms. PIR: Notion detected at 521608 ms. PIR: Notion at 522783 ms. PIR: Notion at 522814 ms.		

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



Using J–Link RTT

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



7. Select USB and click OK.

J-Link RTT Viewer V6.34c		
<u>File Terminals</u> Input Log	🔜 J-Link RTT Viewer V6.34c Configuration 🛛 💦 💌]
Log All Terminals Terr LOG: J-Link RTT Vie LOG: Terminal 0 add	Connection to J-Link USB Serial No Existing Session Specify Target Device Contact file (actional)	
	Target Interface & Speed	
	SWD 4000 kHz	
	RTT Control Block Auto Detection <u>A</u> ddress <u>Search Range</u>	
	OK Cancel	Enter Clear
Ready.		0.00 MB

8. 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.

🔜 J-Link RTT View	wer V6.34c	X
<u>File</u> <u>T</u> erminals	Input Logging Help	
Log All Termi	nals Terminal 0	
LOG: J-Link LOG: Termin	: RTT Viewer V6.34c: Logging started. al 0 added.	
🔜 J-L	ink V6.34c Device Selection	83
0	The selected device "UNSPECIFIED" is unknown to this version of the J-Link software. Please make sure that at least the core J-Link shall connect to, is selected. Proper device selection is required to use the J-Link internal flash loaders for flash download or unlimited flash breakpoints. For some devices which require a special handling, selection of the correct device is in <u>OK</u>	iportant.
	E	nter Clear
Ready.	Establishing J-Link connection 0.00 MB	

SEGGER J-Link V6.34c - Target device settings						×		
Filter Manufacturer *	•	Device RSL10		Core *	•	Little	e endian e #0	▼ ▼
Manufacturer	Device		Cor	re	NumCo	Flash size	RAM size	
ON Semicond	RSL10		Cor	tex-M3 r2p1	1	390 KB	24 KE	}
								_
Select a device f	or J-Link.							
Selecting a device is not required for most devices, but allows more efficient operation of J-Link as well as flash download,modification of flash memory during a debug session as well as unlimited presknoints in flash memory (Flash Breaknoints)								
In case of doubt, select the first entry in the list. "Unspecified Device".								

J-Link RTT Viewer V6.34c	- • ×
<u>File</u> <u>I</u> erminals Input Logging <u>H</u> elp	
Log All Terminals Terminal 0	
LGG: SWD speed too high. Reduced from 2667 kHz to 1800 kHz fo LGG: Scanning AP map to find all available APs LGG: AP[1]: Stopped AP scan as end of AP map has been reached LGG: AP[0]: AHB-AP (IDR: 0x24770011) LGG: Iterating through AP map to find AHB-AP to use LGG: AP[0]: Care found LGG: CPUID register: 0x412FC231. Implementer code: 0x41 (ARM) LGG: FEUnit: 2 code (BP) slots and 0 literal slots LGG: CoreSight components: LGG: ROWTb1[0] (D: E000E000, CID: B105E00D, PID: 000BB000 SCS LGG: ROWTb1[0] [2]: E0000200, CID: B105E00D, PID: 002BB002 DWT LGG: ROWTb1[0] [2]: E0002000, CID: B105E00D, PID: 002BB003 FFB LGG: RTT Viewer connected.	r stability
	Enter Clear
RTT Viewer connected. 0.000	MB



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.

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



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



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

	🖨 Launch Terminal	
	Choose terminal: Serial Terminal Settings Serial port Baud rate: 115200 Data size: 8 Parity: None Stop bits: 1 Encoding: Default (ISO-8859-1)	
	⑦ OK Cancel	
eclipse-workspace - Source not found file Edit Navigate Search Project f	l Eclipse Run Window Help	
⇒ 🗑 🐘 🔍 🗮 🕬 🖛	■ № 3. ©	· · · · · · · · ·
 E pir_shield_example Debug (GDB SE pir_shield_example.elf Pired #1 57005 (Running : U UintGDServerCLexe arm-none-eabi-gdb Semihosting and SWV 	GGER J-Link Debugging] Iser Request)	Name
main c C Ovdeadheee X		•
Break at address "Dxdeadbeee" with no view Disassembly	debug information available, or outside of program code.	

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	v.			Board support package for BDK-GEVK evalution board
		Libraries			Board specific libraries
			LED	1	On-board LED support
			Button	1	On-board push button support
			PCA9655E	1	16-bit I2C IO Expander library
		IDK Shields		-	Support for Arduino / PMOD extension boards
			PIR-GEVB	1	PIR Motion detection using NCS36000
			ALS-GEVB	1	Measure Ambient light levels using NOA1305 ambient light sensor
			MULTI-SENSE-GEVB	rev2.1	Combines 3 sensors: BME680, BNO055, NOA1305
			BLDC-GEVK		
			D-LED-B-GEVK	1	
			D-STPR-GEVK	1	
		ICS Protocol		-	Libraries that allow connected BLE devices to take control over sensors / actuators using ICS Service.
			System Node]	Protocol implementation and sytem node used by other sensor / actuator nodes.
			PIR Node		Exposes motion data provided by NCS36000 from PIR-GEVB
			ALS Node		Exposes ambient light levels measured by NOA1305 from ALS-GEVB
			ENV Node		Exposes environmental data measured by BME680 from MULTI-SENSE-GEVB
			AO Node		Exposes absolute orientation measured by BNO055 from MULTI-SENSE-GEVB
			STPR Node		Allows remote control of two stepper motors connected to D-STPR-GEVB.
			LEDB Node		Allows remote control of two power LEDs connected to D-LED-B-GEVK
			BLDC Node		Allows to remote control BLDC motor connected to BLDC-GEVB.
	Components				Platform independent software drivers for controlling of various external IC.
		LED Driver			
			NCV78763	J	Dual LED Driver and Power Ballast, for Automotive Front Lighting, 1.6 A, 2nd Generation
		Ambient Light Sensor	10.1.005	1	
			NOA1305]	Ambient Light Sensor with I2C Interface and DarkCurrent Compensation
		Motor Driver	AMIC 2054	1	Miner standing stands doi: 1.000 interface (as bindes stands and as
			AIVII 5-3054	-	Nicro-stepping stepper motor driver with SP1 Interface for bipolar stepper motors
		Environmental Sensor	LV090/ UVV]	Sensoriess Three-phaseDrushiess DC MotorController, with GateDrivers, for Automotive
		L IIVII UIIII einai Sensui	hme680	1	l ow nowar dae, praesura, temperatura & humidity sensor
		Motion sensor	51110000	1	Low power gas, pressure, temperature & numberly sensor
			bno055	1	Intelligent 9-axis absolute orientation sensor
		Touch Sensor	2110000	1	
			LC717A00AR	1	Capacitance-Digital-Converter for Electrostatic Capacitive Touch Sensors
	Device				
		BDK			
			HAL]	RSL10 Peripheral abstraction layers for BDK applications.
			Scheduling		Event Kernel wrapper for BDK applications.
			Software Timer		Allows to create multiple timer events while using only single hardware timer.
			Event Callback		Library for executing multiple event handlers when an event occurs.
			Output Redirection		Redirects standard library output calls (printf,) to specified channel
				SEGGER RTT	Output is transmitted using UART peripheral
				UART	Output is transmitted over SWD using the on-board or external J-LINK deug probe
2214			AES]	
BDK	BLE	Destational Com	1		
		Peripheral Server	Dattern Occurren	1	Foregan and the theory buyed to a second and a first and a section the
			Dattery Service	-	Exposes current battery level to connected client and application.
			Device Service		IDK Custom Service used to transmit sensor data using ICS Protocol library.
			r enpheral Server		DEE Peripretal Server implementation for DDN applications.

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
- ◆ B ONSemiconductor.BDK.1.0.0

 ▲ Boards
 ■ Components
 ▶ Onsemiconductor.BDK-GEVK

 ◆ Components
 ▶ Board Support
 ▶ Motor Driver
 ▶ Motor Driver
 ▶ Motor Driver
 ▶ Motor Driver
 ▶ LED Driver
 ▶ Elic Driver
 ▶ Elic Driver
 ▶ Motion Sensor
 ▶ Packages
 ■ Sense Sense
 ■ Sen

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.

<u>File Edit Source Refactor Navigate Search Project Run</u>	Window Help
🖆 🕶 💷 🐚 🥹 🕶 🍕 🕶 📾 i 🕥 🕶 i 🖾 🕶 🚳	3 ▼ 🕄 ▼ 🮯 ▼ 👫 ▼ 🔾 ▼ 🧏 ▼ 🧣 ▼ 🤔 🗁 🖋 ▼ 🗟 📾 🗐 🗊 🍠 🆃 🗄 🐨 🎒 ▼ 🏷 🗢 ▼ ↔ ▼
💫 Project Explorer 🛛 🕒 🖨 🎭 🔻 🗖 🗖	a main.c ∞
▲ 😂 noa1305 example	19//
Includes	2 // Copyright (c) 2018 Semiconductor Components Industries LLC
include	3 // (d/b/a "ON Semiconductor"). All rights reserved.
🔺 🏝 RTE	4 // This software and/or documentation is licensed by UN Semiconductor under
🔺 🗁 BDK	6 // software and/or documentation are available at
BDK_PushButton.c [ONSemiconductor::BDK	7 // http://www.onsemi.com/site/pdf/ONSEMI_T&C.pdf ("ON Semiconductor Standard
BDK_Task.c [ONSemiconductor::BDK.Librari	8 // Terms and Conditions of Sale, Section 8 Software") and if applicable the
BDK.c [ONSemiconductor::BDK.Libraries.Co	9 // software license agreement. Do not use this software and/or documentation
List EventCallback.c [ONSemiconductor::BDK.Lit	10 // conditions. By using this software and/or documentation, you agree to the
B HAL_CIOCK.C [UNSemiconductor::BDK.Librar	12 // limited terms and conditions.
B HAL DC c [ONSemiconductor: BDK Librario	13 //
HAL SPLC [ONSemiconductor:BDK.Libraries	14 #include <stdio.h></stdio.h>
HAL UART.c [ONSemiconductor::BDK.Librat	16 #include "BDK Components.h"
HAL.C [ONSemiconductor::BDK.Libraries.Co	17
I2C_RSLxx.c [ONSemiconductor::BDK.Librari	18 #include "main.h"
ROA1305_ALS.c [ONSemiconductor::BDK.C	19 200 internitional
Image:	200 Int main(Void)
PCA9655E.c [ONSemiconductor::BDK.Comp	22 int32_t status = 0;
RTE_BDK.h [ONSemiconductor::BDK.Librar]	23
RTE_PCA9655E.h [ONSemiconductor::BDK.C =	24 BDK_Initialize();
RTE_SoftwareTimer.h [ONSemiconductor::B	<pre>20 26 printf("\r\nAPP: NOA1305 ALS example.\r\n"):</pre>
George Batt a Consemicanductor:Bl	27
SoftwareTimer c [ONSemiconductor:BDK.Lib]	<pre>28 status = NOA1305_ALS_Initialize();</pre>
SPL RSL xx c [ONSemiconductor::BDK Librari	<pre>29 ASSERT_DEBUG(status == 0);</pre>
stimer.c [ONSemiconductor::BDK.Libraries.S	31 status = NOA1305 ALS StartContinuous(APP ALS PERIODIC INTERVAL MS.
syscalls_segger_rtt.c [ONSemiconductor::BD	<pre>32 &ALS_ReadCallback);</pre>
USART_RSLxx.c [ONSemiconductor::BDK.Lib	<pre>33 ASSERT_DEBUG(status == 0);</pre>
 RTE BDK BDK_PushButton.c [ONSemiconductor: BK_CONSemiconductor:BDKLibrarie BDK.c [ONSemiconductor:BDKLibrarie BDK.c [ONSemiconductor:BDKLibrarie HAL_clock.c [ONSemiconductor:BDKLib HAL_clock.c [ONSemiconductor:BDKLib HAL_SPLc [ONSemiconductor:BDKLib HAL_CIONSemiconductor:BDKLib HAL_CIONSemiconductor:BDKLib HAL_CIONSemiconductor:BDKLib HAL_CIONSemiconductor:BDKLib HAL_CIONSemiconductor:BDKLibrarie R HAL_ONSEmiconductor:BDKLibrarie R HAL_ONSEmiconductor:BDKLibrarie R HAL_ONSEmiconductor:BDKLibrarie R HAL_ONSEmiconductor:BDKLibrarie 	BDK 5 // limited terms and conditions. The terms and condition 6 // software and/or documentation are available at 7 // http://www.onsemi.com/site/pdf/ONSEMI_T&C.pdf ("ON Sen 8 // Terms and Conditions of Sale, Section 8 Software") and 9 // software license agreement. Do not use this software 10 // unless you have carefully read and you agree to the li 11 // conditions. By using this software and/or documentati 12 // limited terms and conditions. 13 // traries 14 #include <stdio.h> 15 #include <bdk_h> 16 #include "BDK_components.h" 17 in.h"</bdk_h></stdio.h>
▷ <u>k</u> NOA <u>O</u> pen	(b)
Open With	C/C++ Editor
Show in Local Terminal	CMSIS Configuration Wizard
	Ctrl+C Generic Text Editor
Daste	Ctrl+V Editor
▷ BRTE_ Bolata	Delete
Electric SEGC Sector Contraction	(n");
Kemove from Context	Ctri+Ait+Snitt+Down I In-Frace Editor
Source	
Move	Other
Rename	F2 NOA1305 ALC StantContinuous (ADD ALC DEDTODI)
	ReadCallback)
🦻 📷 sysca 🔤 Import	ERIG(status -= 0);
🛛 🛃 USAI 🖆 Exp <u>o</u> rt	ebod(status == 0);

A brief description on the header files is given below.

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.

otion	Valu	Je
SYSCLK Frequency	8 M	Hz
APP Task Event Kernel message handler cour		48 MHz
HAL Pinmap Configuration		24 MHz
Board selection		16 MHz
Custom Pinmap		8 MHz
USART0_TX Pin	2	
USART0_RX Pin	4	
SPI0_MOSI Pin	7	
SPI0_MISO Pin	10	
SPI0_SSEL Pin	5	
SPI0_SCLK Pin	6	
I2C0_SCL Pin	1	
SPI1_SDA Pin	0	
I2C0 DIO Low Pass Filter	ENA	BLED
I2C0 DIO Drive Strength	6X	
I2C0 DIO Pull Selection	No	pull
LED Pin	14	
Button Pin	15	

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.

otion	Value
Software Timer Configuration	
Timer resolution [us]	100
Hardware Timer Select	TIMER1
	TIMER0
	TIMER1
	TIMER2
	TIMER3

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.

≔ RTE_PCA9655E.h 🛛								
CMSIS Configuration Wizard								
Option	Value							
Enable PCA9655E shared interrupts								
Interrupt signal DIO Pad	13							
DIO Interrupt Source	0							
EventCallback event ID	1234							

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.

CMSIS Configuration Wizard					
ption	Val	Je			
Stepper Shield Left Channel					
Step Mode	1/4	Micro - Step			
Coil Peak Current	245	mA			
Direction Of Rotation	CW	motion			
NXT Edge Trigger	Risi	ng 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					
Enables jittery PWM					
Steps Per Revolution	200				
Stepper Shield Right Channel					
Step Mode	1/4	Micro - Step			
Coil Peak Current		1 / 32 Micro - Step	L		
Direction Of Rotation		1 / 128 Micro - Step	L		
NXT Edge Trigger		1 / 64 Micro - Step	L		
Turn On / Off Slopes of Motor Driver		Compensated Full Step, 2 phase on	L		
Speed Load Angle Transparency Bit		Compensated Full Step, 1 phase on	L		
Speed Load Angle Gain		1 / 16 Micro - Step	l		
Enables doubling of the PWM frequency		1 / 8 Micro - Step	l		
Enables jittery PWM	1 / 4 Micro - Step				
Steps Per Revolution		Compensated Half Step	l		
•		Uncompensated Half Step	ŀ		
iten Mode		Uncompensated Full Step	F		

≔ RTE_NCV78763_LED.h 🛛				
CMSIS Configuration Wizard				
Option	Value			
Enable Booster				
Booster PWM generation	Internal			
Booster PWM Frequency	242 kHz			
Booster Clock Inversion				
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				
Booster Skip Clock Cycles	Disabled			
Enable Buck Regulator Channel 1				
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				
Comparator Threshold Voltage	0			
Tunes the Toff x VLED value for channel 1	0			
> Overcurrent Settings				
> Enable Buck Regulator Channel 2				
General Settings				
Thermal warning threshold	0			
LED sampling duration selection	88			
	•. •			
Booster Overvoltage Reactivation Defines the hysteresis for the reactivation once t Default: -1 V for D-LED-B-GEVK	he overvoltage shutdown is triggered.			
Source Editor CMSIS Configuration Wizard				

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.



Copyright (c) 2018 Semiconductor Components Industries, LLC (d/b/a ON Semiconductor).

*.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.

Software Components	Sel.	Variant	Vendor	Version	Description
RSL10			ONSemiconduc		ARM Cortex-M3 48 MHz, 32 kB RAM, 384 kB ROM
🔺 💠 BLE		BDK	ONSemiconduc	1.0.0	RSL10 BLE stack implementations for BDK based applications.
Peripheral Serve	er				
BASS					Exposes current battery level to connected client and application.
ICS					IDK Custom Service used to transmit sensor data using ICS Protocol library.
Peripheral Se	n 🗆				BLE Peripheral Server implementation for BDK applications.
A Soard Support		BDK-GEVK	ONSemiconduc	1.0.0	Board Support package for BDK-GEVK evaluation board.
ICS Protocol					a ··· · -
IDK Shields					
AMIS30543_S	т	D-STPR-GEVK			Control two stepper motors connected to D-STPR-GEVK using AMIS-30543 motor driver.
BME680_ENV		MULTI-SENSE-C			Measure temperature, humidity and atmospheric pressure using BME680 environmental ssensor
BNO055_ND		MULTI-SENSE-C			Determine absolute orientation of the board in space using BNO055 sensor
NCS36000_PI	R	PIR-GEVB			PIR motion detection using NCS36000 sensor
NCV78763_LE		D-LED-B-GEVK			Control two power LEDs connected to D-LED-B-GEVK using NCV78763 LED driver.
NOA1305_AL	s 🗆	ALS-GEVB			Measure ambient light level using NOA1305 ambient light sensor
Libraries					
Button					On-board Push Button support
🖗 LED					On-board LED support
PCA9655E					16-bit I2C IO Expander library
Components					Platform independent drivers for various external components and IC.
Ambient Light S	e				
Environmental S	e				
LED Driver					
Motion Sensor					
Motor Driver					
🔺 📤 Dovico					

ON Sewiconductor ⁴ OR BLC V1.0.0 Bluetooth LE Development Kit for RSL10
BN0055 Absolute Orientation Sensor
Absolute orientation ennorr library (accelerometer, gyroscope, magnetometer). More
Data Structures
EV.dt BNOSS_INDF_Califatus
should all and a should be and
INVALUUS solgen Ringers NIDE In PRE ADDRESS, MUCES ADDRESS, MUCES ADDRESS, MUCES ADDRESS, MUCES ADDRESS, ADDRESS, MUCES
12C address of 10 expander on Multisensor shield.
Refere NotoSS_UCOF_UCCEP_PORT (1) O equater for containing SUCOSS related signal.
Redne BNOSS, NOOF JOER, RT PH (0) 10 oppinde minute for BNOSS service Service Service
Rethe BN0055 NDOF_ICERP_RAT_RIN_MASK (1 << BN0005_INDOF_ICERP_RAT_RIN)
Restine BMORS, MOCF_JOCER, JMT_FIN (1) I O equation for number for Discost Interrupt alignal.
Reache BNOSS_NOCF_DEXE_NT_PRI_MALK (1 < BNOSS_NOCF_DEXE_NT_PRI)
Enumerations
erum BNOSS, NDOF, Powerfloods (BNOSS, NDOF, POWER, MODE, NORMAL = 0, BNOSS, NDOF, POWER, MODE, LOW POWER = 1, BNOSS, NDOF, POWER, MODE, SUSPEND = 2) Available power modes of BNOSSS. More
Functions
Ind2_1 Blocks_IDD2_Indiating rold Indiates the RAVESS and use It to Nan Dearses of Freedom IND2F1 costation mode. More
Intit2 1 BNOSS_NDCP_Membranes in the object on Teaching (note) - you provide interaction and the object of the obj
In32_1 BN0655_IND0F_GelCalibration Status (etucd BN0655_IND0F_Caliblus *status) Reads calibration status of BN0555 emotion. More
Int21 E IND059, INDOF Read Limited Accord (Nucle Mode) [Inter_accord_Net] Read Limited Terra Accordation Version Mode.
Int2_1 (MOX69, INCO/F_NeedGravity (Work MOX655, gava, 5xd, 1*p) Reas Usies gava/ work in KS6 more index. Moze.
Int2_E IRX0555_INCOF_RestAngRedBalon (intruct broSS_grav_bad_1*#) Read- tables 2 and 2 and 2 and 2 and 3 and
Int32 1 ENC055 NICOF_ReadbanCrientation (struct bnc055 exiler_Inot (1 ptr) Reads latest absolute ortentation vector in degrees from device. More
Run Time Environment Configuration
These parameters are part of the RTE_BNOSS_NOOP.h RTE configuration file and can be used to adjust library behavior. This file is copied into the Estipse project when the BNOSS_NOOP component is exceeded and can be eated by using the CARS Configuration Witten's editor.
#sefine RTE_BNOSS5_NDOF_EXT_CLK_SRC 1
Detailed Description
Absolute orientation sensor library (accelerometer, gyroscope, magnetometer).
ine chruppe a system in vacage integrang a traxea speeroneer, a traxea georagneo sentor and 22 of microcontroler.

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			ONSemiconduc		ARM Cortex-M3 48 MHz, 32 kB RAM, 384 kB ROM
▷ 💠 BLE		BDK	ONSemiconduc	1.0.0	RSL10 BLE stack implementations for BDK based applications.
Board Support		BDK-GEVK	ONSemiconduc	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			ONSemiconduc	1.0.0	AES module from mbedTLS
Event Callback			ONSemiconduc	1.0.0	Library for assigning of multiple callbacks to events.
🕈 HAL			ONSemiconduc	1.0.0	Peripheral HAL drivers and RSL10 configuration
Output Redirection		SEGGER RTT	ONSemiconduc	1.0.0	Redirects standart output calls using SEGGER RTT
Scheduling			ONSemiconduc	1.0.0	Management layer for Event Kernel Application Task
Software Timer			ONSemiconduc	1.0.0	Allows to create multiple timer events while using only single hardware timer.
Bluetooth Profiles					
Libraries					
Startup		release	ONSemiconduc	2.1.10	RSL10-CMSIS Startup Library and Include Folders (libcmsis)

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.

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

Periphera	al Hardware Abstraction Layer for RSL10. More
Nodul	es
Cloc Defin	k Configurations tes possible clock configurations for proper operation of BDK.
12C 12C i	nterface for communication with connected shields.
SPI i	nterface for communication with connected shields.
UAR UAR	T T interface for communication with connected shields.
Macro	s

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

CN 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.

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.

eclipse-workspace - ice	_example/src/main.c - Eclipse					- 0
Eile Edit Source Refac	tor <u>N</u> avigate Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u>	(indow Help				
	🖻 👌 🐐 🍣 🐸 🌭 🛶 🧶 👄	⋪ • [⊿ ə [일 • 8] • 10 4 • 0 •				Quick Access 🔡 😰 🛛 🗟
E Devices E Boards 😂	E E 2 🐐 🐃 🗆	📾 Packs 📑 Examples 😫		🛛 Only show examples from installed packs \mid 🕲 🍣 🦑	🖕 🔍 🗖	💷 Pack Properties 🛙 🕀 🗁 🐨 👘
type filter text		Search Example				type filter text
Board	Summary	Example Actio	tion De	escription		ONSemiconductor.BDK.0.4.1
🔺 🔩 All Boards	2 Boards	Battery Service Example (Bluetooth Devel 💠 Co	Copy Ex	xample usage of BDK Battery Service Profile		Boards
Bluetooth Development Development	opm RSL10	BDK Push Button Example (Bluetooth Dev 🗢 Co	Copy Sir	imple example on how to generate events when on-boa		A Components
RSL10 Evaluation Bo RSL10		bdk_blinky (Bluetooth Development Kit) 💠 C	Copy Ex	xample that blinks the on-board LED		🔺 📑 Examples
		BME680 Sensor Example (Bluetooth Deve 💠 Co	Copy Us	ses BME680 located on MULTI-SENSE-GEVB to measure		 Battery Service Example (Bluetooth Development Kit)
		BNO055 Sensor Example (Bluetooth Deve 🔶 Co	Copy Us	ses BNO055 located on MULTI-SENSE-GEVB to determin		 BDK Push Button Example (Bluetooth Development Kit)
		Custom Service Firmware (Bluetooth Deve 💠 Co	Copy Ex	xposes sensor data over Cusrom Service BLE Profile.		 Bdk_blinky (Bluetooth Development Kit)
		IDK Custom Service Example (Bluetooth E 🍫 Co	Copy Ex	xample usage of IDK Custom Service Profile		 BME680 Sensor Example (Bluetooth Development Kit)
		LED Ballast Shield Example (Bluetooth De 🍫 Co	Copy Co	ontrol two LEDs connected to D-LED-B-GEVK		 BNO055 Sensor Example (Bluetooth Development Kit)
		NOA1305 Sensor Example (Bluetooth Dev 🍫 Co	Copy Me	leasure Ambient Light levels by using NOA1305 sensor o		 Custom Service Firmware (Bluetooth Development Kit)
		PIR Shield Example (Bluetooth Developm 🔶 C	Copy Ex	xample that blinks on-board LED when motion is detect		IDK Custom Service Exampl 🛞 Expand Selected
		Software Timer Example (Bluetooth Deve 🔶 G	Copy Ex	kample showing Ticker, Timeout and Timer use cases of		LED Ballast Shield Example Copy
		Stepper Shield Example (Bluetooth Devel 🔶 Co	Сору Со	ontrol two stepper motors via D-STPR-GEVB		NOA1305 Sensor Example (PIR Shield Example (Bluetor Software Timer Example (B Software Timer Example (B Software Timer Example (B)
						Stepper Shield Example (Bli Source



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 as such 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 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: Email Requests to: orderlit@onsemi.com

TECHNICAL SUPPORT North American Technical Support:

ON Semiconductor Website: www.onsemi.com

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



Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;

- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);

- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;

- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком):

- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «JONHON», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «FORSTAR».



«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).



Телефон: 8 (812) 309-75-97 (многоканальный) Факс: 8 (812) 320-03-32 Электронная почта: ocean@oceanchips.ru Web: http://oceanchips.ru/ Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А