

RSL10 Smart Shot Color Camera Evaluation Board User's Manual

Event-Triggered AI Imaging Platform,
Bluetooth® Low Energy Enabled



ON Semiconductor®

www.onsemi.com

EVBUM2803/D

Description

The RSL10 Smart Shot Color Camera (SECO-RSL10-CAM-COLOR-GEVK) is a complete ultra-low-power, event-triggered, node-to-cloud platform that enables smart image capture for portable, low duty cycle IoT applications including asset monitoring and event-triggered cameras. Based on the RSL10 SIP and the ARX3A0 CMOS image sensor (Color and Mono), the platform features multiple trigger modes including periodic intervals, motion detection, and environmental sensor triggers (e.g., changes in temperature and humidity).

The camera kit includes the following components:

- Baseboard PCB
- [ARX3A0 Color 121° DFOV IAS Module \(IAS1MOD-ARX3A0CSSC090110-GEVB\)](#)
- [ARX3A0 Mono 121° DFOV IAS Module \(IAS1MOD-ARX3A0CSSM090110-GEVB\)](#)
- Micro USB Cable

By default, the color ARX3A0 IAS module is mounted onto the PCB. To use the mono IAS module, swap the module onto the baseboard PCB. The firmware supports both module, and is capable of automatically identifying the modules.

The RSL10 SIP serves as the processing hub of the camera, and Bluetooth® Low Energy (Bluetooth LE) connectivity enabling remote control and transfer of captured image and sensor data to the provided *RSL10 Smart Shot* mobile app (available on GooglePlay™, iOS®).

The mobile app is used to control the camera via BLE interface, set trigger conditions for events, capture and store the images captured by the camera, and finally act as a gateway to provide node-to-cloud connectivity. Connected to an Image Analytics/AI engine, located in the cloud, the app is capable of detecting and identifying objects using an AI cloud service. The mobile application returns the captured image, along with the list of identified objects and confidence levels.

EVAL BOARD USER'S MANUAL

The compact design form-factor of the camera can be used for developing compact cameras with 360 fps mono imaging based on the ARX3A0 CMOS image sensor. ARX3A0 offers high sensitivity at both visible and NIR wavelength, and has a super low power mode that consumes 3.2 mW when active. Additionally, ARX3A0 can be programmed to monitor specific zones of its field of view, automatically taking an image when the scene's contents change. Embedded on the platform hardware, the *SPCV1100A* from SunplusIT® provides multiple color processing features and JPEG image compression.

The impressive ultra-low-power operation of the RSL10 SIP and ARX3A0 is complemented by a dedicated FAN53880 Power Management IC (PMIC) and additional smart power management modes implemented in hardware. Other devices enabling the efficient power management are the NCP705 low Iq current linear dropout regulator and the NCP1421 ultra-Low Iq boost regulator, when camera board supplied from primary battery. Thanks to these features, the RSL10 Smart Shot Camera supports an extended battery lifetime of 5+ years – depending on the specific use case.

Key ON Semiconductor Parts used:

- [RSL10 SIP](#)
- [IAS1MOD-ARX3A0CSSC090110-GEVB](#)
- [IAS1MOD-ARX3A0CSSM090110-GEVB](#)
- [LE25U20AQG](#)
- [FAN53880](#)
- [NCP705](#)
- [NCP1421](#)
- [NCP3901](#)
- [FPF1003A](#)

Applications

- Asset Management
- Smart Home/Buildings
- Smart Agriculture
- Smart Retail

Key Features

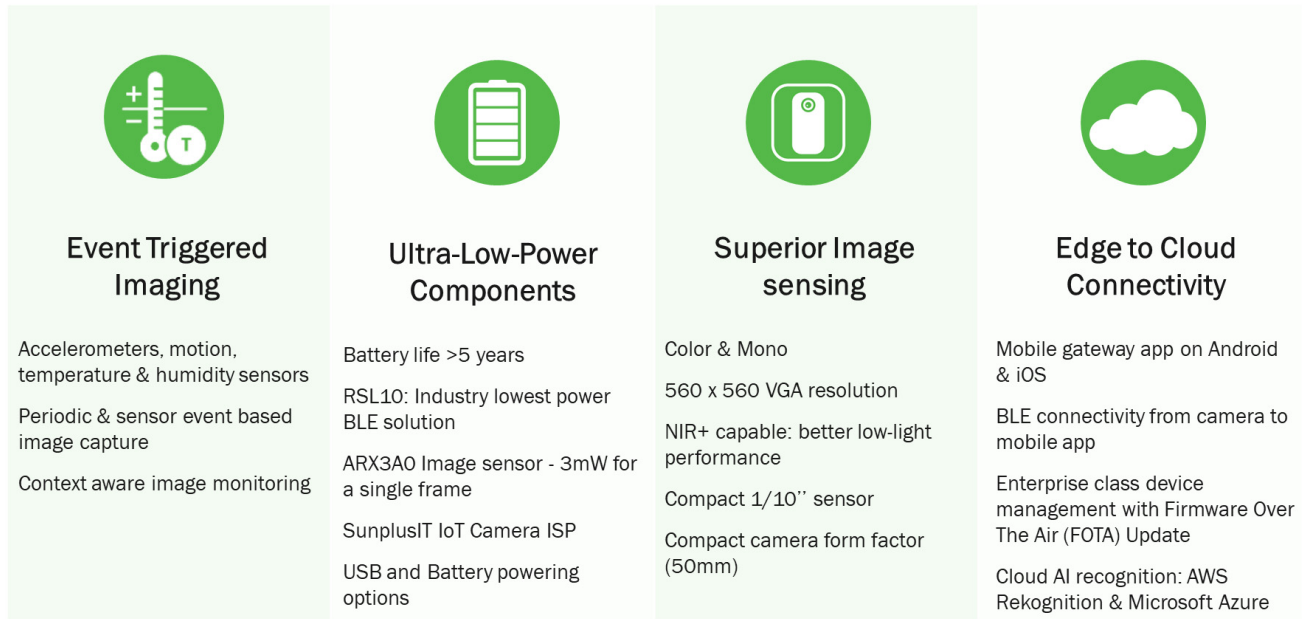


Figure 1. Key Features

System Overview

Figure 2 depicts the system architecture of the RSL10 Smart Shot Camera.

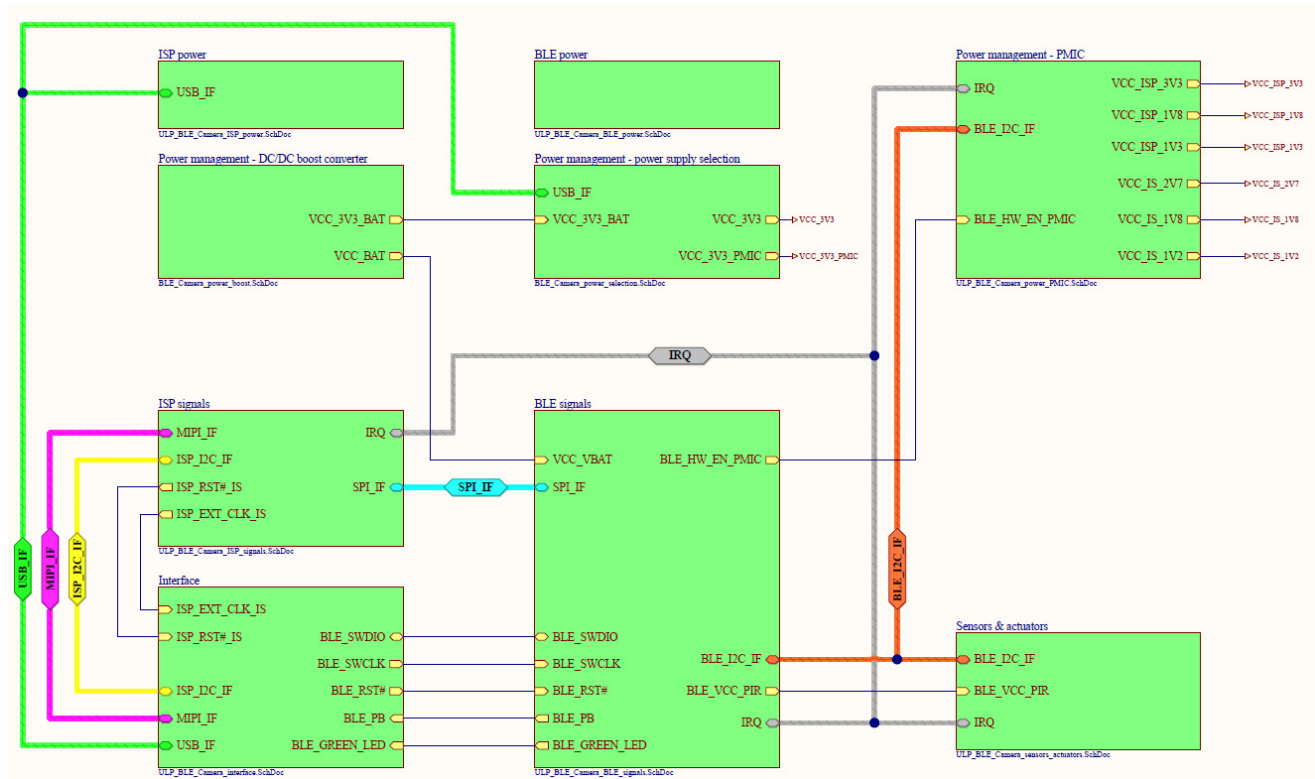
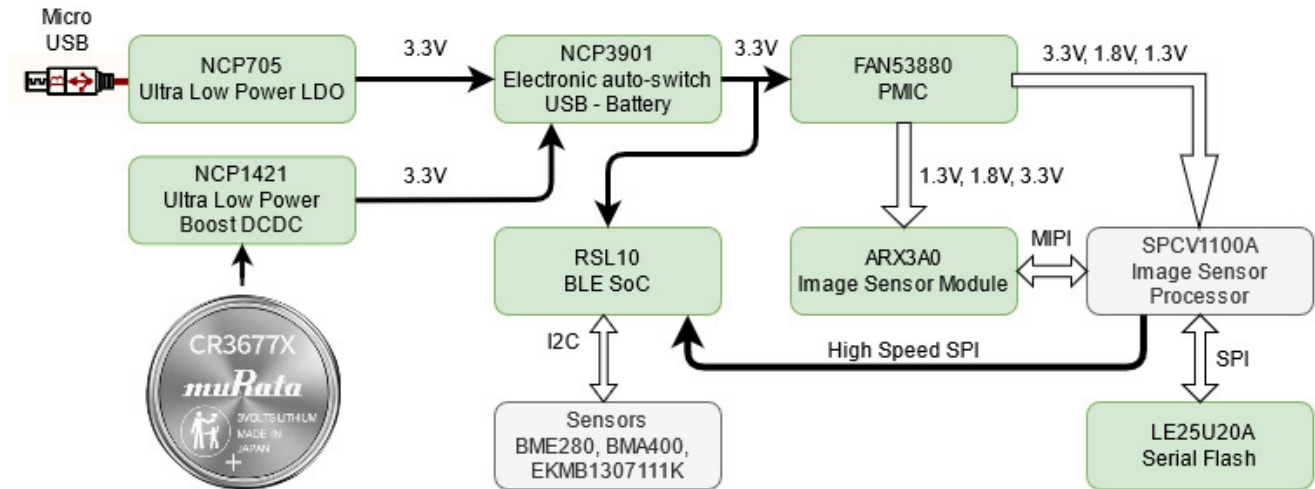


Figure 2. RSL10 Smart Shot Camera System Architecture

The platform consists of the following major blocks:

- RSL10 SIP Bluetooth Low Energy System in Package (SiP)
- ARX3A0 Color and Mono 121° DFOV IAS Modules
- SPCV1100A Image Sensor Processor from Sunplus IT
- Environmental Sensors from Bosch Sensortec– BMA400 (accelerometer), BME280 (humidity, temperature), EKMB1307111K (motion)
- Power Management – NCP1421 (low power boost DCDC), NCP705 (low power LDO), FAN53880 (PMIC), NCP3901 (electronic switch)

The entire system is powered by a unique supply architecture that delivers optimized system performance and low power consumption. The board can be powered by a USB cable or Murata Primary 3V battery CR3677x. A Li battery can be added with battery protection circuitry as described further in this text.

Whether a USB cable or Lithium battery, the NCP3901 will identify the input source and select the correct supply source for operation. In case of USB supply operation, NCP705 serves for step down conversion from 5V down to 3.3V – used for RSL10 and PMIC supply. For battery operated mode, NCP1421 boost DCDC converter outputs 3.3V.

NCP3901 is the electronic low power switch that selects between the USB or battery supply operation. Priority is given to take power from USB in case of both battery and USB supply are connected to the system.

For high-performance image processing, the system features the color and mono ARX3A0 with SPCV1100A

Image Sensor Processor (ISP) from Sunplus IT. Interconnectivity between the ARX3A0 camera module and SPCV1100A ensures a high speed MIPI interface. Due to the short peak power delivery needed to process the image in SPCV1100A and overall system power consumption savings, supply voltage rails for proper operation of ISP and camera modules ensures ultra low power PMIC FAN53880, featuring highly efficient multi-rail power delivery with typical few hundreds of nA current consumption when switched of programmatically.

A high-speed SPI interface between the SPCV1100A and the RSL10 SIP is used transfer the captured and processed image over Bluetooth Low Energy. More information about this process, including the mobile app, is described in Software development kit *ON Semiconductor RSL10-SmartShot_BSP.2.1.0+1*. All components and their placement on the camera platform is shown in Figure 3.

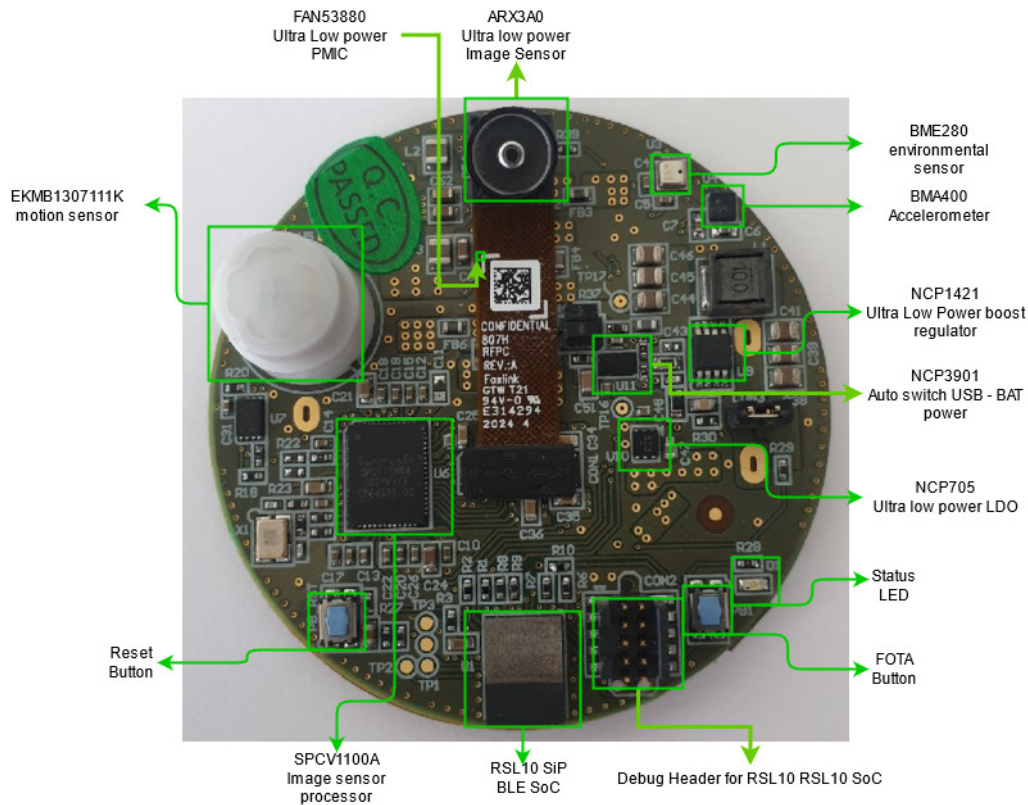


Figure 3. RSL10 Smartshot Camera Components

Block Diagram

Figures 4 and 5 depict the detailed block diagram for both power management part and signaling chain within the system.

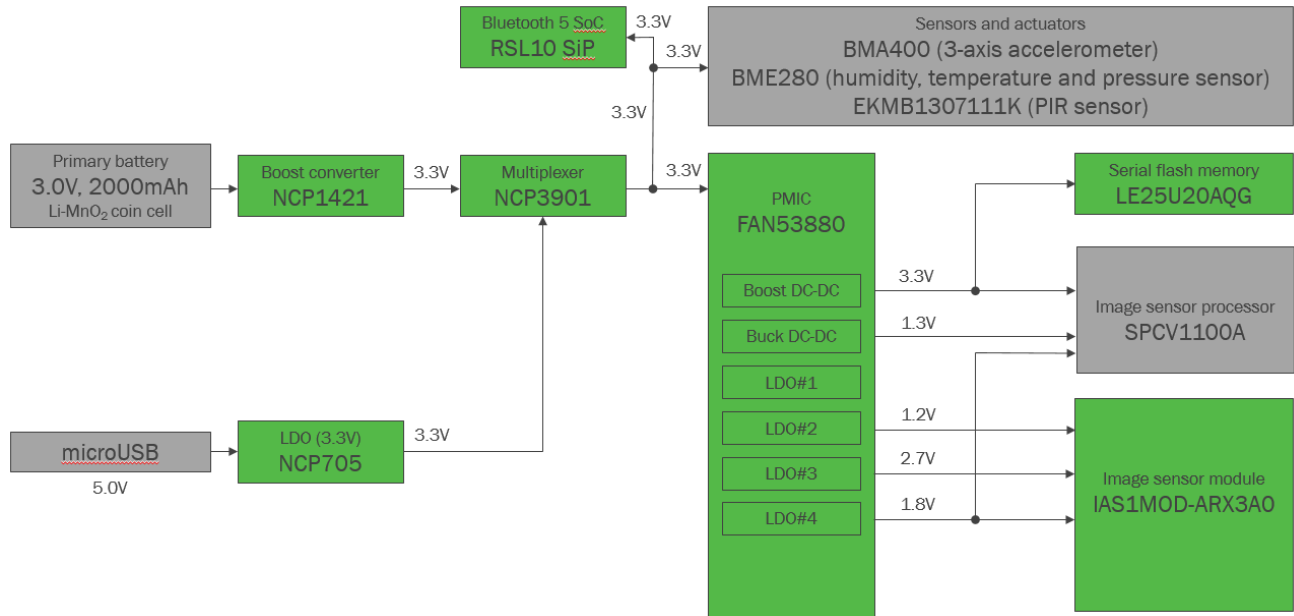


Figure 4. RSL10 Smart Shot Camera Block Diagram (Power Management)

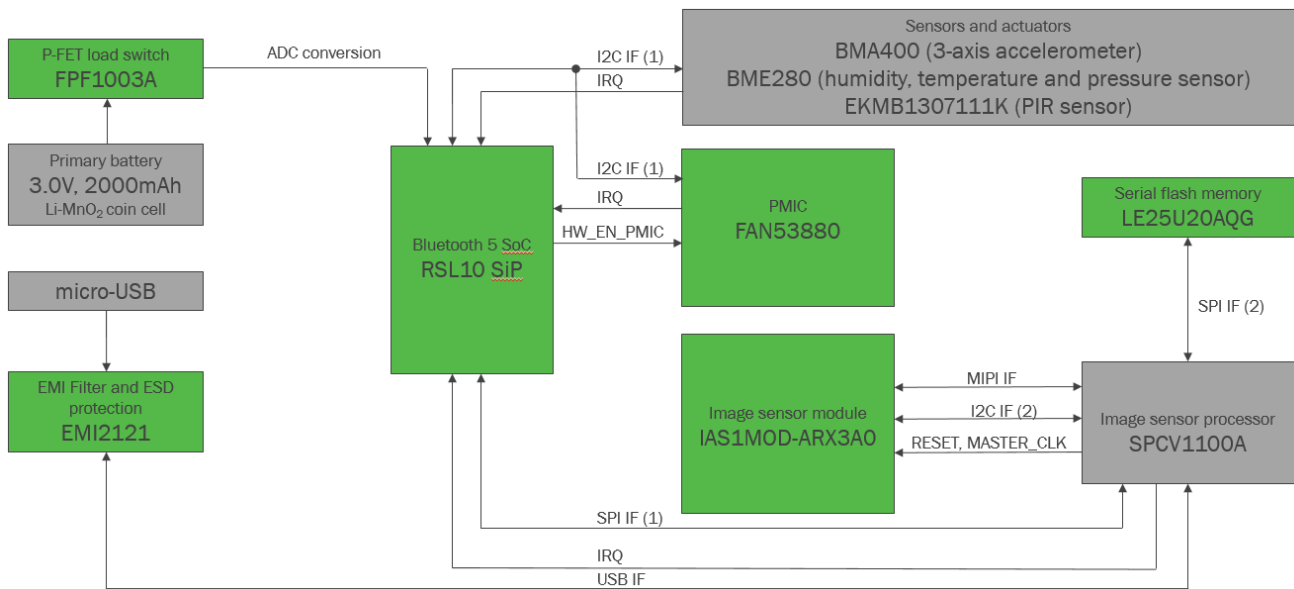


Figure 5. RSL10 Smart Shot Camera Block Diagram (Signals)

Connecting the Rechargeable Battery

For standalone operation, the Smart Shot Camera can be adapted for Primary battery connection. In order to mount the battery on the bottom of the PCB, ON Semiconductor bundled sticker to avoid any potential shorts despite battery is attached on the standoffs. CR3677x is not bundled in the

design, please contact Murata for further information. Here is the procedure when battery mounted onto the board:

1. Remove the sticker from paper and put on bottom side of the PCB
2. Insert the battery into the PCB and solder (Note: this primary, non-rechargeable battery)



Figure 6. Process of Attaching the Sticker onto the Bottom Side of the PCB and Battery Insertion

RSL10 Smart Shot Camera Software Development Environment

The RSL10 Smart Shot Camera is available with a complete software development environment and source code for the RSL10 SIP, including documentation explaining how the sensor data and image transfer services are defined and executed inside RSL10. This chapter describes installation prerequisites and instructions, features and capabilities of the package, and how to modify the source code.

For getting started using the kit, refer the “Getting Started Document”.

Installation Prerequisites

Follow these prerequisites in order to successfully install *ON Semiconductor.RSL10-SmartShot_BSP.2.1.0+1* CMSIS package:

- Install 64-bit version of Java from <https://www.java.com/en/download/>

- Install J-Link Version 6.32i or later from <https://www.segger.com/downloads/jlink> (select J-Link software and documentation pack)
- Download and install “ON Semiconductor IDE Installer” from [RSL10: Radio SoC, Bluetooth® 5 Certified](#)
- Download the latest RSL10 CMSIS pack (baseline SDK) under “RSL10 Software Package” and follow the installation steps in “RSL10 SDK Getting Started Guide” from above website. Save this CMSIS pack in a folder, for example, C:\cmsis_packs
- Download and install ARM.CMSIS (due to CMSIS driver for SPI) – [Download](#)
- Download and install the *ON Semiconductor.RSL10-SmartShot_BSP.2.1.0+1* CMSIS-Pack to discover and modify the RSL10 SoC source code with complete software documentation attached to it.

Importing the CMSIS-Packs

Once all packs are successfully imported, they can be viewed in the CMSIS pack manager perspective as shown below (Figure 7).

Search Pack			type filter text
Pack	Action	Description	
<ul style="list-style-type: none"> Device Specific <ul style="list-style-type: none"> ON Semiconductor.BDK <ul style="list-style-type: none"> Up to date Bluetooth Development Kit Device Support and ON Semiconductor.RSL10 <ul style="list-style-type: none"> Up to date ON Semiconductor RSL10 Device Family Pack ON Semiconductor.RSL10-ATAG_BSP <ul style="list-style-type: none"> Up to date ON Semiconductor RSL10 Asset Tracking Tag B ON Semiconductor.RSL10-SmartShot_BSP <ul style="list-style-type: none"> Up to date ON Semiconductor RSL10 SmartShot Boards S Generic <ul style="list-style-type: none"> 3 Packs Software Packs with generic content not specifi ARM.CMSIS <ul style="list-style-type: none"> Up to date CMSIS (Cortex Microcontroller Software Interfa 			<ul style="list-style-type: none"> ON Semiconductor.RSL10-SmartShot_BSP.2.1.0 <ul style="list-style-type: none"> Boards Components Examples <ul style="list-style-type: none"> Smartshot_bootloader (SECO-RSL10-CAM-GEVB) Smartshot_demo_cam (SECO-RSL10-CAM-GEVB) Smartshot_demo_color_cam (SECO-RSL10-CAM-COLOR-GEVK) Packages

Figure 7. Imported CMSIS Packages from Prerequisites including ON Semiconductor.RSL10-SmartShot_BSP.2.1.0+1 (Board Support Package)

After installing the RSL10 Smart Shot CMSIS Pack, the display as *ON Semiconductor.RSL10-Smartshot_BSP*. The RSL10 Smart Shot CMSIS Pack consists of three sample codes that are bind not only to SECO-RSL10-CAM-COLOR-GEVK, but also to its

predecessor SECO-RSL10-CAM-GEVB platform. To have the source code and all related documentation visible, and right click on the selected sample code (Figure 8) to imports the project into the Project Explorer workspace.

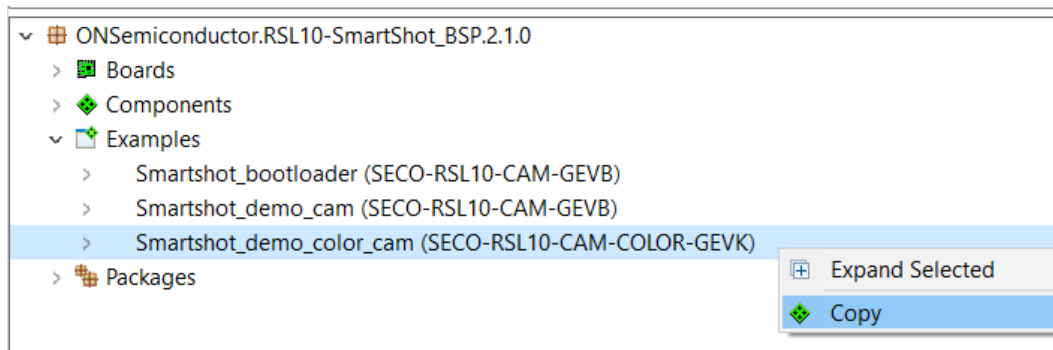


Figure 8. Right Click on the Sample Code easily access the Source Code and Complete Documentation

Once the above action is executed, user has access to source codes for RSL10 SoC attached this platform. It's split in two sample codes:

1. *SmartShot application*: it provides the source code and documentation for actual Smartshot application running on RSL10. It's supporting both boards SECO-RSL10-CAM-COLOR-GEVK and SECO-RSL10-CAM-GEVB (predecessor).
2. *SmartShot bootloader sample code*: Both boards support local and remote Firmware Over The Air (FOTA) updates for the RSL10.

The SmartShot application demonstrates wireless camera platform based on the RSL10 Bluetooth 5 Radio System-on-Chip (SoC). The camera is capable of capturing single image or continuous stream of images. Additionally sensor triggers are exposed for motion, acceleration, temperature and humidity change events to allow image capture based on environmental conditions.

All features of this demo application can be evaluated using the RSL10 Smartshot mobile application available for Android devices.

Operation modes of the application are shown in Figure 9. and are as follows:

Advertising Mode

- Default idle mode of the device.
- Only RSL10 is active with all other systems, including sensors, disabled. RSL10 enters deep sleep mode between advertising event to reduce power consumption.
- Advertising interval: 211.25 ms 1 minute after device startup and peer device disconnection and 852.5 ms after 1 minute without any peer connection.
- Device name: smartshot_demo_color_cam

- LED is turned on for the duration of advertising event with low brightness.

Connected Mode

- Entered after connection to a central device is established.
- Sensors can be enabled using the External Sensor Trigger Service to report environmental events.
 - ◆ Motion event – Captures nearby motion using PIR sensor.
 - ◆ Acceleration event – Uses accelerometer to capture motion of the board itself.
 - ◆ Temperature event – Monitors ambient temperature and reports if configured thresholds are crossed.
 - ◆ Humidity event – Monitors relative humidity and reports if configured thresholds are crossed.
- Low Power connection parameter update is automatically initiated after 30 seconds after peer device connection is established.
- LED is turned on with low brightness.

Image Capture Mode

- Entered after one-shot or continuous image capture command is received over Picture Transfer Service.
- Low Latency connection parameter update is automatically initiated before image capture operation.

- Powers up ISP, captures image and prepares compressed JPG image data.
- LED is turned on with medium brightness.

Image Transfer Mode

- Entered after image data are ready and central device starts image data transfer.
- Only limitation is ability of central device to receive and process data.
- Low Power connection parameter update is automatically initiated after image transfer operation completes.
- LED is turned on with high brightness.

Power Down Mode

- All systems are put into low power modes with lowest power consumption.
- Low power mode can be entered by keeping the on-board button pressed after releasing the reset button.

FOTA Mode

- Application control is given to FOTA application that waits for 30 seconds for central device to connect and perform firmware update.
- FOTA mode can be entered by pressing the on-board push button during normal operation.

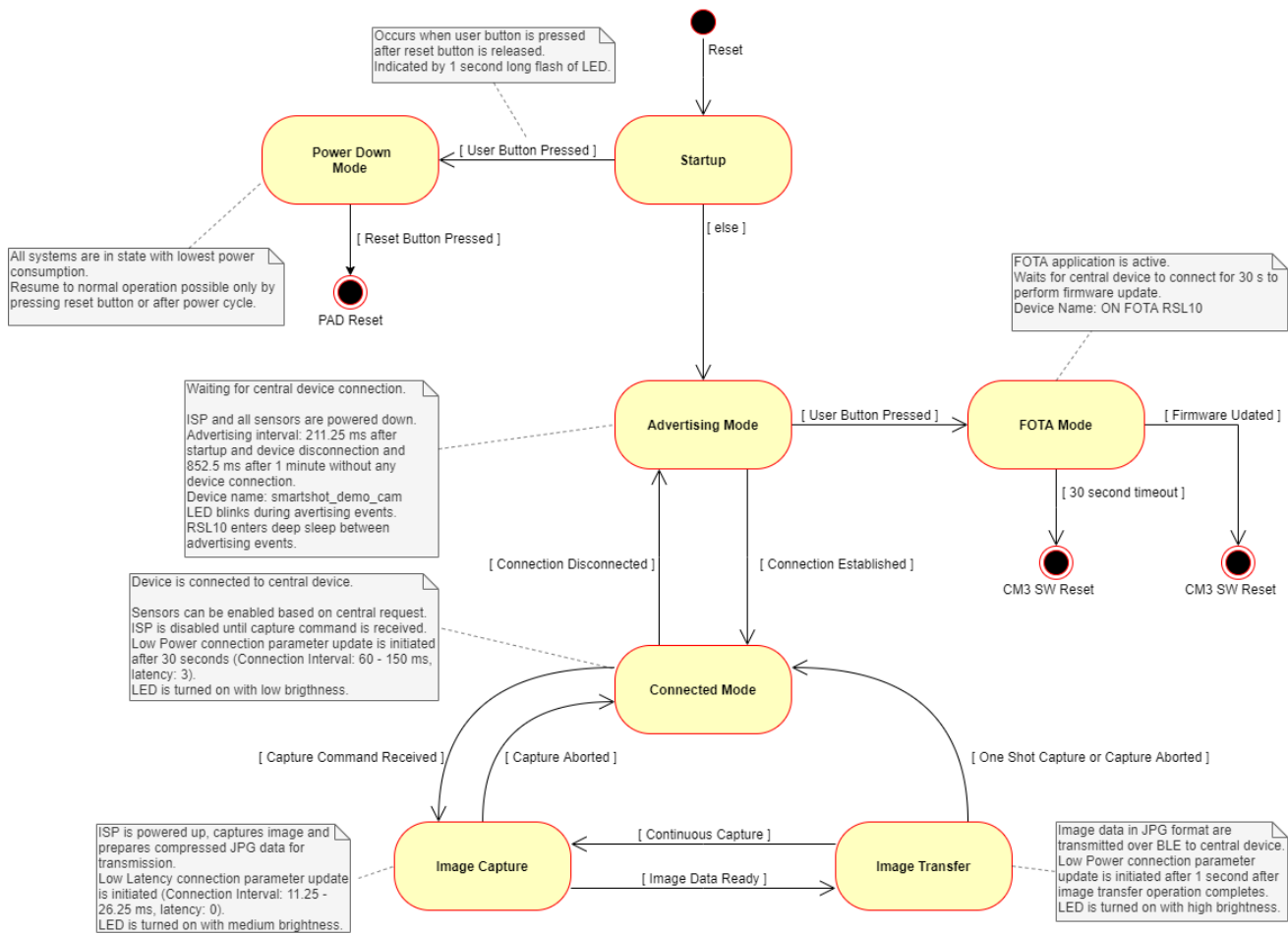


Figure 9. State Diagram of the Application

Hardware Requirements

The smartshot color demo cam example code is compatible only with the SECO-RSL10-CAM-COLOR-GEVK evaluation board.

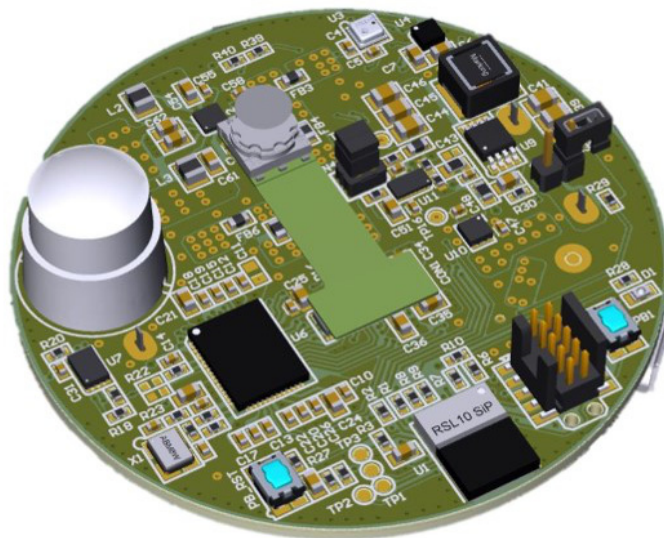


Figure 10. SECO-RSL10-CAM-COLOR-GEVK Evaluation Board

BUILDING PROJECT

CMSIS–Pack Dependencies

This example in *ON Semiconductor.RSL10–SmartShot_BSP.2.1.0+1* depends on software components from the following CMSIS–Packs:

- ON Semiconductor.RSL10 – [Download](#)
- ARM.CMSIS – [Download](#)

Required minimal versions of listed packs are part of project configuration.

Refer to *RSL10 Getting Started Guide* from [RSL10 Documentation Package](#) for instructions on installing of additional CMSIS–Packs into supported development environments.

FOTA Dependencies

This application generates FOTA capable images that require additional external dependencies.

- Python v2.7 or later
- Install package *ecdsa* version 0.13 or later.
- Install package *pyserial* version 3.2 or later.
- Make sure Python is added to the system path.

Refer to *RSL10 Firmware Over–The–Air User’s Guide* from [RSL10 Documentation Package](#) for additional details on FOTA implementation.

Build Configurations

This sample code contains two pre–configured build configurations:

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

Configuration

The application provides multiple RTE configuration files that allow to modify application behavior. Configuration files support and can be opened using CMSIS Configuration Wizard Editor that is available as one of the editors in supported IDEs (Figure 11).

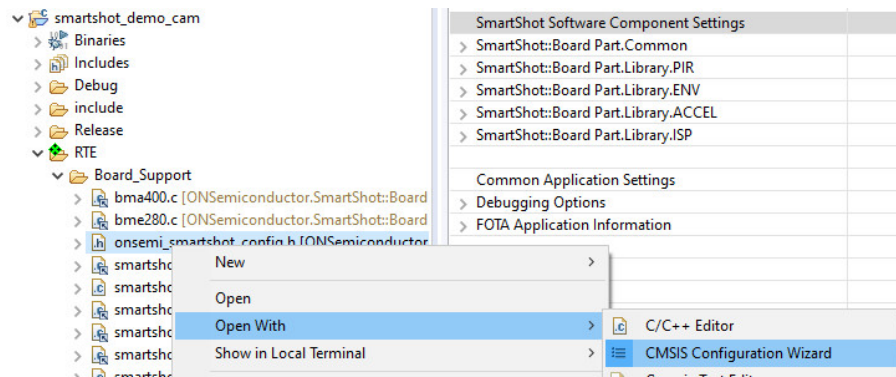


Figure 11. Selecting CMSIS Configuration Wizard as Editor for File in ON Semiconductor IDE

List of RTE configuration headers included with this example project:

- RTE/Board_Support/onsemi_smartshot_config.h – Allows to configure options exposed by the Board Support components (Figure 12).
- RTE/Board_Support/smartshot_assert_config.c – Allows to provide customized handler for failed assertions.
- RTE/Device/RSL10/RTE_Device.h – Configuration file for RSL10 peripherals. Used to configure I2C and SPI interfaces.

onsemi_smartshot_config.h	
CMSIS Configuration Wizard	
Option	Value
SmartShot Software Component Settings	
SmartShot::Board Part.Common	
Application Name	smartshot_demo_cam
SmartShot::Board Part.Library.PIR	
SmartShot::Board Part.Library.ENV	
Measurement parameters for BME280 sensor	
Temperature measurement	<input checked="" type="checkbox"/>
Oversampling	2x
Pressure measurement	<input type="checkbox"/>
Humidity measurement	<input checked="" type="checkbox"/>
IIR Filter Coefficient	2
SmartShot::Board Part.Library.ACCEL	
SmartShot::Board Part.Library.ISP	
Common Application Settings	
Debugging Options	
Enable ASSERT	<input checked="" type="checkbox"/>
PRINTF Output Interface	SEGGER RTT
Enable SystemView Real-Time Trace	<input type="checkbox"/>
Enable Deep Sleep	<input checked="" type="checkbox"/>
Power up ISP on boot	<input type="checkbox"/>
FOTA Application Information	
Application Identifier	SS_CAM
Version	

Figure 12. Example Content of onsemi_smartshot_config.h Header opened in Configuration Wizard

Flashing Procedures

To allow for FOTA functionality in the application the application is divided into multiple blocks that need to be loaded into device Flash memory. These components are shown in Figure 13. and consist of:

- **Bootloader**

Bootloader is the permanent part of the application that resides at the start of Flash memory and loads application images if available. Pre-compiled bootloader binaries for supported platforms are located in the /Boards/<Board_Name>/Binaries/ folders of the CMSIS-Pack. The bootloader project is also available as example project smartshot_bootloader/

- **FOTA Application Image**

The application image with .fota extension is

generated by mkfotaimg.py python script when project is build. This binary image merges together pre-compiled fota.bin binary with application binary and adds signatures or checksums of the binaries that are used to check image integrity during FOTA firmware update.

- ♦ FOTA BLE Stack binary

This binary contains shared parts of BLE stack and DFU application that is used during firmware update process.

- ♦ Application binary

This binary contains application itself and it is dependent on BLE stack functions from FOTA image.

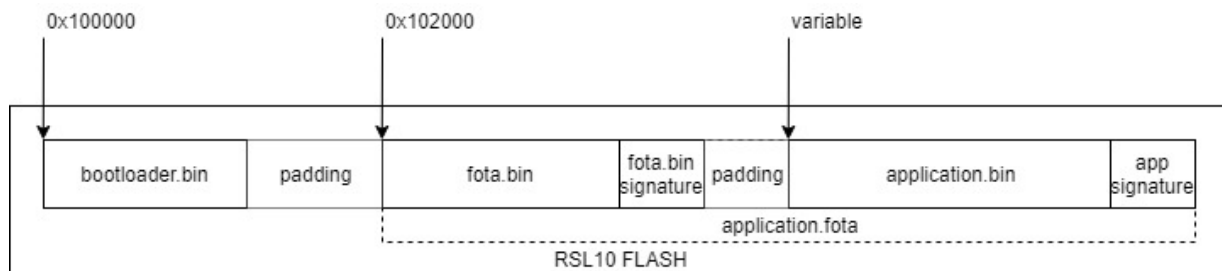


Figure 13. Layout of RSL10 Flash Memory with All Required Application Components

NOTE: Pre-compiled FLASH images are available in the /Boards/<Board_Name>/Binaries/ sub folder of the CMSIS-Pack.

Generating Initial Flash Image (bootloader + FOTA + application)

Below steps will describe how to generate and flash initial image into an empty or erased development board. Instructions provided below are for `srec_cat` utility but the same result can be achieved by any tool that can modify and merge binary files and provides output in Intel HEX format.

Using `srec_cat` to merge images

The `srec_cat.exe` utility must be available on your computer. This program can be downloaded from project's SourceForge page:

<https://sourceforge.net/projects/srecord/>

1. Build project and copy generated `smartshot_demo_color_cam.fota` file from build folder into your preferred working directory.
2. Copy bootloader image file `smartshot_bootloader.hex` for your board from /Boards/<Board_Name>/Binaries/ folder of the CMSIS-Pack and place it inside your working directory.
3. Open terminal in your working directory and run the `srec_cat` utility with following arguments:

```
srec_cat.exe smartshot_bootloader.hex
-intel smartshot_demo_color_cam.fota
-binary -offset 0x102000 -o
flash_image.hex -intel
```

After this step a `flash_image.hex` file will appear in the working directory.
4. Load the `flash_image.hex` file into development board using the RSL10 Standalone Flash Loader Utility and J-Link debug probe with appropriate debug adapter.

Updating Firmware using FOTA

The Firmware over the Air (FOTA) allows wireless firmware update.

Three methods are available for performing of FOTA updates on FOTA enabled devices:

- RSL10 FOTA Mobile Application ([Android](#), [iOS](#))
- FOTA.Console PC Tool with RSL10 USB Dongle
- Remote DFU service

Both methods are described in *RSL10 Firmware Over-The-Air User's Guide* from [RSL10 Documentation Package](#) in section 4.4 and section 9.

The SECO-RSL10-CAM-COLOR-GEVK board can be switched into FOTA mode by pressing and holding the on-board push button PB1 during normal operation or directly from RSL10 FOTA Mobile Application via DFU Service. Switch to FOTA mode is indicated by 1 second long flash of LED.

Please refer for detailed description in section *Firmware over the Air (FOTA) Updates* in **RSL10 Color Smart Shot Camera Getting Started Guide**.

The RSL10 Smart Shot Camera supports wireless firmware updates using the RSL10 Bluetooth 5 radio.

To begin, you will need the following:

- Download *RSL10_SMARTSHOT_FIRMWARE_UPDATE_PACKAGE.zip*
- Unzip above package and download the `smartshot_demo_color_cam.fota` image file to your mobile phone
- Install the *RSL10 FOTA* mobile app (available on [GooglePlay™](#), [iOS®](#)).

NOTE: Note: FOTA only updates the RSL10 SoC firmware. To update the Image Sensor Processor firmware (SPCV1100A) refer to the *SmartShot Firmware Update Guide.html* under *RSL10_SMARTSHOT_FIRMWARE_UPDATE_PACKAGE.zip*.

The boards can be switched into FOTA mode by pressing and holding the on-board push button PB1 during normal operation over RSL10 FOTA Mobile Application or remotely over DFU Service. Switch to FOTA mode is indicated by 1 second long flash of LED. The platform will then be discoverable on the RSL10 FOTA app.

Easy Update of RSL10 Firmware and ISP Firmware

Download

RSL10_SMARTSHOT_FIRMWARE_UPDATE_PACKAGE.zip. It consist the latest binaries not only for SECO-RSL10-CAM-COLOR GEVK, but also its predecessor SECO-RSL10-CAM-GEVB. Unzipping this package provides complete binaries and detailed *SmartShot Firmware Update Guide.html*.

RSL10_SMARTSHOT_FIRMWARE_UPDATE_PACKAGE > firmware_update

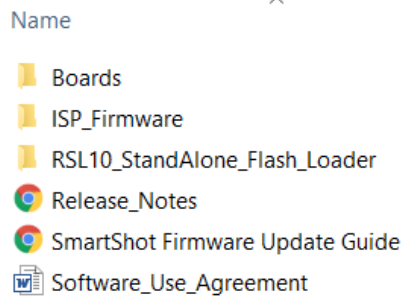


Figure 14.

APPLICATION DESCRIPTION

Advertising Interval

After device startup the advertising interval is set to high duty interval (211.25 ms). Without any peer device connection in 1 minute advertising interval is set to low duty interval (852.5 ms) to save power. After peer device disconnection the advertising interval is set again to high duty for 1 minute.

BLE Connection Parameters

BLE connection parameters are automatically updated by device depending on the state of application.

Low Power BLE Connection Parameter update is initiated 30 seconds after connection is established. This is to ensure low power consumption of device when peer device is connected, but high data throughput is not required. Low Power BLE Connection Parameter update is also initiated 1 second after any picture transfer is done.

Low Latency BLE Connection Parameter update is initiated as soon as image transfer request is received to increase maximum data transfer rate. This is done to ensure that image data are transferred to peer device as fast as possible.

- Low Power BLE Connection Parameters
 - ◆ Connection Interval Min: 60 ms
 - ◆ Connection Interval Max: 150 ms
 - ◆ Connection Latency: 3
- Low Latency BLE Connection Parameters
 - ◆ Connection Interval Min: 11.25 ms
 - ◆ Connection Interval Max: 26.25 ms
 - ◆ Connection Latency: 0

Important note: Apple devices and iOS are expected to be slower than Android devices. iOS does not allow to use as good connection parameters as Android does → smaller throughput. Additionally older devices like 6 SE might not have all features we use to speed up the transfer so it most likely falls back to some older BLE specification that is significantly slower.

Multi-shot is much faster than single-capture because ISP is operating continuously. So no delay due to ISP

startup and auto-exposure calibration is present after the first picture.

RSL10 Deep Sleep Mode

Application is utilizing the Deep Sleep Mode feature of RSL10 to save power during idle time when no processing is required.

Deep Sleep Mode is entered:

- Between advertising intervals during advertising operation.
- Between connection intervals while connection is established.
 - ◆ Sleep is entered only if device is not using Low Latency connection parameters.
 - ◆ Sleep is not entered while image capture or transfer is ongoing.

RSL10 wakes up on one of the following events:

- BLE Base Band – Wake-up due to upcoming advertising or connection interval.
- External Wake-Up – If at least one interrupt capable sensor is enabled.

BLE Services

This firmware configures the device to act as a peripheral server. Connectable advertising is started after initialization to allow peer devices to connect. High Duty (211.25 ms) Advertising Interval is used for 1 minute after device startup and peer device disconnection. Low Duty (852.5 ms) Advertising Interval is used after 1 minute without any peer connection. Deep sleep mode is utilized while advertising and peer device connection mode to save power. External Wake-Ups from Accelerometer and PIR sensors are used to wake-up RSL10 from Deep Sleep mode.

Several services are exposed to any connected client. This sample code does not impose any security or authentication requirements to connect or access any of its services and characteristics.

Device bonding is optionally available for peer devices that desire to use an encrypted connection. No additional state is preserved for bonded devices between multiple

connections (i.e. notifications enabled flags in CCC descriptors).

This application defines the following custom services:

- **Picture Transfer Service** Server (PTSS) – allows central devices to initiate image capture and receive processed image data.
- **External Sensor Trigger Service** Server (ESTSS) – exposes measured sensor values and allows to set trigger conditions if value meets settable conditions.
- **Device Firmware Update Service** Server (DFUS) – exposes version information from the device and initiate FOTA update.

This application uses the following general service:

- **Device Information Service** (DISS) – The Device Information Service exposes manufacturer and/or vendor information about a device.

Picture Transfer Service

This service is used to request capture of an image using the on-board camera module and to transmit captured image data. For full description of the service refer to *Picture Transfer Service* document in *Documentation* folder of the CMSIS-Pack.

Table 1. CHARACTERISTICS AND THEIR DESCRIPTORS EXPOSED BY PICTURE TRANSFER SERVICE IN THIS SAMPLE CODE

Characteristic	Descriptor	Permissions	Data Type	Payload Length (octets)	Default Value
Control Point		Write, Write Without Response	blob	1	N/A
	User Description	Read	string	13	"Control Point"
Info		Notify	blob	max 5	N/A
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	User Description	Read	string	4	"Info"
Image Data		Notify	blob	max 246	N/A
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	User Description	Read	string	10	"Image Data"

External Sensor Trigger Service

This service is used to report sensor values and to send notifications when configurable trigger conditions are met. For full description of the service refer to *External Sensor Trigger Service* document in *Documentation* folder of the CMSIS-Pack.

This instance implements 4 sensor triggers in the following order:

- **Motion Trigger**

- ◆ Reports status of the on-board PIR sensor.
- ◆ Possible values are 0 or 1 depending on the state of the sensor.
- ◆ Read of characteristic value reports current sensor state.
- ◆ Supported Value Triggers:
 - 0x00 – On value change
 - 0x02 – Value is on boundary
 - 0x07 – Value trigger disabled
- ◆ Supported Time Triggers:
 - 0x00 – No time trigger
 - 0x02 – Do not notify more than settable time

- ◆ Recommended trigger configuration:

Value Trigger Setting: 0x0201000000 – Value is on boundary with value set to 1. This configuration notifies only on start of motion event.

Time Trigger Setting: 0x02050000 – Notify at most every 5 seconds.

- **Acceleration Trigger**

- ◆ Reports when the board detects acceleration events.
- ◆ Possible values are 0 or 1.
- ◆ Read of the characteristic is not recommended. Acceleration event is reported only briefly, and should be detected by use of notifications.
- ◆ Supported Value Triggers:
 - 0x00 – On value change
 - 0x01 – Value crossed boundary
 - 0x07 – Value trigger disabled
- ◆ Supported Time Triggers:
 - 0x00 – No time trigger
 - 0x02 – Do not notify more than settable time.

This trigger setting configures sensor sample rate

to match with report interval. Default sample interval is 10 seconds.

- ◆ Recommended trigger configuration:
Value Trigger Setting: 0x0201000000 – Value is on boundary with value set to 1. This configuration notifies only on start of motion event.
Time Trigger Setting: 0x02050000 – Notify at most every 5 seconds.

• **Temperature Trigger**

- ◆ Reports current ambient temperature.
- ◆ The Characteristic Presentation Format descriptor describes how to parse the temperature value:
Format: SINT32
Exponent: 10e-2
Unit: degree Celsius
- ◆ If sensor is enabled by read operation, then first read value is invalid and should be ignored.
- ◆ Supported Value Triggers:
0x00 – On value change
0x02 – Value is on boundary
0x05 – Value crossed interval
0x07 – Value trigger disabled
- ◆ Supported Time Triggers:
0x00 – No time trigger
0x02 – Do not notify more than settable time
- ◆ Recommended trigger configuration:
Value Trigger Setting: 0x05D0070000B80B0000
– Temperature crosses either the 20° or 30°

interval boundary.

Time Trigger Setting: 0x02050000 – Notify at most every 5 seconds. Sets sample interval to 5 seconds.

• **Humidity Trigger**

- ◆ Reports current relative humidity in percent.
- ◆ The Characteristic Presentation Format descriptor describes how to parse the value:
Format: SINT32
Exponent: 10e-3
Unit: Percentage
- ◆ If sensor is enabled by read operation, then first read value is invalid and should be ignored.
- ◆ Supported Value Triggers:
0x00 – On value change
0x02 – Value is on boundary
0x05 – Value crossed interval
0x07 – Value trigger disabled
- ◆ Supported Time Triggers:
0x00 – No time trigger
0x02 – Do not notify more than settable time
- ◆ Recommended trigger configuration:
Value Trigger Setting: 0x053075000050C30000 – Humidity crosses either the 30% or 50% interval boundary.
Time Trigger Setting: 0x02050000 – Notify at most every 5 seconds. Sets sample interval to 5 seconds.

Table 2. CHARACTERISTICS AND THEIR DESCRIPTORS EXPOSED BY *EXTERNAL SENSOR TRIGGER SERVICE* IN THIS SAMPLE CODE.

Characteristic	Descriptor	Permissions	Data Type	Payload Length (octets)	Default Value
Motion Trigger		Read, Notify	SINT32	4	0 (false)
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	Value Trigger Setting	Read, Write	blob	9	No Value Trigger
	Time Trigger Setting	Read, Write	blob	4	No Time Trigger
	Presentation Format	Read	blob	2	Boolean, 10e0, Unitless, first
	User Description	Read	string	13	"Motion Trigger"
Acceleration Trigger		Read, Notify	SINT32	4	0 (false)
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	Value Trigger Setting	Read, Write	blob	9	No Value Trigger
	Time Trigger Setting	Read, Write	blob	4	No Time Trigger

Table 2. CHARACTERISTICS AND THEIR DESCRIPTORS EXPOSED BY *EXTERNAL SENSOR TRIGGER SERVICE* IN THIS SAMPLE CODE.

Characteristic	Descriptor	Permissions	Data Type	Payload Length (octets)	Default Value
	Presentation Format	Read	blob	2	Boolean, 10e0, Unitless, second
	User Description	Read	string	13	"Acceleration Trigger"
Temperature Trigger		Read, Notify	SINT32	4	0
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	Value Trigger Setting	Read, Write	blob	9	No Value Trigger
	Time Trigger Setting	Read, Write	blob	4	No Time Trigger
	Presentation Format	Read	blob	2	32 bit signed integer, 10e-2, Celsius, third
	User Description	Read	string	13	"Temperature Trigger"
Humidity Trigger		Read, Notify	SINT32	4	0
	Client Configuration	Read, Write	blob	2	Notifications Disabled
	Value Trigger Setting	Read, Write	blob	9	No Value Trigger
	Time Trigger Setting	Read, Write	blob	4	No Time Trigger
	Presentation Format	Read	blob	2	32 bit signed integer, 10e-3, Percentage, fourth
	User Description	Read	string	13	"Humidity Trigger"

Device Firmware Update Service (DFUS)

Allow an BLE Updater application from PC or Mobile phone to extract version information from the device and

initiate FOTA update. Refer to the RSL10 Firmware Over The Air Users Guide listed in Additional Resources section for more information about this service.

Table 3. CHARACTERISTICS AND THEIR DESCRIPTORS EXPOSED BY *DFU SERVICE* IN THIS SAMPLE CODE.

Characteristic	Descriptor	Permissions	Data Type	Payload Length (octets)	Default Value
Device ID		Read	SINT32	16	
	User Description	Read	string	9	"Device ID"
BootLoader Version		Read	SINT32	8	
	User Description	Read	string	18	"BootLoader Version"
BLE Stack Version		Read	SINT32	8	
	User Description	Read	string	17	"BLE Stack Version"

Table 3. CHARACTERISTICS AND THEIR DESCRIPTORS EXPOSED BY DFU SERVICE IN THIS SAMPLE CODE.

Characteristic	Descriptor	Permissions	Data Type	Payload Length (octets)	Default Value
Application Version		Read	SINT32	8	
	User Description	Read	string	19	"Application Version"
BLE Stack Build ID		Read	SINT32	32	
	User Description	Read	string	18	"BLE Stack Build ID"
Enter DFU		Write	SINT32	1	
	User Description	Read	string	14	"Enter DFU Mode"

Device Information Service

This service supports the following features:

- Manufacturer Name String
- Model Number String
- Software Revision String
- Hardware Revision String

Debugging

The sample code exposes several options for ease of debugging. These options can be found in the Board Support configuration file

RTE/Board_Support/onsemi_smartshot_config.h under the *Debugging Options* category.

Available options are:

- *Enable ASSERT* – Enables all assertions placed in application code and Board Support libraries.
- *PRINTF Output Interface* – Select default output channel for debug messages or disable them entirely.
- *Enable SystemView Real-Time Trace* – Enables support for SystemView trace in the application. *No trace modules are provided with this application.*
- *Enable Deep Sleep* – Allows to disable deep sleep functionality to maintain continuous debug session.
- *Power up ISP on boot* – Automatically powers up ISP on startup to allow ISP firmware update.

RTT Logging

RTT is the default channel for printing of debug messages and is enabled by default. See documentation of the `SmartShot::Board`

Support: Components: SEGGER component for example usage.

Use of sleep modes affects the ability to read logs through the debug probe. By default logging is possible only while SmartShot is connected to central device, when it does not use deep sleep. Alternatively the deep sleep can be disabled in debug options to allow for continuous readout of debug messages regardless of connection state.

Image Sensor Processor Firmware Update

By default the board comes with pre-flashed ISP from factory and the pre-loaded RSL10 firmware does not allow reflashing of the ISP. If there is need to update firmware of the ISP then the application must be build with the *Power up ISP on boot* debugging option. After flashing it into RSL10 the ISP is powered up and allowed to update its firmware until first image capture request is received.

RSL10 Smartshot Mobile App

The RSL10 Smart Shot Camera is supported by a mobile application available on Android and iOS . Using the app, developers can access a variety of features including cloud connectivity, remote configuration of sensors (thresholds and default settings), and change the image capture trigger modes. Connected to an Image Analytics/AI engine, the app is capable of detecting and identifying objects using an AI cloud service. The platform returns the captured image, along with the list of identified objects and confidence levels, on the app.

This section returns parametric description of each mode. Please refer for handy manual in **RSL10 Color Smart Shot Camera Getting Started Guide**.

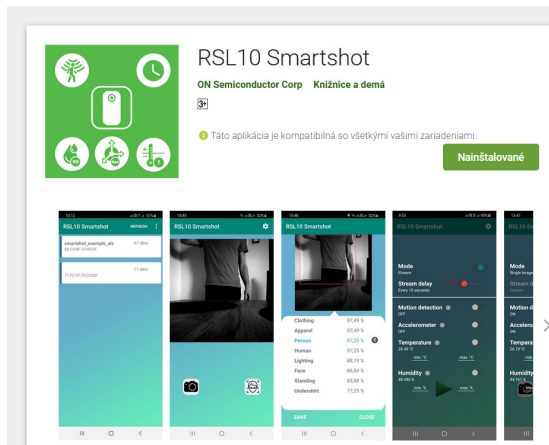


Figure 15. RSL10 Smartshot Mobile App

In the Current version of app let take a single image and enables recognition by the AWS Rekognition service, run stream of images and use the board like "alarm system" by using sensors placed on the board.

By setting some of the triggers sensors (PIR sensor, accelerometer, temperature and humidity measure), the platform will take an image at a particular moment when some of the sensors detect change in state or value.

INSTALLATION

The RSL10 Smartshot app is available on Google Play and iOS. After successful installation, you can open the app. If opening the app for the first time, all permissions must be enabled, including manually turning on GPS and Bluetooth functionalities.

RSL10 SMARTSHOT APP FEATURES

This section describes all functionalities of the RSL10 Smartshot set-up.

Discover Bluetooth Devices

Using the RSL10 Smartshot app, you can immediately starts to discover near Bluetooth devices. The app will only connect to the RSL10 Smartshot Camera which will be listed as "smartshot-demo-cam" or similar (Figure 16).

NOTE: If the platform is not discoverable, select the **Refresh** button.

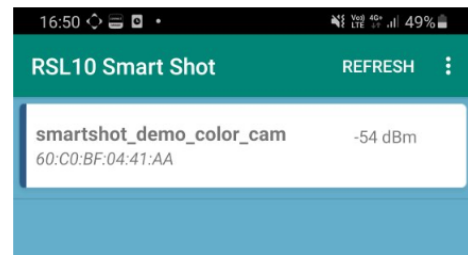


Figure 16. Discovery of the RSL10 Smart Shot Camera

Taking an Image

After successfully connecting to the platform, camera and recognition icons will appear.

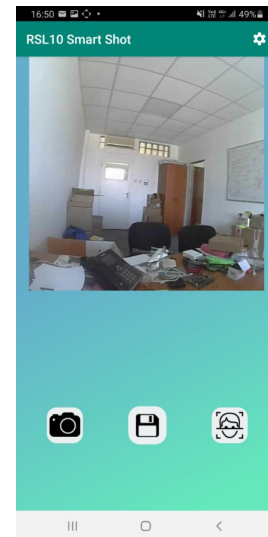


Figure 17. Taking the Picture

The most basic function is to capture a single image, which is done by selecting on the camera icon (left). Subsequently, the progress bar will be displayed to represent the progress of the receiving image. After receiving, the image is displayed on the screen. Next step may be to recognize this image by Amazon Rekognition.

Amazon Rekognition

You can provide your captured image to Amazon Rekognition, which will analyze the image and returns the labels of objects that were recognized on image. To use this function, you will need to create an AWS account and have your smartphone connected to the internet (Figure 18). If you do not have an AWS account, you can register for one [here](#).

To connect your AWS Account with the RSL10 Smartshot app, enter your *accessKey* and *secretKey*. After doing so, a screen with two rows will appear after click on the recognition icon, or you can find it in the option menu on the main (BLE discovery) screen.

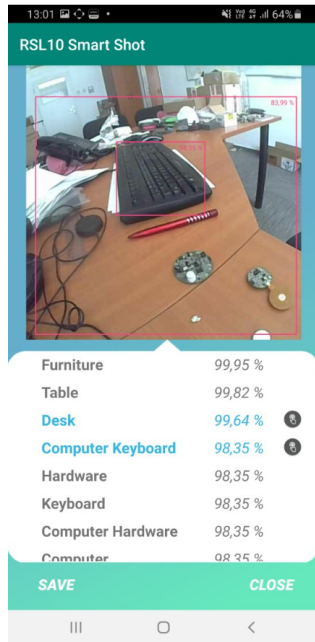


Figure 18. Amazon Rekognition

After analyzing the captured image, a list of detected objects is displayed. The blue label corresponds to a bounding box for a particular object within image

(Figure 18). The accuracy of compliance is represented by percentage number next to every label. The image can be saved for later analysis, or downloaded to your PC from your smartphone's storage. You can find the saved images in option menu on first (introductory) screen (Figure 19).

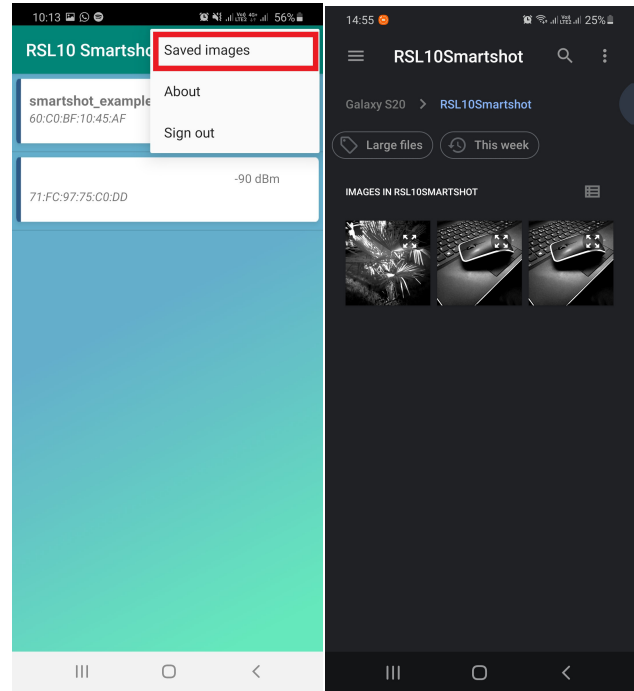


Figure 19. Saving Captured Images

NOTE: The RSL10 Smartshot app will only recognize images captured by RSL10 Smartshot platform.

Set Amazon Rekognition Permission

To use Amazon Rekognition services, you will need to confirm permissions in your AWS account.

1. Log into to your AWS account
2. Choose "IAM" service in "Security, Identity, & Compliance"

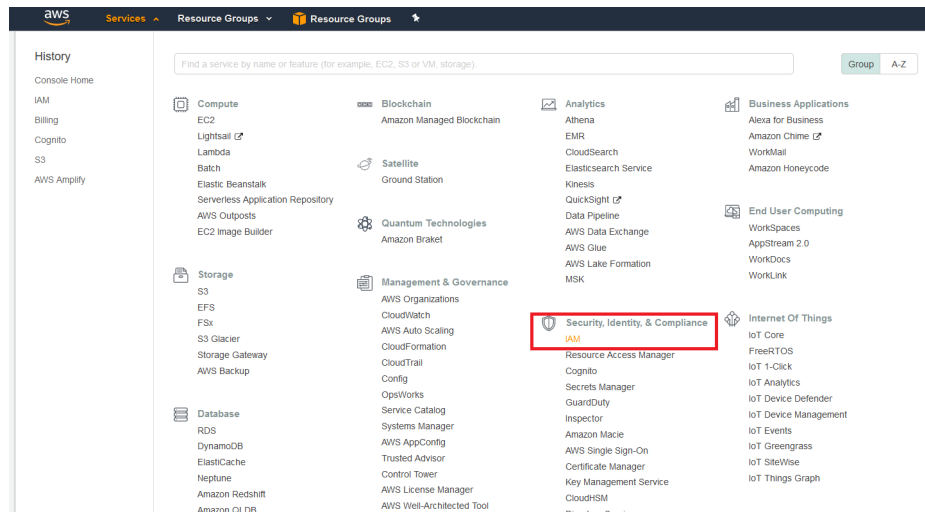


Figure 20.

3. In the left options menu, choose “Access management” → “Users” and add new user (Programmatic access → Next → Attach existing policies directly)

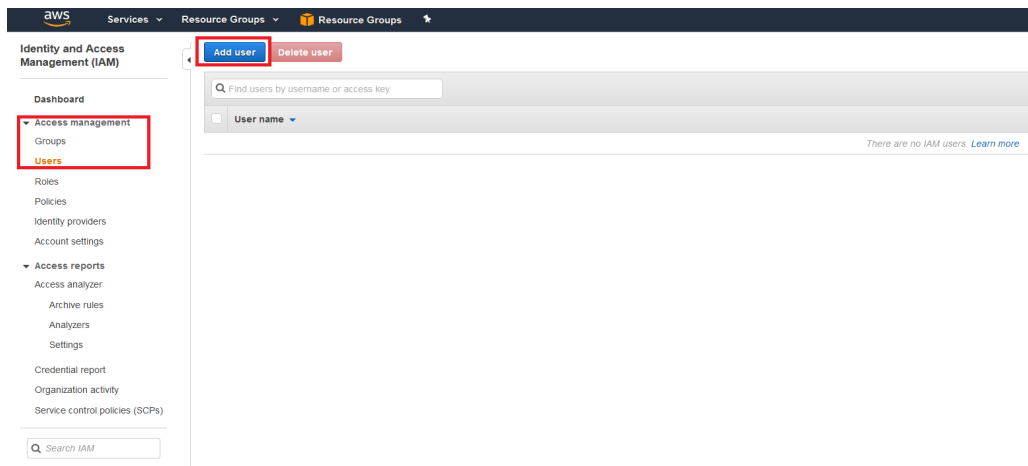


Figure 21.

4. Find “AmazonRekognitionReadOnlyAccess” and “AmazonRekognitionCustomLabelsFullAccess”, check the boxes and click on “Next: Review”

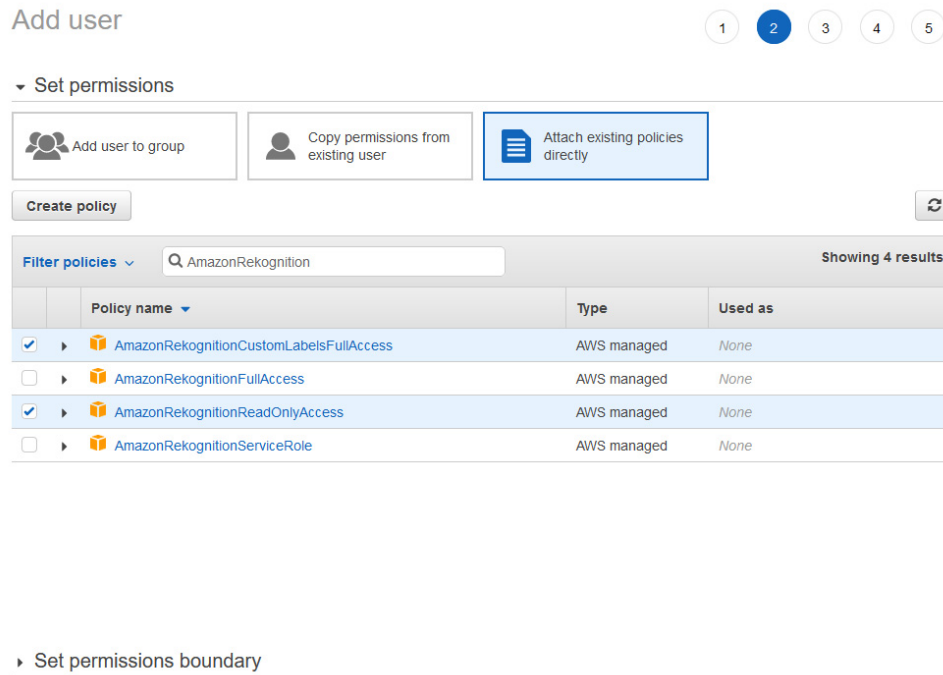


Figure 22.

5. Skip tags page and in review click on “Create user”
6. The last page shows the generated accessKey and secretKey. Copy SecretKey now and save it on

safe place for later use, because you won’t be able to see it in future

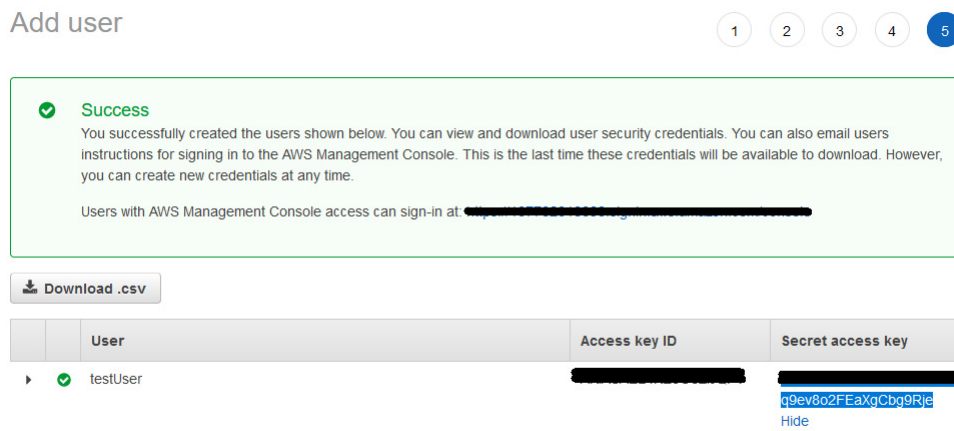


Figure 23.

7. Now users have accessKey and secretKey, so one can utilize them in the app.

Streaming Images

As well as using the platform to taking a single image, you can run a stream of images with different frame rates.

Sensor Triggers

The RSL10 Smartshot platform allows the use of 4 sensors to set triggers, which will take a picture when change

in value is detected. These functionalities can be found after tapping on gear wheel icon in the upper right corner placed on main screen (same as for the stream settings). You can set one or multiple triggers depending on your preference, or expose all sensors at once to trigger the image capture based your defined threshold limits .

The PIR sensor and accelerometer take a picture when the movement is detected, respectively acceleration. To set triggers for temperature and humidity, you have to enter a

lower and upper limits for both values. If one of these limits are exceeded, the RSL10 Smartshot platform will take an image and notice for particular sensor will be displayed.

The use of sensors triggers is possible only in “Single image” mode. The following sections describe the settings of each trigger. You can also find tips in the option menu by tap on information icons.

Motion Detection

By switching on “Motion detection” and hiding option menu, board starts to detect of motion by PIR sensor placed on RSL10 Smartshot platform. Board takes a picture and send captured data (image) to the app. The app will notify of a received image by the blinking motion icon. **Minimal time between two detection is 5 seconds.**

Acceleration Detection

By switching on “Accelerometer” and hiding the option menu, the platform starts to detect change of acceleration. The platform takes a picture and sends the captured data (image) to the app. The app will confirm that an image has been received with the blinking of the accelerometer icon. **The minimal time between detection is 5 seconds.**

Temperature Detection

The function of the temperature trigger needs two limits or boundaries, lower and upper, which define monitored interval. The current temperature value is displayed below the temperature label. You can also enter temperature limits with a maximum of two decimal places. After enter minimal and maximal values, switch on the button and hiding option menu, and the temperature trigger will become. If temperature crosses the lower limit, the RSL10 Smart Shot will take an image and display the temperature icon in blue. Likewise, the icon will be displayed in red if the upper limit has been exceeded. **The minimum time between detection is 5 seconds.**

In cases when value crosses the limit closely, the RSL10 Smartshot app may not notice crossing the same limit backward. For example, the upper limit is set to 25°C. If the

temperature exceeds the limit over 25.1°, the RSL10 Smart Shot will take a picture. However, if the temperature crosses the upper limit backward by value 24.95°C, the RSL10 Smart Shot will not notice. This is caused by time between detections and taking a picture.

In the opposite case, when temperature is increasing/decreasing rapidly, there may be a large difference between the limit and the value that has exceeded the limit. For example, if the limit is set to 25°C and the detected value is 35°C, 35°C is recognized as the first value that crossed the limit.

Humidity Detection

The humidity trigger function needs two limits or boundaries, lower and upper, which define monitored interval. The current humidity value is displayed below the humidity label. You can set the humidity limits with a maximum three decimal places. After entering the minimal and maximal values, switch on the button and hide the option menu to activate the humidity trigger active. The humidity trigger can be set between 0 to 100%. If humidity crosses the lower limit, the RSL10 Smart Shot Camera takes an image and displays the humidity icon in blue. Blue indicates that the lower limit has been crossed. If the upper limit has been exceeded, the humidity icon will be displayed in red. **Minimal time between two detections is 5 seconds.**

In cases, when value crosses the limit closely, the RSL10 Smartshot app may not notice crossing the same limit backward. For example, if the upper limit is set to 40% and the humidity level crosses the limit by 40.08%, the RSL10 Smart Shot will take a picture. If humidity crosses the upper limit backward by value 39.97%, the RSL10 Smart Shot will not notice. This is caused by time between two detection and taking a picture.

In the opposite case, when humidity is increasing/decreasing rapidly, may be a large difference between the limit and the value that has exceeded the limit. For example, if the limit is set to 40% and the detected value is 65.4%, 65.4% is the first value that has crossed the limit.

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

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

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

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

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

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

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

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

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

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

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

ADDITIONAL INFORMATION

TECHNICAL PUBLICATIONS:

Technical Library: www.onsemi.com/design/resources/technical-documentation
onsemi Website: www.onsemi.com

ONLINE SUPPORT: www.onsemi.com/support

For additional information, please contact your local Sales Representative at www.onsemi.com/support/sales