RSL10 Asset Tag
User Manual

UM70049/D

The RSL10 Asset Tag is an innovative development platform designed for a variety of industry 4.0 asset tracking and monitoring applications. With environmental sensors, a 3−axis accelerometer and support for Bluetooth® Low Energy connectivity and Quuppa Intelligent Locating System™ Real Time Localization Services (RTLS), the tag can be used to provide valuable data and insights into an asset's state of being and location. The RSL10 Asset Tag offers an industry−leading extended battery life of ~five years, a significant improvement compared to similar devices. This is made possible by the ultra−low−power consumption of the RSL10 radio System−on−Chip (SoC), advanced gate switching (FPF1005), low power environmental sensors, and optimized firmware.

This user manual provides an overview of the RSL10 Asset Tag (RSL10 ASSET TAG) hardware and software supporting software, and instructions for use.

Key Features
- Multi−Protocol Wireless Connectivity
- Bluetooth LE
  - Periodic Advertising
  - Triggered Mode (Acceleration Threshold)
  - Firmware Over the Air (FOTA) Updates
- Real Time Locating Services (RTLS) provided by the Quuppa Intelligent Locating System™
- Unprecedented Low Power Consumption for 5 Year Battery Life
- Environmental Sensing Technology (Temperature, Pressure) and a 3-axis Accelerometer
- Advanced Sensor Management
  - Ultra Low Leakage Gate Switches
  - Ultra Low Power 17 μJ Active Mode
  - 150 nJ per second in Standby
- Small Form Factor (27 mm Diameter)
- Supported by the RSL10 Sensor Beacon Mobile App (GooglePlay™)
- Easy Debugging/Programming Interface

Unboxing the RSL10 Asset Tag

The RSL10 Asset Tag board is shipped with pre−flashed firmware and a CR2016 3 V battery, as shown in Figure 1.

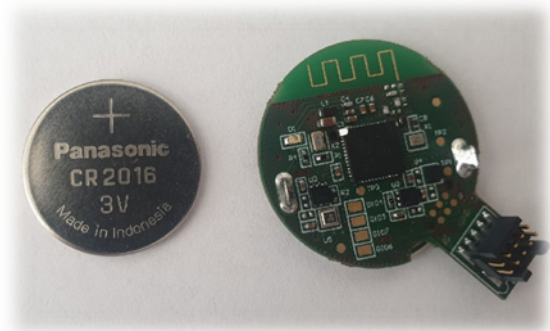## ON Semiconductor®

www.onsemi.com

USER MANUAL

Figure 1. RSL10 Asset Tag

To get started with the platform, insert the battery into the battery holder, as shown in Figure 2 below.
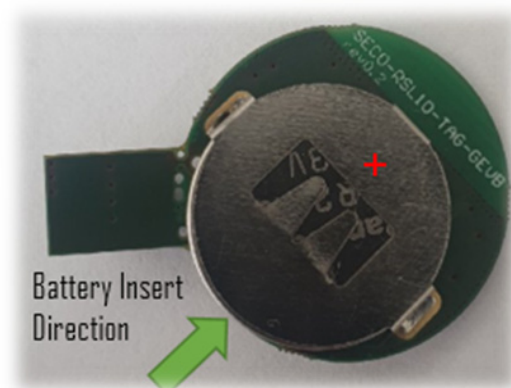

Figure 2. Inserting the CR2016 Battery into the RSL10 Asset Tag

HARDWARE DESCRIPTION

The RSL10 Asset Tag includes many different elements including environmental sensors and the RSL10 SoC, which is central to the platform. The Bluetooth® 5 certified RSL10 is embedded alongside a PCB antenna and matching circuit, and is used to capture and send sensor data and other wireless functions including non−connected Bluetooth LE advertising and Firmware Over the Air (FOTA) updates. The tag also features a 10−pin header for debugging that can be cut/removed depending on the application's housing needs.
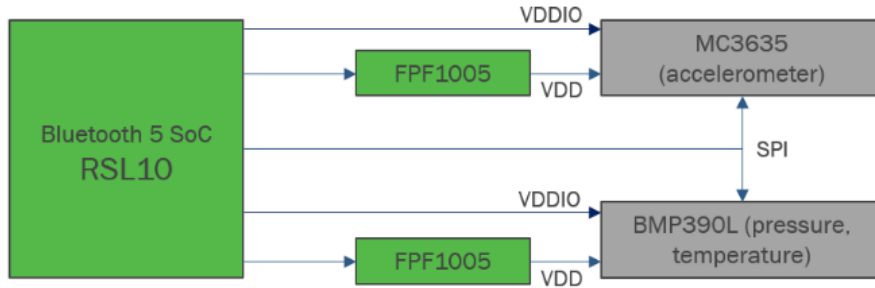

Figure 3. High−Level System Block Diagram

Sensors and Gating Switches

Alongside the RSL10, the platform features two sensors with unique, low power consumption profiles:

BMP390L – low power environmental pressure sensor with embedded temperature sensor

MC3635 – low power accelerometer for motion sensing

In order to decrease the power consumption further, each sensor features a unique supply gating thanks to the FPF1005 MOSFET load switch with an average current consumption of 1nA at 25 °C. The BMP390L and MC3635 provide a two−pin power supply scheme: VDD core and VDDIO. The VDDIO pin is always connected to the battery voltage and supplies the digital domain of the sensor. The VDD pin is the sensors' core voltage gated over the FPF1005. Thanks to this arrangement as well as the RSL10 Deep Sleep modes, the entire platform returns 25 nA on average when in Power Down Mode and unprecedented power consumption in Default, Triggered and Storage Modes.

If you're designing battery−operated devices, power consumption must be minimized. I^2C devices generally consume more power than SPI and, if power usage is a concern, an SPI interface should be used for your design. The RSL10 Asset Tag uses an SPI bus that significantly reduces overall consumption while in operation and generally features higher throughput bandwidth.

Low Cost Dome Switch

In order to reduce the overall cost of the solution and enable easier embedding into the device enclosure, a multi−purpose dome switch has been selected for the design. It sits underneath the battery, as depicted in Figure 4.
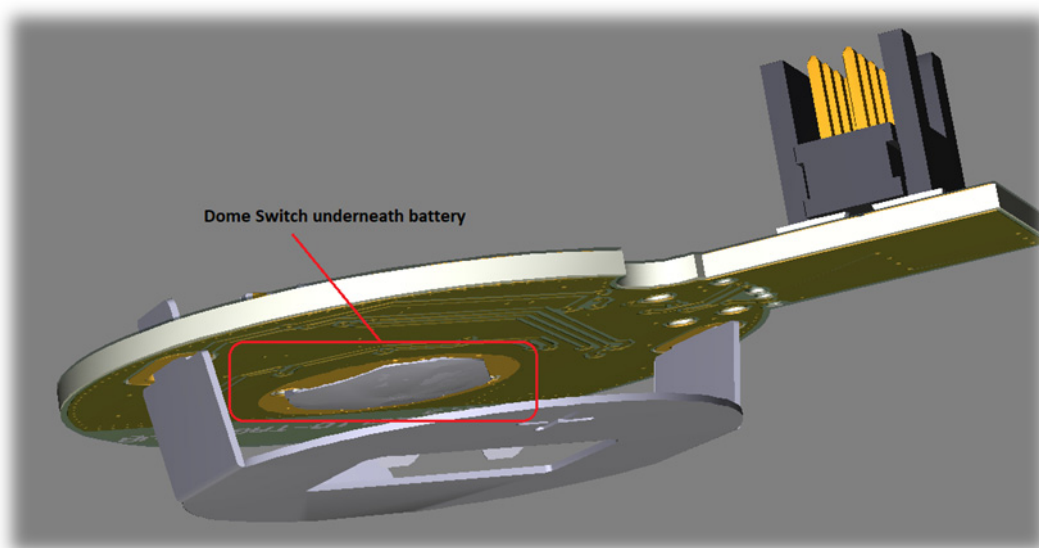

Figure 4. RSL10 Asset tag Dome Switch

When the battery is inserted, depending on their product development requirements , designers can decide to enable toggling of the switch by holding the board with two fingers and pressing the battery holder. You should hear a clicking sound to confirm that the dome switch has been pressed.

Various modes of operation are set this way, as described in further chapters.

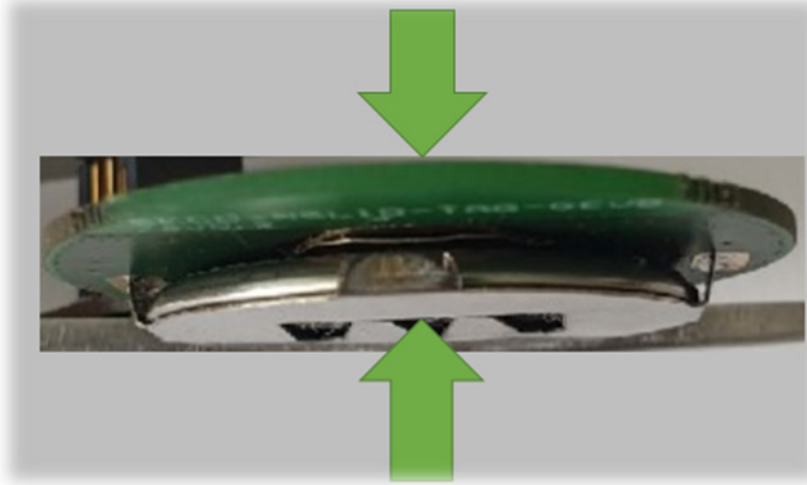NOTE: You do not have to toggle the switch upon first insertion.

Figure 5. Toggling the Dome Switch

PCB Antenna and Matching Circuit

The RSL10 Asset Tag features a low cost PCB antenna and matching circuit (Fig.6) that has been thoroughly optimized and measured for Bluetooth LE 5.0 emissions.
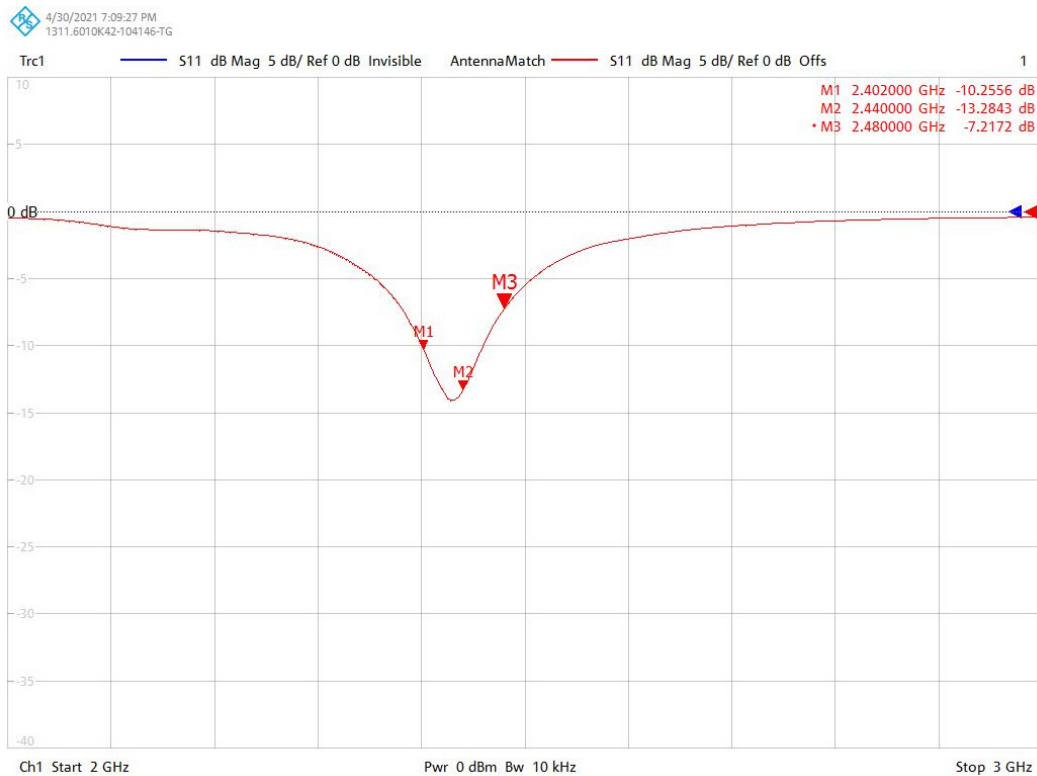

Figure 6. RSL10 Asset Tag Impedance Measure Match

External Pins

External pads are available on the RSL10 Asset Tag that can be used:

- Firmware debugging in Sleep Modes
- RF measurements (Direct Test Mode)
- External trigger pins
- External sensors

Figure 7 depicts the location of the external GIO pads. Due to flexible RSL10 crossbar structure, GIOs can be configured for various purposes using the RSL10 Software Development Kit (SDK).
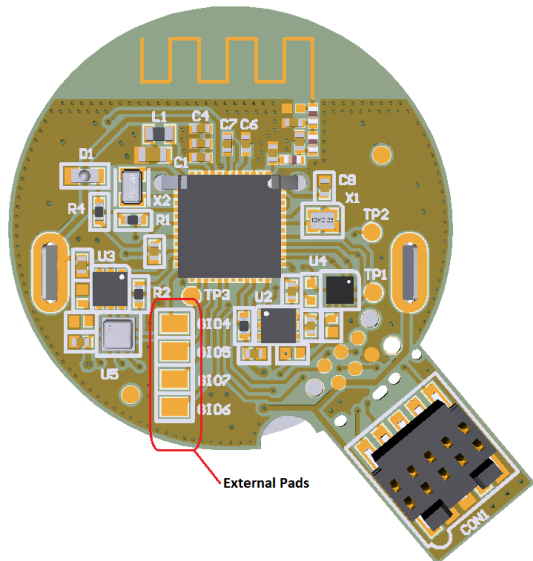

Figure 7. RSL10 Asset Tag External GIO Pads

RSL10 ASSET TAG Software Development Kit

The RSL10 Asset Tag is supported by a fully−featured and flexible Software Development Kit (SDK) that provides an excellent out−of−the−box experience. The SDK is provided with a CMSIS−Pack, easy installation and menu−based configurations for starter programmers.

10 PIN SEGGER DEBUG INTERFACE

The RSL10 Asset Tag features a dedicated debug interface:

- 6 pin needle adapter (mainly suited for mass productions and flashing the production firmware)
- 10 pin header (appendix for comfortable debugging and firmware development)

A standard J−Link debugger is supported for this purpose. For successful debugging, follow this procedure:

1. Insert battery into RSL10 Asset Tag board (battery creates reference voltage for debugger's digital signals)
2. Connect external J−link debugger (10 pin / 6 pin)
3. Install the RSL10 IDE and CMSIS packs,as described in the following sections
4. For debugging principles with ON Semiconductor IDE, please refer to Getting Started Guide at https://www.onsemi.com/products/connectivity/wireless−rf−transceivers/rsl10
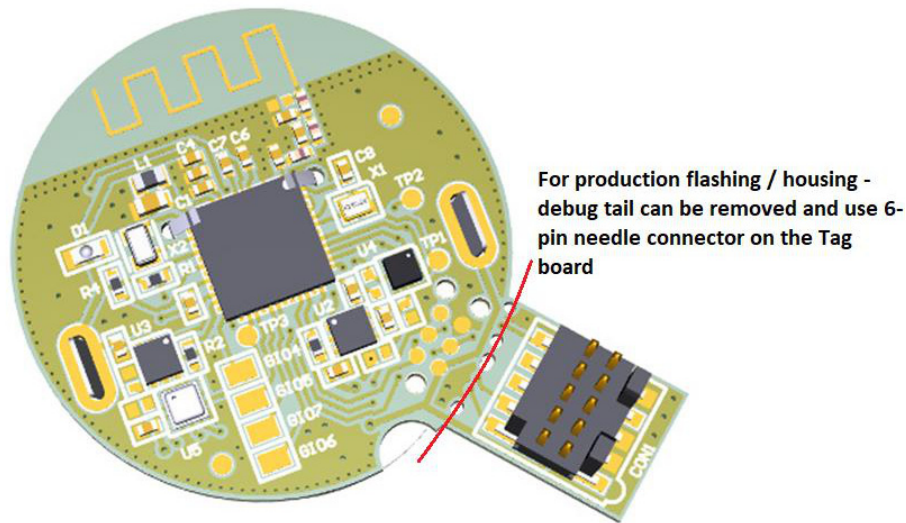

Figure 8.

Figure 9. SECO−RSL10−TAG−GEVB Evaluation Board and Connection to J−LINK Debugger

For production flashing / housing - debug tail can be removed and use 6-pin needle connector on the Tag board

Software Prerequisites

- Java (64−bit version)
- J−Link software (version 6.32i or higher)
- ARM CMSIS 5
- ON Semiconductor IDE Installer
- RSL10 Software Package (Installation steps provided in the RSL10 Getting Started Guide)
- RSL10 Asset Tag CMSIS Pack (*ONSemiconductor.RSL10−ATAG_BSP.1.0.0*)

Importing the CMSIS Packs

After the CMSIS packs are successfully imported, they can be viewed in the CMSIS pack manager perspective as shown below (Fig. 10). The RSL10 Asset Tag CMSIS Pack supports two sample codes:

- Atag_bootloader application
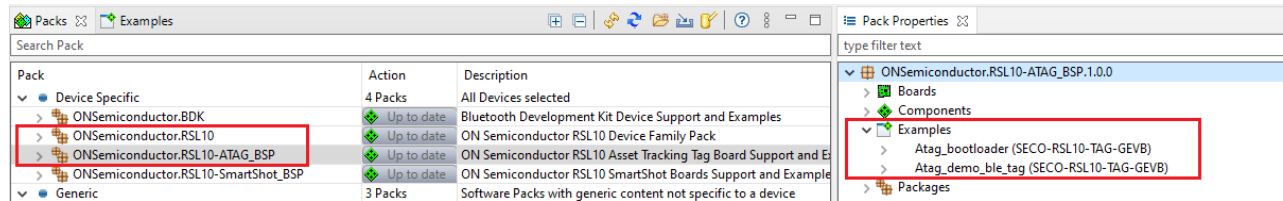- Atag_demo_ble_tag application

Figure 10. Imported CMSIS Packs

Copying Sample Code into the Project Space

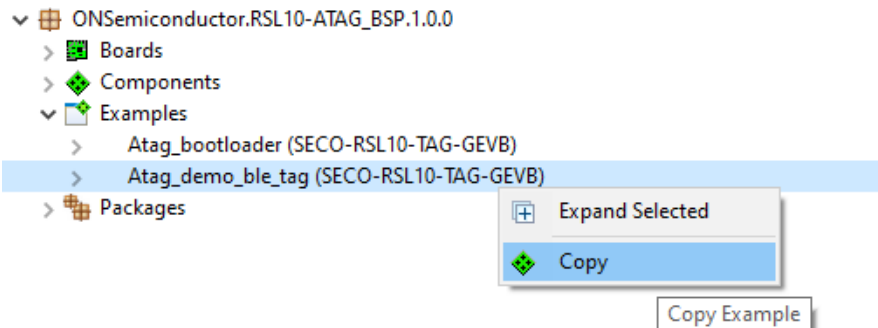1. Right click on the Atag_demo−ble−tag sample application and select Copy.

Figure 11. Copying Sample Code into the Project Workspace

Once the sample code is copied, you will have access to the complete source code and readme features detailing the sample application and description (Figure 12).
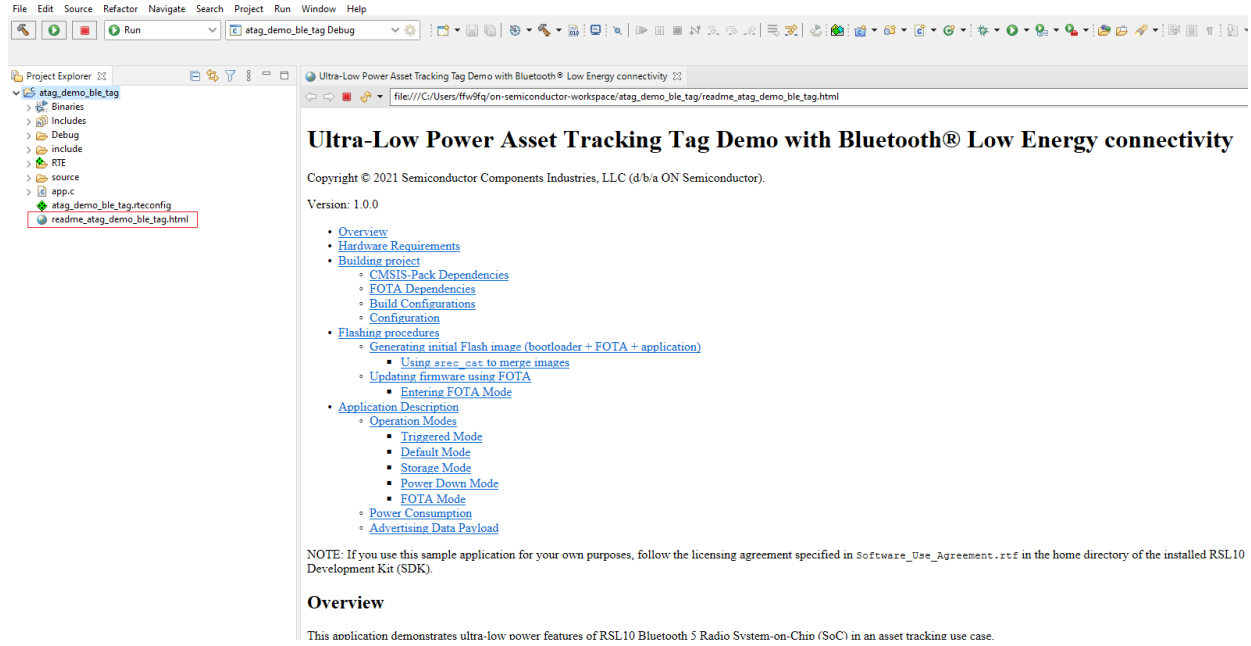

Figure 12. Expanded Sample Code with Available Documentation

RSL10 ASSET TAG DEMO SAMPLE APPLICATION

Overview

The RSL10 Asset Tag demo application demonstrates the ultra−low−power features of RSL10 Bluetooth 5 radio System−on−Chip (SoC) in an asset tracking use−case. The use case has several modes of operation that switch automatically based on external triggers including an accelerometer sensor or push button.

The application uses the following operation modes:

- Triggered Mode
  - ♦ Used for tracking of moving objects.
  - ♦ High advertising rate sufficient for walking speeds
  - ♦ Advertising data contain pressure data to enhance tracking performance in multi−story buildings
- Default Mode
  - ♦ Used for tracking of stationary objects.
  - ♦ Low advertising rate to report object presence
  - ♦ Continuously detects motion events to switch into Triggered Mode as needed

- Storage Mode
  - ♦ Power save entered when object remained stationary for specified period of time
  - ♦ No advertising
  - ♦ Continuously detects motion events to switch into Triggered Mode as needed
  - ♦ Can be disabled in the project configuration if continuous tracking is required
- Firmware Over the Air Mode
  - ♦ Used to update device firmware.
  - ♦ Entered by pressing the push button 3 times in quick succession.
- Power Down Mode
  - ♦ Device is disabled and can be woken up only by pressing the push button.
  - ♦ No advertising and no motion detection.

The state diagram in Figure 13 shows the relationship between all operation modes and default conditions for transitioning between states. Detailed descriptions and configurations of each state are described in the *Application Description* section.
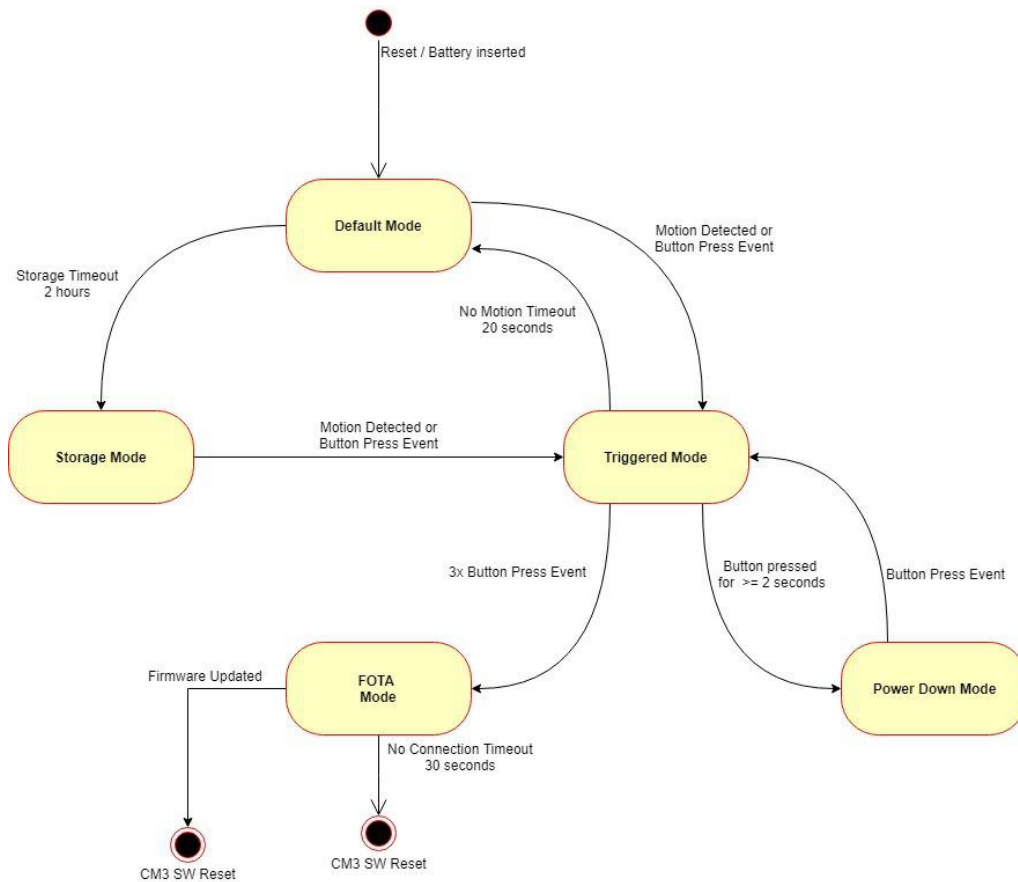

Figure 13. Application State Diagram

BUILDING A PROJECT

Firmware Over the Air (FOTA) Dependencies

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

- Python v2.7 or later (* Ensure this is added to the system path)
  - Package *ecdsa* version 0.13 or later
  - Package *pyserial* version 3.2 or later

Refer to *RSL10 Firmware Over−The−Air User Guide* with RSL10 Documentation Package for more details on FOTA implementation.

Build Configurations

This sample code contains two preconfigured build configurations:

1. Debug – Project is built with *−O0* optimization for enhanced debugging experience.
2. Release – Project is built with *−Os* optimization for reduced code size and speed optimizations that do not negatively impact code size.

Configuration

The application provides multiple RTE configuration files that allow developers to modify the application's behaviour. The configuration files can be opened using the CMSIS Configuration Wizard Editor that is available as one of the editors within the supported IDEs.
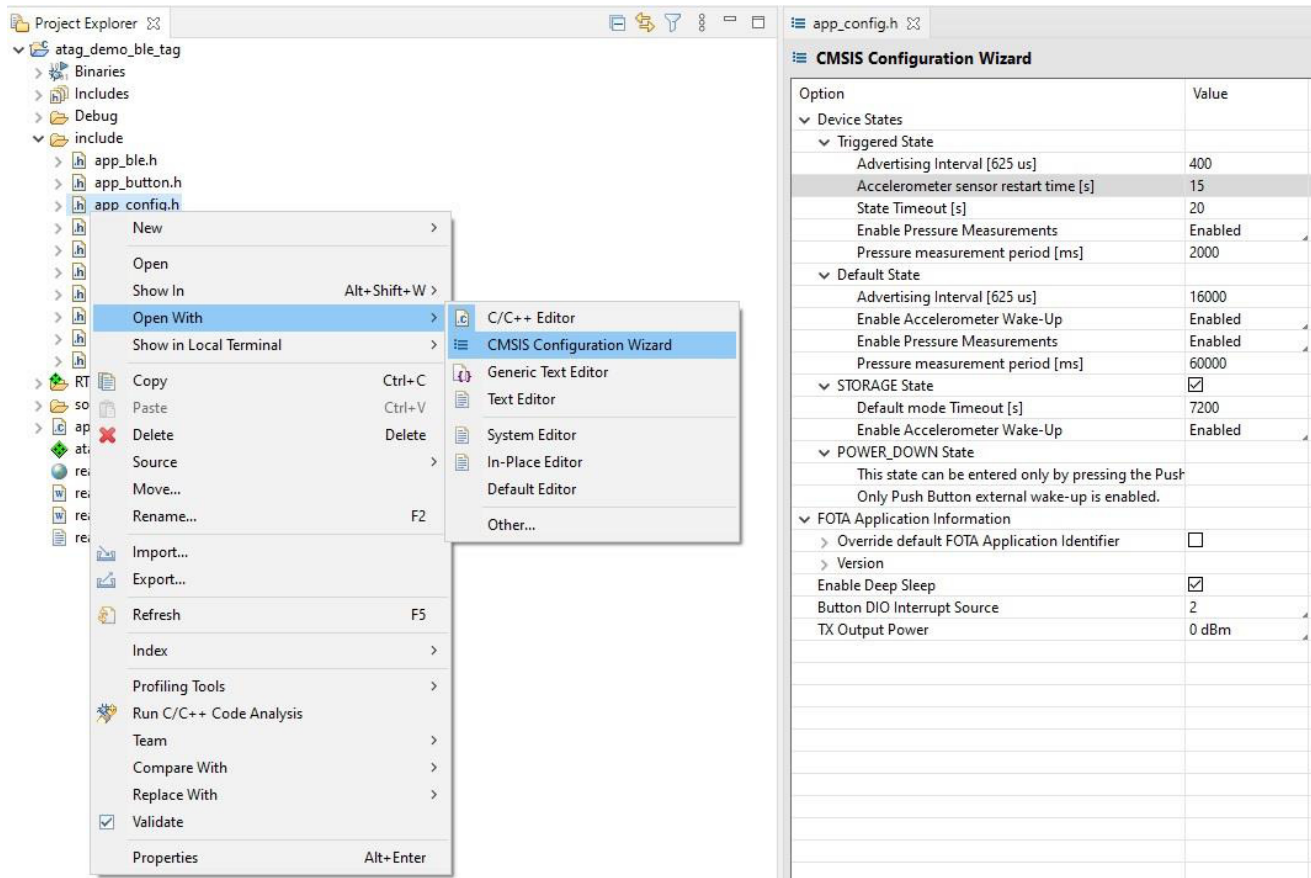

Figure 14. Selecting CMSIS Configuration Wizard as Editor for File in ON Semiconductor IDE

Here is a list of the RTE configuration headers included with this sample project:

- include/app_config.h – Contains configuration variables related to operation states and application itself.
- RTE/Board_Support/onsemi_atag_config.h – Allows you to configure options exposed by the board support components.
- RTE/Board_Support/atag_assert_config.c – Allows you to provide customized handler for failed assertions.

- RTE/Board_Support/m+drv+mc36xx_config.h – Contains additional parameters of the low−level driver library used by the MC3635 accelerometer sensor.
- RTE/Device/RSL10/RTE_Device.h – Configuration file for RSL10 peripherals. Used to configure the I^2C and SPI interfaces.

Flashing Procedures

To enable FOTA functionality, the application is divided into multiple blocks that need to be loaded into device's Flash memory. These components are shown in Figure 15 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/* folder of the CMSIS−Pack. The bootloader project is also available as example project *atag_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 the pre−compiled *fota.bin* binary with the application binary and adds signatures or checksums of the binaries that are used to confirm image integrity during FOTA firmware updates

- ◆ FOTA Bluetooth LE stack binary

  This binary contains shared parts of the Bluetooth LE stack and DFU application that are used during firmware update process.

- ◆ Application Binary

  This binary contains the application itself and it is dependent on the Bluetooth LEstack functions from the FOTA image.
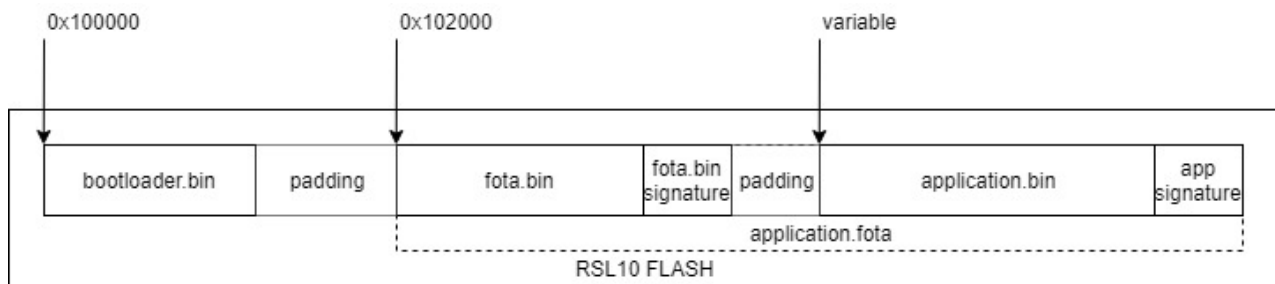

Figure 15. Layout of RSL10 Flash Memory with All Required Application Components

Note:

Pre−compiled FLASH images of the application with default settings are available in the */Boards/<Board_Name>/Binaries/* sub folder of the CMSIS−Pack.

The default CMSIS−Pack root directory used by ON Semiconductor IDE is (Fig. 16): *C:\\Users\<username>\ON_Semiconductor\PACK\ONSe miconductor\RSL10−ATAG_BSP\1.0.0*
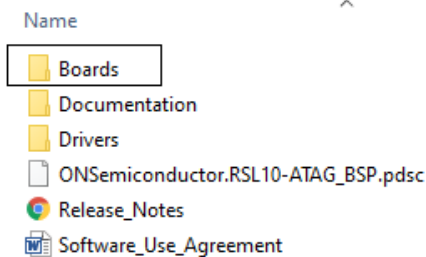

Figure 16. Un−packed RSL10 Asset Tag CMSIS−Pack

Generating the Initial Flash Image (Bootloader + FOTA + Application)

The steps below describe how to generate and flash an initial image onto an empty or erased development board. The provided instructions below are for the *srec_cat* utility.

The same result can be achieved by any tool that can modify and merge binary files and provides output into a .hex file format.

Using srec_cat to Merge Images

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

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

1. Build project and copy generated *atag_demo_ble_tag.fota* file from the Debug/Release build folder into your preferred working directory.
2. Copy bootloader image file *atag_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 atag_bootloader.hex −intel atag_demo_ble_tag.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.

Firmware Over the Air (FOTA) Updates

The Firmware over the Air (FOTA) allows firmware updates to be sent wirelessly. Two 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

Both methods are described in the *RSL10 Firmware Over−The−Air User's Guide* from RSL10 Documentation Package in section 4.4 and section 9.

Entering FOTA Mode

The RSL10 Asset Tag board can be switched into FOTA Mode by pressing the pushbutton three times in quick succession. After successful entering FOTA mode, the device will start advertising with the device name set to **ON FOTA RSL10**.

APPLICATION DESCRIPTION

Operation Modes

This section provides detailed description of available operation modes and their default configuration parameters.

Triggered Mode

This mode is entered as a response to external triggers that signal that the device is in motion or in response to an outside interaction using the pushbutton. As such, the advertising interval and sample rate of pressure data are much smaller than compared to Default Mode to allow better tracking of an object in motion. Typical examples for this mode include a person walking through a building or a cargo crate being transported through a facility or assembly line.

Accelerometer sensor and motion detection is disabled while the device is in Triggered Mode. The sensor is automatically re−enabled a few seconds before transition back to Default Mode to check for additional motion events.

Table 1.

Feature	Value	Description
Advertising Interval	250 ms	High advertising rate to keep track of moving objects.
No Motion Timeout	20 s	Time after which device returns to Default Mode if no motion events are detected.
Accelerometer restart time	15 s	Time after which to re−enable accelerometer and motion detection.
Sensor sample rate	2 s	Sample rate for pressure, temperature and battery voltage data.

Default Mode

This mode is intended for tracking of objects that are stationary for longer periods of time. In this cas,e the device switches to low advertising rate to report that the device is still present in the area.

Typical examples for when device is in this mode are a person sitting at a desk or a cargo crate on a shelf. Depending on whether or not Storage Mode is enabled, the device will either stop advertising entirely after a set period of time or remain active until next motion trigger is received.

Motion detection can be disabled to require button press before entering into Triggered Mode. Likewise, the measurement of pressure, temperature and battery voltage can be disabled too if the use case does not require these fields.

Table 2.

Feature	Value	Description
Advertising Interval	10 s	Low advertising rate to save power but still report object presence.
No Motion Timeout	2 h	Time after which device returns to Storage mode if no motion events were detected.
Sensor sample rate	60 s	Sample rate for pressure, temperature and battery voltage data.
Motion Trigger	Enabled	Accelerometer is enabled at all times to allow switch to Triggered mode.

Storage Mode

Storage Mode is entered after the device is in a longer period of inactivity with no motion. The device powers down and only the accelerometer and motion trigger are left active to allow for transition back into Triggered Mode. An example of this mode includes a forklift being left overnight after a work shift has ended. This operation mode can be completely disabled if not desired for the use case.The RSL10 is completely powered down (external wake−up with no RAM retention) when entering into this mode. When the pushbutton is pressed or a motion event occurs, the device executes Flash wake−up that resets the device and enters into Triggered Mode.

Table 3.

Feature	Value	Description
Advertising Interval	N/A	Device is disabled.
Sensor sample rate	N/A	Sensors are disabled.
Motion Trigger	Enabled	Accelerometer is enabled at all times to allow switch to Triggered mode.

Power Down Mode

This mode is identical to Storage Mode with these two conditions:

- Can be entered only by holding the push button pressed for more than 2 seconds.
- Accelerometer is always disabled while in Power Down Mode. The pushbutton is the only way to wake−up the device.

FOTA Mode

When switched into this mode, the application control is passed to the DFU application that handles firmware update process. After firmware updates are complete, the device is reset.

Power Consumption

The table below lists expected power consumption of the RSL10 Asset Tag while the board is in a given operation mode. Various combinations of sensor settings are listed based on the available configuration options.

Table 4.

Operation Mode	Adv. Interval	TX Power	Accelerometer	Pressure Sample Interval	Avg. Current @ 3 V	Life Expectancy1,2	Life Expectancy1,2
Triggered Mode	250 ms	0 dBm	Off	2s	32.7 μA	8.7 months	0.7 years
Triggered Mode	250 ms	0 dBm	Off	Off	29.4 μA	9.6 months	0.8 years
Default Mode	10 s	0 dBm	6.5 Hz SNIFF	60 s	1.45 μA	195 months	16.3 years
Default Mode	10 s	0 dBm	Off	60 s	1.10 μA	257 months	21.5 years
Default Mode	10 s	0 dBm	Off	Off	982 nA	288 months	24.1 years
Storage Mode	N/A	0 dBm	6.5 Hz SNIFF	Off	342 nA	829 months	69 years
Power Down Mode	N/A	0 dBm	Off	Off	25 nA	N/A	N/A

- 1 Assumes 230 mAh CR2032 battery and 90% battery efficiency (207 mAh)
- 2 Does not include startup and state transition events.

Advertising Data Payload

This example uses a custom 128−bit service data' advertising structure to expose the following information about the tag:

- Firmware Version
- Device State
  - Current Operation Mode
  - Motion event counter
  - Button press counter
- Ambient Temperature
- Atmospheric Pressure
- Battery Voltage

For documentation of the advertising structure refer to the Custom Advertising Data Types manual (UM700019)

- HTML documentation of the Utility.BLE Advertising data.ATAG AD Generator CMSIS−Pack component.

Byte	Field	Bit							
		7 (MSB)	6	5	4	3	2	1	0 (LSB)
0	AD Structure Length	ad_length [7:0] = ATAG_ADV_FRAME_LEN							
1	AD Data Type	ad_type [7:0] = GAP_AD_TYPE_SERVICE_128_BIT_DATA							
2 - 17	128-bit Service UUID	uuid[127:0] = ATAG_ADV_UUID							
18	Payload Version	payload_version [7:0] = 0x00							
19	Firmware Version	major [3:0]				minor [3:0]			
20		patch [7:0]							
21	Device State	dev_state [1:0]		motion_ind [2:0]			button_ind [2:0]		
22	Ambient Temperature	temperature [15:8]							
23		temperature [7:0]							
24	Atmosperic Pressure	pressure [23:16]							
25		pressure [15:8]							
26		pressure [7:0]							
27	Battery Voltage	batt_level [7:0]							

Figure 17. TAG Advertising Payload Structure Overview taken from HTML Documentation

RSL10 SENSOR BEACON MOBILE APP

The RSL10 Asset Tag supports data and states visualizations using the RSL10 Sensor Beacon app.

Getting Started with the RSL10 Sensor Beacon App

1. Ensure that *geo−location* and *Bluetooth* settings are enabled in the phone
2. Open the mobile app RSL10 Sensor Beacon
3. The app's discovery screen will show the State of the Tag, RSSI, temperature and pressure readings. The mobile app filters the RSL10 Asset Tags and shows as many as possible if in closer proximity of the mobile device (Figures 18, 19, 20, 21).

Figure 18. Mobile App Discovery Screen

Figure 19. Discovery Screen – Default State

Figure 20. Discovery Screen – Triggered State Invoked by Motion

Figure 21. Discovery Screen – Triggered State Invoked by Button

4. By tapping on the discovery screen, you will switch to the Advanced screen where you can explore device status and sensor readings, battery voltage and firmware revision (Fig. 22). You can also switch the screen into a chart mode by swiping left close to the 'two dots'. The data charts will show and store sensor values and state over time.

Figure 22. Switching from Advanced to Chart Mode

5. In order to save the data into a .csv file within your mobile phone, tap on the left arrow. Then, on the Advanced Options screen, select the three dots to see the *Export Data* button.
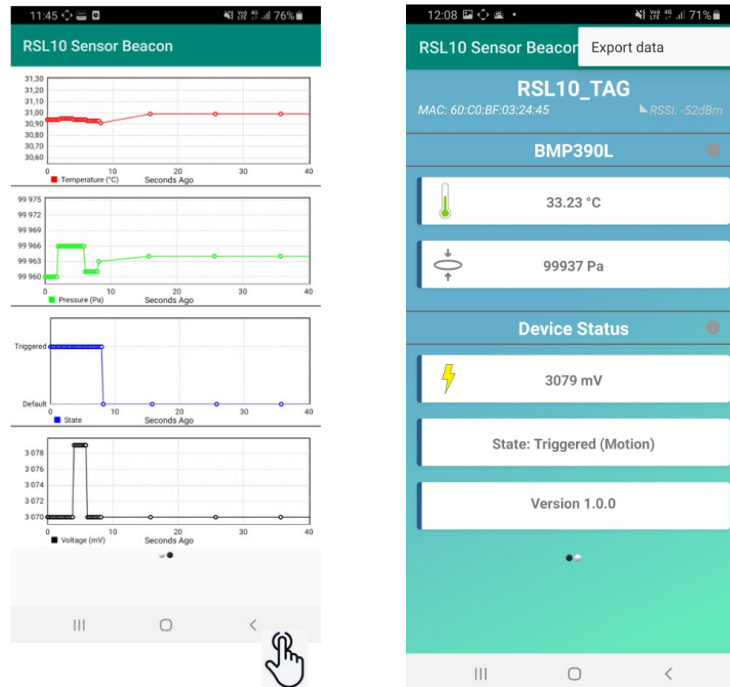

Figure 23. Exporting Data into the .csv File

Bluetooth is a registered trademark of Bluetooth SIG.
Google Play is a trademark of Google LLC.
Quuppa Intelligent Locating System is a trademark of Quuppa company in the EU and/or elsewhere.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Email Requests to: orderlit@onsemi.com

ON Semiconductor Website: www.onsemi.com

TECHNICAL SUPPORT
North American Technical Support:
Voice Mail: 1 800−282−9855 Toll Free USA/Canada
Phone: 011 421 33 790 2910

Europe, Middle East and Africa Technical Support:
Phone: 00421 33 790 2910
For additional information, please contact your local Sales Representative

◊