UG249: ZigBee® Occupancy Sensor Reference Design (RD-0078-0201) Kit User's Guide

Silicon Labs' ultra-low power, small, and cost-effective complete reference design based on the ZigBee HA (Home Automation) 1.2 standard focuses on occupancy sensors that can be used for occupancy and motion sensing. This ZigBee occupancy sensor reference design kit features Silicon Labs' EFR32MG Wireless Microcontroller (MCU), the Si7021 relative humidity and temperature sensor, and the Si1133 ambient light sensor.

KEY FEATURES

- Quick start demonstration
- Recommended next steps
- Kit overview
- Device operation
- Hardware and firmware considerations
- Engineering and manufacturing testing

---

1 Introduction

The Silicon Labs ZigBee Occupancy Sensor Reference Design Kit, shown in the following figure is ready out of the box to demonstrate ZigBee occupancy sensing applications. Each unit is an ultra-low power, small, and cost-effective complete reference design based on the ZigBee HA 1.2 standard. This reference design features Silicon Labs' EFR32MG Wireless Microcontroller (MCU), an Si7021 relative humidity and temperature sensor, and an Si1133 ambient light sensor.

This document will provide a quick start demonstration and next steps, system overview and operation, hardware and firmware considerations, and engineering and manufacturing test.

Figure 1. ZigBee Occupancy Sensor Reference Design

The reference design can be directly integrated into a connected ZigBee home automation application, as shown below. All hardware design files and firmware source code are available.
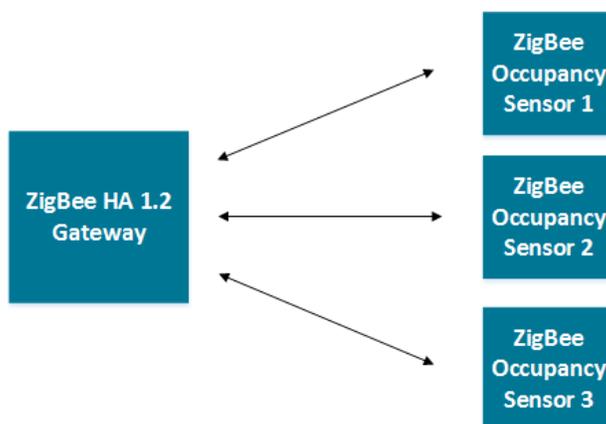

Figure 2. ZigBee Home Automation Application with Occupancy Sensors

2   Quick Start Demonstration

The ZigBee Occupancy Sensor Reference Design can be demonstrated as part of a ZigBee network. All ZigBee networks must have one device, such as a gateway, that plays the role of coordinator and allows commissioning of new devices on the network. Other ZigBee devices may be present in order to evaluate interoperability with the reference design. An example ZigBee network is shown in the figure below.
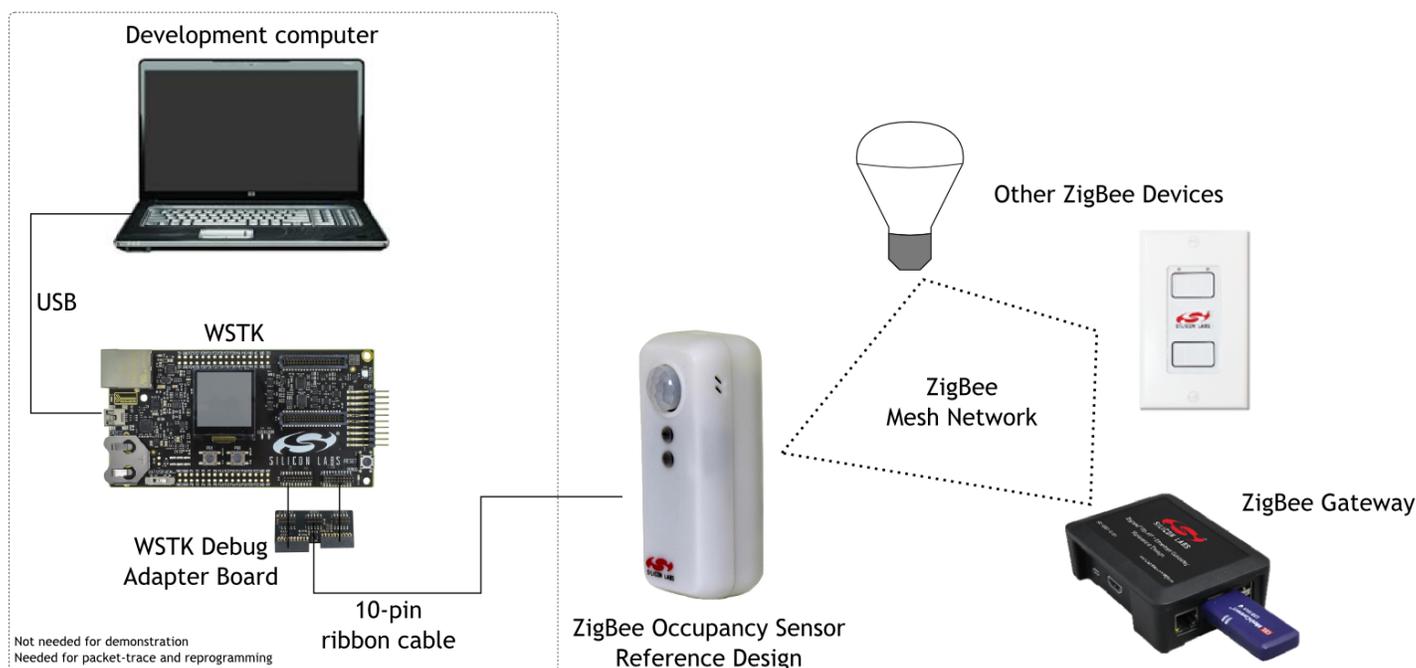

Figure 3. Example ZigBee Network

A video showing the quick-start demonstration is available at http://www.silabs.com/zigbeeoccupancysensor.

The following steps refer to several features on the printed circuit board of the reference design, shown in Figure 7.

Verify the Kit Contents

1.   ZigBee Occupancy Sensor Reference Design with 2 x AAA batteries.
2.   Quick start card to obtain the latest reference design collateral.

Set up an HA 1.2 ZigBee gateway

Silicon Labs offers the RD-0001-0201 or RD-0002-0101 ZigBee gateways (sold separately) for the purpose of evaluating the ZigBee Occupancy Sensor Reference Design. A commercially available Home Automation (HA) 1.2-compliant gateway that supports an occupancy sensor with device profile ID = 0x0107 will also work for this purpose. Depending on the commercially available gateway, the occupancy sensor can also display its temperature, humidity, and ambient light sensor reading. The RD-0001 and RD-0002 gateways support the occupancy sensor's additional features, but other commercially available gateways may not.

Supply power to the occupancy sensor

1.   Remove the top cover of the occupancy sensor.
2.   Set the SW1 switch towards the VBATT position.
3.   The occupancy sensor is powered when the red LED D2 momentarily flashes at least once.

Join the occupancy sensor to the ZigBee network

1.   Follow the directions provided with the ZigBee gateway to enable the occupancy sensor reference design to join its ZigBee network.
2.   Depress and hold the S1 button for more than one second, then release, to begin the network join procedure. The LED D2 will blink once every two seconds to indicate it is searching for a ZigBee network.
3.   The occupancy sensor has joined the ZigBee network when the red LED D2 blinks six times.
4.   The top cover can now be replaced.

Demonstrate occupancy sensor functionality

1. Any motion in front of the Fresnel lens within its detection range (as defined in Section **PIR Motion Sensor Behavior**) sends a 'motion detected' message to the gateway, with a detection rate based on the reference design's J1 header setting latched at power-up.

    The J1 header has two different modes indicated on the PCB silkscreen:

    *   TEST mode: an occupied state persists for seven seconds with no motion detected.
    *   NORM mode: an occupied state persists for 15 minutes with no motion detected.

    TEST mode is intended for demonstration and installation purposes but drains the batteries at a faster rate. NORM mode is intended for normal occupancy detection mode and is optimized for increased battery life by decreasing the rate that messages are sent when a room is continuously occupied. To change between modes, the reference design needs to be power-cycled after the jumper is set to the desired mode.

2. The occupancy sensor sends temperature, humidity and light illuminance sensor updates to the gateway every thirty minutes by default if the update rates are not configured differently by the gateway.

3 Recommended Next Steps

A video showing the recommended next steps is available at http://www.silabs.com/zigbeeoccupancysensor.

3.1 Evaluate the occupancy sensor

The ZigBee occupancy sensor reference design can support most occupancy sensor application requirements. Typical areas of evaluation include:

1. ZigBee network behavior, such as network join and leave.
2. Monitoring motion trigger events.
3. Monitoring relative humidity, ambient temperature and light illuminance.
4. RF performance, such as range.

3.2 Evaluate the Firmware

If firmware modification is needed to support your application:

1. Visit the ZigBee Getting Started page and order a development kit.
2. Refer to Section **Firmware Details for the ZigBee Occupancy Sensor Reference Design** of this document.

3.3 Build Proof of Concept

Out of the box, the occupancy sensor reference design can be used for a rapid proof of concept. Leveraging the provided plastic housing, one can demonstrate the look, feel, and function of a final product. Refer to Section **Firmware Details for the ZigBee Occupancy Sensor Reference Design** to change functional behavior, such as LED blinking or button functionality.

3.4 System Integration

Section **Hardware Details for the Occupancy Sensor Reference Design** describes the considerations for integrating the reference design into a typical application. Often the reference design can be designed into a system without modification. In cases where a change is required, the hardware section of this document also offers modification guidelines.

3.5 Engineering and Manufacturing Tests

Section **Engineering Tests** provides engineering test results, such as FCC and HA 1.2. Section **Manufacturing Tests** offers considerations for simplified manufacturing tests.

3.6 Manufacturing

Please contact Silicon Labs at http://www.silabs.com/support if you would like access to our manufacturing partner network for the reference design, or a modified version of the design.

4 Overview of the Reference Design

4.1 Part Numbers

The part number convention is RD-XXXX-YYYY, where:

RD   Reference Design

XXXX   Reference Design Number

YYYY   Reference Design Component

This document uses the reference design number (RD-XXXX) when describing the complete design, and the reference design component (RD-XXXX-YYYY) when describing a specific component.

The following table provides a description and PCB marking for each part number.

Table 1. Part Numbers and Description

Part Number	PCB Marking	Description
RD-0078-0201	N/A	ZigBee Occupancy Sensor Reference Design Kit with EFR32MG, Si7021, Si1133, and PCB Antenna.
RD-0078-0101	IST-A0078 Rev 4.0	ZigBee Occupancy Sensor Reference Design with EFR32MG, Si7021, Si1133, and PCB Antenna.

4.2   Features and Benefits of the ZigBee Occupancy Sensor Reference Design

This section is divided into three parts: features and benefits of the top-level reference design, features of the EFR32MG ZigBee SoC, and features of the reference design firmware.

Table 2. ZigBee Occupancy Sensor Reference Design Top-Level Features and Benefits

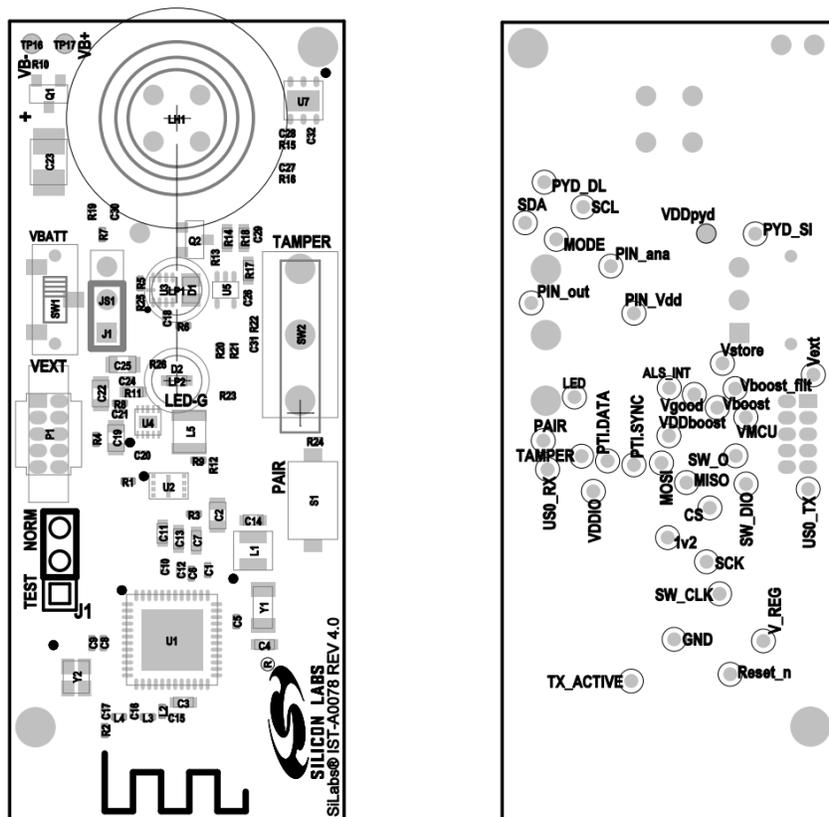| Reference Design Features	Benefit
Excelitas PYD1698 dual-element pyroelectric detector in a TOS-4 package with digital output and accompanying Fresnel lens.	Senses motion events with programmable detection sensitivity thresholds. Fresnel lens is designed for the sensor, easing mechanical design.
Si7021 I2C relative humidity and temperature sensor	Senses ambient temperature and humidity to aid home automation systems.
Si1133 I2C ambient light sensor	Senses ambient light level to aid home automation systems.
1024 kB SPI Flash	Supports firmware image storage during over-the-air upgrades.
Status LED	Provides feedback such as motion detection and network status.
10-pin Mini Simplicity-compatible connector	Facilitates programming and ZigBee packet trace debugging.
User button	Facilitates network commissioning (join/leave).
Optional tamper lever switch	Can detect if the plastic housing has been compromised/opened.
Normal / test jumper header	Allows users to put the device in install mode, which sacrifices battery life for a faster return to the unoccupied state.
Power toggle switch	Allows for the occupancy sensor to be powered by either the batteries or through the Mini Simplicity header.
Test points	Simplifies hardware-level debugging.
Plastic housing with 2 x AAA battery holder with on-board reverse battery protection circuitry	Supports low-cost, small footprint battery.

Figure 4. Reference Design Printed Circuit Board Diagram (Front and Back)

Table 3. Features of the EFR32MG ZigBee SoC

	Hardware capabilities
Processor	32-bit ARM® Cortex -M4 with DSP and FPU
RAM	64 kB
Crystal frequency	38.4 MHz
Encryption Accelerators	AES128
Advanced Crypto Accelerators	SHA-1, SHA-2, ECC
Peripherals	ADC, UART, SPI, TWI
On-chip flash size	256 kB
External serial flash size	1024 kB
MAC/PHY	2.4 GHz IEEE 802.15.4
Packet trace	Included
Voltage range	2.1 V - 3.6 V
Battery Life	5 years on 2 AAA batteries
Temperature range	0°C to 55°C
Transmit power	+ 13 dBm
Antenna	PCB, Inverted F
FCC certification	Pre-certified FCC Part 15

Table 4. Features and Benefits of the ZigBee Occupancy Sensor Reference Design Firmware

Feature	Details
Pre-compiled and source application firmware	
Over-the-Air (OTA) upgradable	Allows deployed devices to be upgraded in the field.
ZigBee HA 1.2 pre-certifiable clusters	Basic, power configuration, identify, OTA upgrade, temperature measurement, occupancy, illuminance, diagnostics.
Test mode operation	Aids with installation of the sensor and testing occupancy sensitivity settings.
Application firmware (source code) provided	Available in the EmberZNet™ PRO stack library

5 Operation Details

5.1 ZigBee Network Behavior

This section describes the various operational states a powered occupancy sensor reference design can occupy as it joins and leaves a ZigBee network. The input mechanism for operation utilizes a push button, S1. The red LED operates as output signaling information as the reference design traverses or occupies a functional state. Refer to the figure below for a pictorial representation of this process.
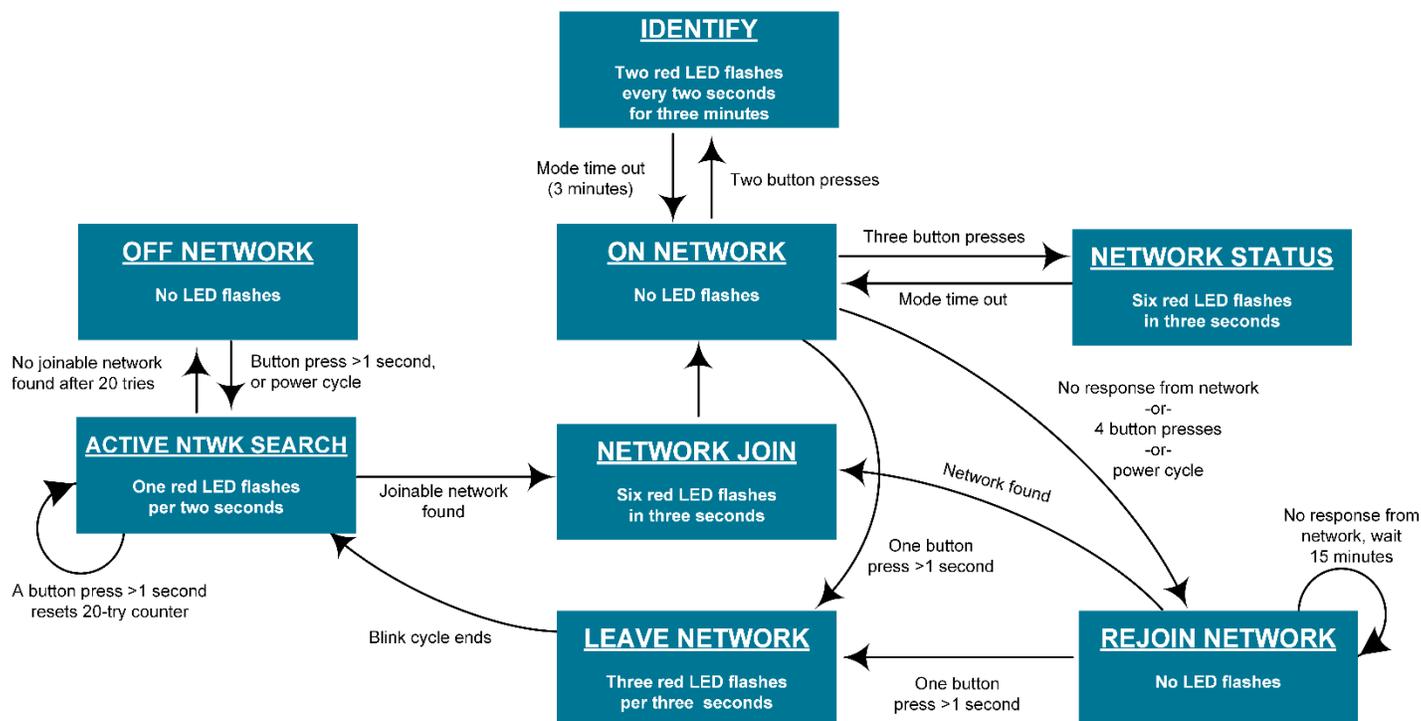
Figure 5. Device Operation State Diagram

First Time Occupancy Sensor Power-up

After the hardware has been powered for the first time, the device is in a factory default mode and is placed in the ACTIVE NTWK SEARCH state. If the occupancy sensor successfully joins a network and then is powered off, upon re-powering it enters the REJOIN NETWORK state instead of the ACTIVE NTWK SEARCH state.

OFF NETWORK State

This state is a condition where the device is not connected to a network and is not aggressively searching for a network. This occurs after the design has completed the ACTIVE NTWK SEARCH without finding a joinable network. To leave this state and enter the ACTIVE NTWK SEARCH state, press the S1 button for more than one second. The device does continue to perform infrequent searches for networks while in the OFF NETWORK state.

ACTIVE NTWK SEARCH State

In this state the device is actively searching for a network but has not yet joined a network. While in this state, the D2 LED flashes once every two seconds. ACTIVE NTWK SEARCH lasts for 20 attempts of scanning all channels, with each all-channel scan taking roughly 6-8 seconds. Thus the search will continue for 2-3 minutes, after which it enters the OFF NETWORK state if no network is found. Power-cycling or a button press while in this state restarts the search process. If a joinable network is found it leaves this state and enters the NETWORK JOIN state.

NETWORK JOIN State

After an occupancy sensor leaves the ACTIVE NTWK SEARCH state and enters this state, the D2 LED flashes six times in three seconds to indicate successfully finding and joining a network. This state is temporary and places the occupancy sensor into the ON NETWORK state once the LED completes its flash sequence. Power cycling only restarts the LED flash sequence, and does not prevent entry into the ON NETWORK STATE once the power is reapplied.

ON-NETWORK State

The ON-NETWORK state is a condition where the occupancy sensor is connected to a ZigBee HA 1.2 network and is not searching for another network. This state contains multiple entry and exit paths depending on various factors.

If power to the occupancy sensor is removed and later restored, the reference design does not seek other networks unless the user forces it to leave its joined network. Network information is always preserved in non-volatile memory so that it persists through power cycles. Button S1 provides a mechanism for users to change the network state of the device. In the following, rapid button presses are those that occur with no more than 0.5 seconds between presses.

- *If S1 is pressed once for more than one second,* the reference design enters the LEAVE NETWORK state.
- *If S1 is pressed rapidly two times,* the reference design temporarily enters the IDENTIFY state while remaining on its joined network.
- *If S1 is pressed rapidly three times,* the reference design temporarily enters the NETWORK STATUS state while remaining on its joined network.
- *If S1 is pressed rapidly four times,* the reference design enters the REJOIN NETWORK state.

IDENTIFY State

This state has two different purposes, depending on the method used to initiate it:

The first method is done locally to the unit by pressing S1 rapidly two times. The main purpose of this method is to support EZ Mode commissioning, whereby the occupancy sensor becomes discoverable by other devices on the network as a candidate for pairing. Note that if not on a network, pressing S1 rapidly two times causes the device to enter the ACTIVE NTWK SEARCH state.

The second method is done remotely by sending an Identify command from a gateway. The main purpose of this method is to allow an installer to visually locate the occupancy sensor in question.

No matter which method is used, after an occupancy sensor leaves the ON-NETWORK state and enters this state, the D2 LED flashes twice per second if on a network, then pauses for a full second, repeating the pattern for up to three minutes. This state is temporary, and the occupancy sensor will return to the ON-NETWORK state after three minutes. While in this state, the device will still report any change in open/close state or any change in tamper alarm state (if tamper monitoring is enabled).

If power is cycled while in this state, the design enters the REJOIN NETWORK state.

NETWORK STATUS State

The purpose of this state is to help an installer or user determine if a device is successfully connected to a network. After an occupancy sensor leaves the ON-NETWORK state and enters this state, the D2 LED flashes six times if on a network. This state is temporary, and the occupancy sensor will return to the ON-NETWORK state once the state's timer reaches its time limit. While in this state, the device will still report any change in occupancy state.

If not on a network, it enters the ACTIVE NTWK SEARCH state.

REJOIN NETWORK State

The purpose of this state is to ensure the design can successfully rejoin a previously joined network with little user intervention, while minimizing battery drain when searching for the network on both its original connected channel and on all channels. Upon locating its original joined network, the design enters the NETWORK JOIN state where it will rejoin the network.

For battery savings, if it receives no response from the network, the device waits 15 minutes before attempting to search for the network again.

LEAVE NETWORK State

Upon entering this state, the occupancy sensor leaves the joined network. It indicates this process is occurring by flashing the D2 LED three times within a two-second period. This state is temporary and will place the occupancy sensor into the ACTIVE NTWK SEARCH state once the LED completes its flash sequence.

5.2   PIR Motion Sensor Behavior

The ZigBee Occupancy Sensor Reference Design utilizes a dual-element PIR sensor with a provided 17-zone Fresnel lens with 90-degree aperture to detect motion within a visible distance. The reference design placed 2 meters vertically from the ground will detect a body moving 0.5 meters/sec up to 12 meters away within its 90-degree aperture. A full "arm-circle" wave can be detected up to 5 meters

away. Alignment of the sensor is critical to a predictable sensor viewable area. The specifications are based on both the PIR sensor and Fresnel lens as well as how the plastic housing is placed.

6 Hardware Details for the Occupancy Sensor Reference Design

This section describes the key aspects of the reference design.

6.1 ZigBee Occupancy Sensor Reference Design RD-0078-0101

This section describes the key hardware specifics of the RD-0078-0101 module.

Key Highlights:

- EFR32MG Wireless MCU with an embedded flash
- Si7021 relative humidity and temperature sensor
- Si1133 I2C ambient light sensor
- 1024 kB external flash
- Network / Motion Detect LED
- Button for commissioning
- Optional tamper switch
- Two-layer 0.062" printed circuit board
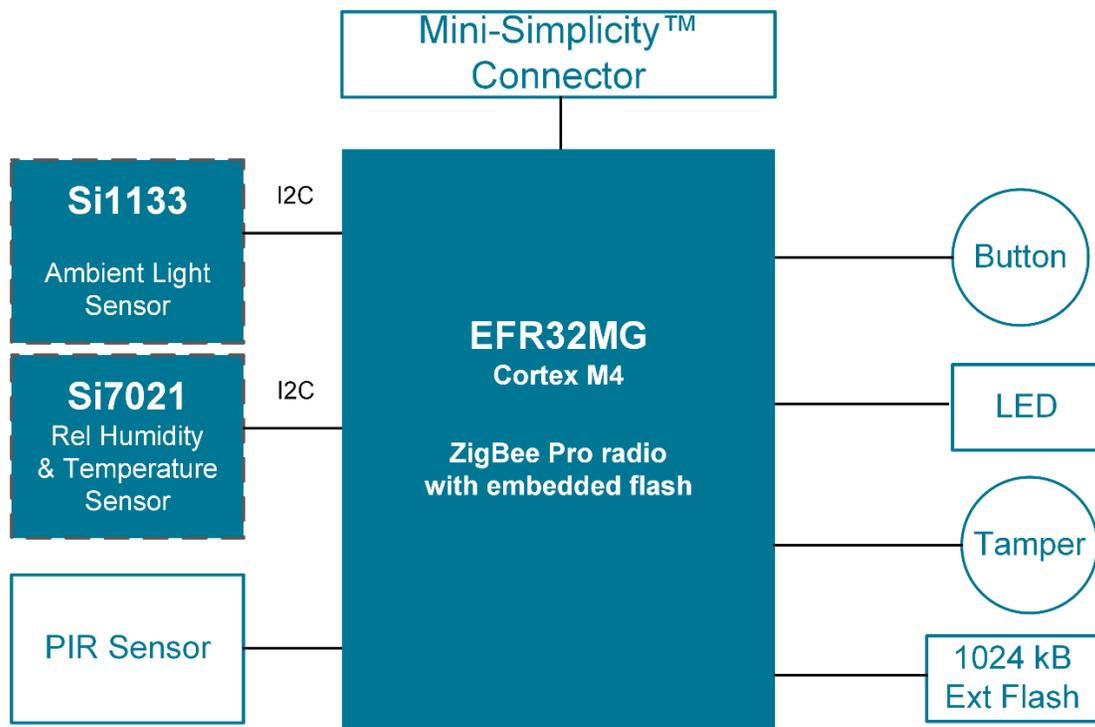
Figure 6. Reference Design Block Diagram
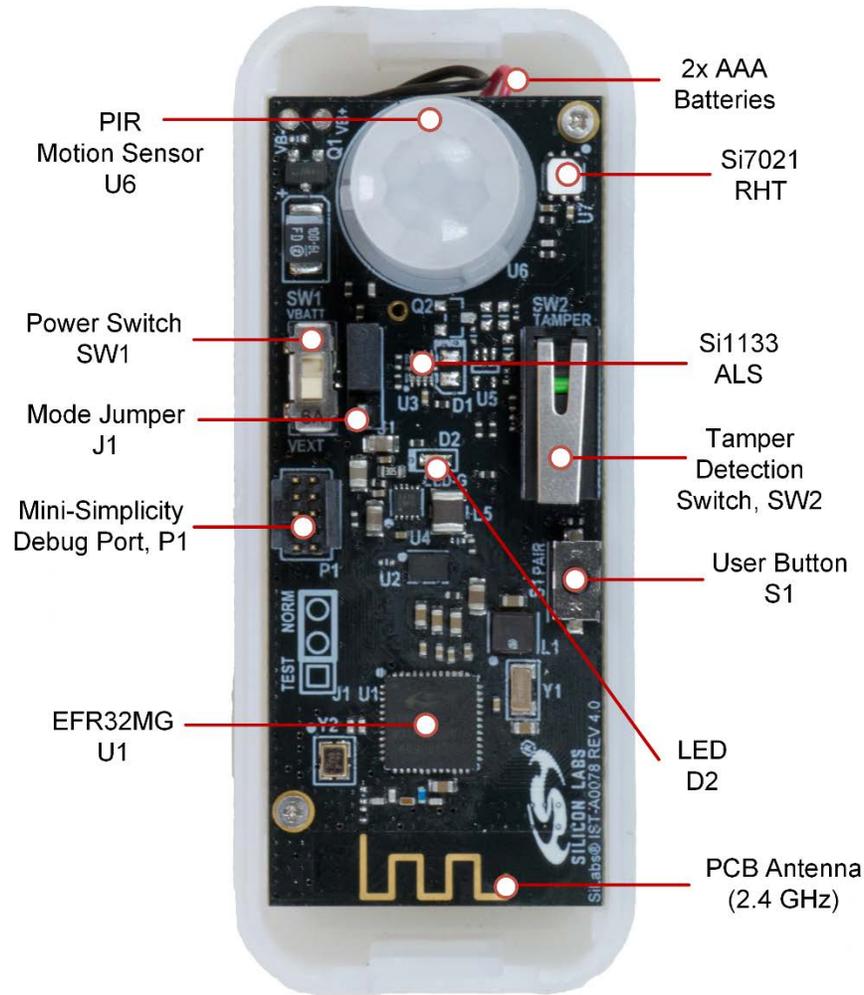
Figure 7. Reference Design

Table 5. Debug Port (Compatible with Mini Simplicity Connector)

Pin #	EFR32 Pin Name	Pin Function	Description
D1	VDD	VDD	Supply Voltage
D2	GND	GND	Ground
D3	RESETn	RESETn	Reset
D4	PA1	UART_RX	UART Receive
D5	PA0	UART_TX	UART Transmit

Pin #	EFR32 Pin Name	Pin Function	Description
D6	PF2	SW_O	JTAG Data Out, Serial Wire Out
D7	PF1	SW_DIO	JTAG Mode Select, Serial Wire Data In/Out
D8	PF0	SW_CLK	JTAG Clock, Serial Wire Clock
D9	PB13	PTI_SYNC	Packet Trace Framing Signal
D10	PB12	PTI_DATA	Packet Trace Data

6.2 Hardware Modification Guidelines

The following guidelines are provided for users wishing to modify the reference design.

6.2.1 Modifications to Avoid

In general, design modifications should be avoided, as they will affect the RF performance. The reference design as-shipped has had the RF circuitry tuned to account for the effects of PCB shape, antenna type, plastic housing, battery location and other considerations, therefore any changes to any aspect of the design can impact performance. Consequences of RF degradation include worsening ability to both transmit and receive over-the-air messages on the ZigBee network.

However, if some design modifications are required, we strongly recommend that the RF section, specifically the area immediately surrounding the EFR32MG chip and extending to the PCB edge around the antenna, not be modified in any way.

6.2.2 Bill of Material Cost-Saving Options

To further reduce overall bill-of-material (BOM) cost of this occupancy sensor, the following items and their supporting circuitry can be considered for removal with minor firmware changes:

- P1 programming header.
- U7 Si7021 relative humidity and temperature sensor.
- U3 Si1133 ambient light sensor.
- SW2 Tamper switch.

7 Firmware Details for the ZigBee Occupancy Sensor Reference Design

This section describes the Occupancy Sensor Reference Design application firmware.

7.1 Obtaining the Firmware Application

The firmware application in pre-compiled binary form is available for download from: http://www.silabs.com/zigbeeoccupancysensor. The firmware application source code is available as part of EmberZNet PRO v5.7.4.1 or later, which is available to registered users of a development kit.

For more information visit the ZigBee Getting Started page.

7.2 Programming the Occupancy Sensor Reference Design

The reference design provides two methods to reprogram the device:

- Using the Mini Simplicity header with an .s37, .bin, or .hex image file.
- Using the over-the-air (OTA) upgrade feature with an .ota image file.

7.2.1 Board Header Reprogramming

The occupancy sensor can be reprogrammed with an available .s37, .bin, or .hex file and a Mini Simplicity programmer. The Mini Simplicity Connector on the occupancy sensor is the P1 header. Notice the orientation of the connector, where the keyed side of the connector corresponds to the Mini Simplicity key marking found with the silkscreen drawing surrounding the header (see Figure 7. Reference Design).

When using the board header for programming, it is possible to update or replace the bootloader in addition to updating the application. A bootloader is a special piece of firmware that is primarily intended to allow future application firmware updates when a hardwired connection to the device is no longer available (allowing over-the-air firmware updates of deployed devices, for example). The reference design ships with a bootloader already installed, so you do not need to replace it. However, if you ever need to reload the bootloader, it is important to select the `app-bootloader-spiflash.s37` for the `EFR32MG1P232F256GM48` IC.

7.2.2 Over-the-Air Reprogramming

The occupancy sensor can be reprogrammed with an available .ota file and a device that can perform OTA upgrades, such as the RD-0001 or RD-0002 ZigBee Gateway Kits supported by Silicon Labs. Refer to the gateway documentation for more information on how to reprogram via OTA update.

7.3 Firmware Build Instructions

The instructions below describe how to build the device firmware using existing example code in the EmberZNet PRO stack.

1. Install EmberZNet PRO 5.7.4.1 or later.
2. Create a new Silicon Labs AppBuilder Project and select the Ember ZNet PRO SoC stack release.
3. Select the HaOccupancySensor sample application.
4. During Project Setup, verify that the proper chip (EFR32MG1P232F256GM48) is selected.
5. In AppBuilder, under the "hal configuration tab", verify the architecture (EFR32MG1P232F256GM48) and Board header (ist_a0078: Occupancy Sensor Reference Design) are properly selected.
6. Generate and note the directory in which the project files were created.
7. Save the project file into the directory you just created.
8. Compile in IAR (for version compatibility information see the EmberZNet PRO release notes and *QSG106: Getting Started with EmberZNet PRO*).

At this point you can load the image onto the Occupancy Sensor reference design using the Board Header Reprogramming method described above.

7.4   General Configuration

The device ID used is `0x0107`, which represents an occupancy sensor device, and the ZigBee Device Type is set to `Sleepy End Device`. The board header is set to `ist_a0078: Occupancy Sensor Reference Design.`

7.5   Cluster Support

This section details clusters that were implemented as part of this project.

Cluster Name	Cluster Hex Value
Basic	0x0000
Power Configuration	0x0001
Identify	0x0003
Over The Air Boot Loading	0x0019
Poll Control	0x0020
Illumination Measurement	0x0400
Illumination Level Sensing	0x0401
Temperature Measurement	0x0402
Relative Humidity Measurement	0x0405
Occupancy Sensing	0x0406
Diagnostics	0x0B05

Details about these clusters are given below.

Basic

This server cluster supports the attributes for determining basic information about a device, setting user device information such as location, and enabling a device. The following attributes are used:

- ZCL version
- Application version
- Stack version
- Hardware version
- Manufacturer name
- Model identifier
- Power source
- Cluster revision

The cluster also supports the `ResetToFactoryDefault` command (`RTFD`).

Power Configuration

This server cluster supports the attributes for determining more detailed information about a device's power source(s), and for configuring under/over voltage alarms. The following attributes are used:

- Battery voltage
- Battery percentage remaining
- Battery size
- Battery quantity
- Battery voltage min threshold
- Battery voltage threshold 1
- Battery voltage threshold 2
- Battery voltage threshold 3

Identify

This server and client cluster supports the attributes and commands for putting a device into Identification mode. The following server attributes are used:

- Identify time
- Cluster revision

The server cluster supports the `Identify` and `IdentifyQuery` commands. The client cluster supports the `IdentifyQueryResponse` command.

Over-The-Air (OTA) Bootloading

This client cluster contains commands and attributes that act as an interface for ZigBee Over-the-air bootloading. The following attributes are used:

- OTA Upgrade Server ID
- Offset (address) into the file
- OTA Current File Version
- OTA Downloaded File Version
- OTA Upgrade Status
- Manufacturer ID
- Image Type ID
- Minimum Block Request Period
- Image Stamp

The following commands are used:

- `QueryNextImageRequest`
- `ImageBlockRequest`
- `UpgradeEndRequest`

Poll Control

This cluster provides a mechanism for the management of an end device's MAC Data Poll rate. For the purposes of this cluster, the term "poll" always refers to sending a MAC Data Poll from the end device to the end device's parent. The following attributes are used:

- Check-in interval
- Long poll interval
- Short poll interval
- Fast poll timeout
- Check in interval min
- Long poll interval min
- Fast poll timeout max
- Cluster revision

The client command is used: `CheckIn`.

Illuminance Measurement

This server cluster supports the attributes and commands for configuring the measurement of illuminance, and reporting illuminance measurements. The following attributes are used:

- Measured value
- Min measured value
- Max measured value
- Cluster revision

Temperature Measurement

This server cluster supports the attributes and commands for configuring the measurement of temperature, and reporting temperature measurements. The following attributes are used:

- Measured value
- Min measured value
- Max measured value
- Cluster revision

Relative Humidity Measurement

This server cluster supports the attributes and commands for configuring the measurement of relative humidity, and reporting relative humidity measurements. The following attributes are used:

- Measured value
- Min measured value
- Max measured value
- Cluster revision

Occupancy Sensing

This server cluster supports the attributes and commands for configuring occupancy sensing, and reporting occupancy status. The following attributes are used:

- Occupancy
- Occupancy sensor type
- Cluster revision

7.6 Command Line Interface Support

Besides the General CLI, the following CLI command summaries are supported:

- address-table
- connection-manager
- eeprom
- end-device-support
- ezmode-commissioning
- identify
- idle-sleep
- illuminance-measurement-server
- mfglib
- occupancy-pyd1698-cli
- ota-bootload
- ota-client
- ota-storage-common
- ota-storage-eeprom
- relative-humidity-measurement-server
- reporting
- temperature-measurement-server

Help for a plugin listed above can be shown by entering the command plugin <plugin name>.

7.7 Included Plugins

Numerous plugins are used from the EmberZNET PRO stack. Notable ones are described in this section, along with their set options.

Battery Monitor

This plugin monitors the value of the battery during the middle of radio transmissions, and uses that data to calculate the voltage level of the battery. The following options are set:

- Monitor Timeout (Minutes) [0-1000] = 30
- Sample Collection FIFO size [1-20] = 16

Button Interface

This plugin gives additional button press tracking functionality on top of the HAL button API. It generates callbacks dependent on the amount of time a button was pressed, allowing a user to specify the difference between a short press and a long press, and to receive callbacks when those conditions are met. The following option is set:

- Button Timeout (milliseconds) [1-16534]: 10000

Illuminance Si1141

This plugin reads from a Si1141 illuminance sensor. The following option is set:

- Number of samples [1-200] = 5

Occupancy PYD-1698

This plugin provides an interface for the PYD-1698 occupancy sensor. The PYD-1698 communicates using a proprietary serial interface, so direct register access to GPIO registers and 4-10 cycle processor waits are used to bit bang the interface. The following options are set:

- Detection Threshold [0-255] = 8
- Hardware Blind Time [1-16] = 16
- Pulse Counter [1-4] = 1
- Window Time [0-3] = 1
- Filter Source [0-1] = 0
- Operation Mode [0-2] = 2
- Occupancy Timeout [0-65535] = 10
- Calibration Occupancy Timeout [0-16383] = 8
- Firmware Blind Time [0-65535] = 0

Relative Humidity Si7021

This plugin reads from a Si7021 humidity sensor. The following option is set:

- Measurement Accuracy [8,10, 11, 12] = 12

Temperature Si7021

This plugin reads from a Si7021 temperature sensor. This plugin has no settable options.

Occupancy Sensor Server Cluster

This plugin implements the Occupancy Sensor server cluster. It uses callbacks from a HAL occupancy sensor plugin to populate the Occupancy and Occupancy Sensor Type attributes of the Occupancy Sensor cluster. The following option is set:

- Maximum Report Period (seconds) [1-165434] = 1800

Relative Humidity Measurement Server Cluster

This plugin implements the Relative Humidity Measurement server cluster. It periodically polls a humidity sensor and uses that value to populate the Measured Value, Min Measured Value, and Max Measured Value attributes of the Relative Humidity Measurement Cluster. The following options are set:

- Maximum Measurement Frequency (seconds) [1-16534] = 1800
- Default Reportable Change decipercentage [0-10000] = 50

Temperature Measurement Server Cluster

This plugin implements the Temperature Measurement server cluster. It periodically polls a temperature sensor and uses that value to populate the Measured Value, Min Measured Value, and Max Measured Value attributes of the Temperature Measurement Cluster. The following options are set:

- Maximum Measurement Frequency (seconds) [1-16534] = 1800
- Default Reportable Change (millidegrees Celsius) [0-32767] = 500

Silicon Labs Device UI

This plugin implements a user interface for a Silicon Labs device. It uses a single button to implement joining and leaving a network and an LED to indicate network activity (such as searching for, joining, and leaving a network). This plugin can optionally be used along with the EZ-Mode commissioning plugin, in which case the EZ-Mode role option should be used to set whether the device is the initiator or target during commissioning. If the device will be used as an initiator, the application should use the `emberAfPluginSilabsDeviceUiSetEzModeClusters` API to set the clusters for which binding table entries will be created during the commissioning process. The following options are set:

- Consecutive Press Timeout [0-65535] = 500
- Button Debounce Time [0-65535] = 50
- Number of Blinks on Join [0-100] = 6
- Number of Blinks on Leave [0-100] = 3
- Role of device in EZ-Mode (if supported) [0-1] = 0
- EZ-Mode Cluster Direction [0-1] = 0
- EZ-Mode Maximum Number of Clusters [1-255] = 3
- EZ-Mode Endpoint [1-255] = 1

7.8 Mfglib CLI

To assist with the manufacturing process, the occupancy sensor reference designs includes the manufacturing library as well as the manufacturing library CLI commands. On the factory floor, a unit that has been programmed with these images may also be used to make radio measurements.

7.8.1 Theory of Operation

The purpose of the manufacturing library CLI plugin (mfglib CLI plugin) is to allow access to commands that put the radio into test mode, so that the radio modules can be tested at manufacturing time with the final occupancy firmware images.

7.8.2 CLI Commands

Table 6. Commands for the Manufacturing Library

CLI Command	Notes
`plugin mfglib mfgenable <0│1>`	Sets the token to pause the Device UI plugin from scanning for channels to allow for starting the manufacturing library before the stack operation begins. 0 clears the token, 1 sets the token. When the token is set, the application has 10 seconds after the next power-up to preempt the stack from resuming normal operation and start the mfglib.
`Plugin mfglib start <0│1>`	Starts the manufacturing library, which prevents network activity as well as enables the rest of the manufacturing library commands. The argument dictates whether or not to track incoming messages and provide receiver statistics on the incoming packets.
`Plugin mfglib stop`	Stops the manufacturing library.
`Plugin mfglib set-channel <channel>`	Sets the radio to the specified channel. **Note:** The channel must be between 11 and 26 inclusive.

CLI Command	Notes
Plugin mfglib set-power <power> <mode>	Sets the radio power to be used for manufacturing library transmit commands. The <mode> enables or disables boost mode for the EM35x series of chips.
plugin mfglib stream <start|stop>	Sets the radio to transmit a modulated carrier. **Note:** The carrier will be modulated with a random stream of characters.
plugin mfglib tone <start|stop>	Sets the radio to transmit a carrier.

Note: Additional CLI commands are available for use with the manufacturing library plugin. They can be shown by entering the command `plugin mfglib`. The commands listed in the table above are the most commonly used with an occupancy sensor reference design during the manufacturing process.

As an example in using these commands, here is the procedure for starting the manufacturing library:

1.   Connect to the reference design and issue the mfgenable command: `plugin mfglib mfgenable 1`

2.   Power cycle the reference design.

3.   Within 10 seconds of power cycle, issue the start command `plugin mfglib start 0`

At this point, any of the manufacturing library commands can be safely executed without interference from the networking stack.

For example, to take measurements on a modulated carrier on channel 15 at a power of 20, issue the following commands:

```
plugin mfglib set-channel 15
plugin mfglib set-power 20 0
plugin mfglib stream start
```

Now take your measurements.

```
plugin mfglib stream stop
```

Note: A transmit power of 20 is only valid for certain variants of the EFR32 SoC at this time.

When the manufacturing tests are completed, it is important to disable the manufacturing library delay at bootup with the following command:

```
plugin mfglib mfgenable 0
```

It is not necessary to issue the manufacturing library stop command unless you wish to initiate normal networking activity without a power cycle, as the manufacturing library operation will be reset at the next power cycle.

7.9   Manufacturing Library – OTA

This plugin implements the manufacturing library custom cluster from Silicon Labs. It provides a means, through radio messages, to put the device into RF testing mode. In the past, enabling such features on a device required the user to physically alter the device (by cutting the case, adding a connector, and connecting to a ribbon cable). For some tests, this is almost impossible in practice. For others, it will alter the results of the test (such as for FCC tests). To overcome these issues, Silicon Labs developed a way to invoke the manufacturing library commands using radio commands.

7.9.1   Theory of Operation

Silicon Laboratories has created a custom ZigBee cluster that provides RF commands for the purpose of putting the device receiving the commands into manufacturing test mode. Specifically, it can force the device to transmit an unmodulated tone, transmit a modulated tone, or receive packets for a short time to check the TX or RX path of the device in the field.

First, the user must obtain a gateway that supports this functionality. Our HaGatewayReference application supports the MFGLIB Cluster, although it is possible to use our development environment to create such a gateway.

After the gateway creates the network and the occupancy sensor reference design joins the network, the gateway can send commands to the occupancy sensor reference design to tell it which RF test to perform and which parameters to use, such as channel and TX power.

Each of the RF commands also includes a timeout parameter, which sets the amount of time (in milliseconds) to enable the command. Using a timeout parameter of 0 means that the device will remain in the specified RF test until the next device power cycle.

7.9.2   CLI Commands

Table 7. Manufacturing Library Over-the-Air Commands

OTA Command	Notes
`zcl mfg-code 0x1002`	Sets the manufacturing code to 0x1002, which is the Silicon Labs code used for these custom clusters.
`zcl mfg-code 0`	Clears the manufacturing code.
`zcl mfglib rx-mode <channel> <power> <time>`	Puts the device into RX mode. **Note:** When in RX mode, the device under test keeps track of the number of packets received, the RSSI and the LQI of the first packet received. It is recommended to use a non-zero timeout as these parameters are stored in volatile memory and will be reset at power cycle
`zcl mfglib stream <channel> <power> <time>`	Sets the radio to transmit a modulated carrier. **Note:** The carrier will be modulated with a random stream of characters.
`zcl mfglib tone <channel> <power> <time>`	Sets the radio to transmit a carrier.

The following is an example of the commands used to order the reference design to transmit a modulated carrier on channel 15 with a power of 13 dBm until the next device power cycle.

Note: It is assumed the gateway already created a network and that the occupancy sensor reference design has joined this network with device ID 0x1234.

```
zcl mfg-code 0x1002
zcl mfglib-stream 15 13 0
send 0x1234 1 1
```

8 Ecosystem Considerations

In this reference design the firmware examples are ZigBee HA-certifiable. As such, they should work with any of the ZigBee-certified Home Automation gateways in the market. However, each gateway may have some individual requirements beyond the ZigBee Home Automation requirements. Contact the individual gateway ecosystem provider for support in getting your device onto their ecosystem, as several of these ecosystem providers have developer websites and communities to facilitate the effort.

In our testing, we have discovered that many gateway manufacturers require an end device to support two optional attributes in the Basic cluster: manufacturer name and model identifier. While our reference designs have values for these attributes that are reflective of the Silicon Laboratories reference design work, it is expected that each customer would customize these attributes for their own product.

In our testing, we have also discovered that a few of the gateway manufacturers use the manufacturer name and model identifier (among other things) to implement a white list. A white list is a list of approved devices that the gateway will allow onto its network. If your device is not on the white list, it will not be allowed to join the network. For these gateways, you must contact the gateway platform provider directly to inquire how to add your manufacturer name and model identifier to their gateway's white list.

9 Engineering Tests

This reference design completed the following product testing for pre-certification purposes:

- ZigBee Home Automation (HA) v1.2
- FCC Emissions
- Antenna Radiation Patterns

9.1 ZigBee Home Automation HA v1.2

The occupancy sensor went through a preliminary HA 1.2 testing to verify compliance with the ZigBee standard. Note that, as of this writing, the ZigBee alliance has not ratified the specification for the illuminance measurement and relative humidity measurement clusters, and as a result the verifiability of these clusters to HA 1.2 was not tested.

9.2 FCC Emissions Testing

This design went through a preliminary FCC pre-scan testing to verify compliance with FCC Part 15 restrictions. Such testing is done to ensure customers are given a design that can pass FCC tests. FCC compliance does not transfer to modules built according to this reference design, even if the design is copied exactly.

10 Manufacturing Tests

This section describes how a user can develop a low-cost manufacturing test methodology in order to test products that utilize this reference design. Recommended test equipment and test procedures will also be outlined in this section.

10.1 Test Coverage

By following these guidelines, the following items will be tested and verified for functionality:

- RF TX/RX performance
- Current consumption
- Motion sense functionality
- Sensor ICs functionality
- Common ZigBee operations
- Frequency offset (with optional spectrum analyzer)

10.2 Test Equipment List

The test system is composed of the following components:

- Desktop PC
- RF Shielded Box
- Power Supply (Agilent E3646A + Ammeter)
- 2x Silicon Labs Wireless Starter Kits (WSTK) with Mini Simplicity programming/packet trace adapters
- Golden Node (Silicon Labs ZigBee device that can act as a coordinator)
- Frequency Counter (HP 53131A)
- Device Under Test (DUT)

10.3 Test System Diagram
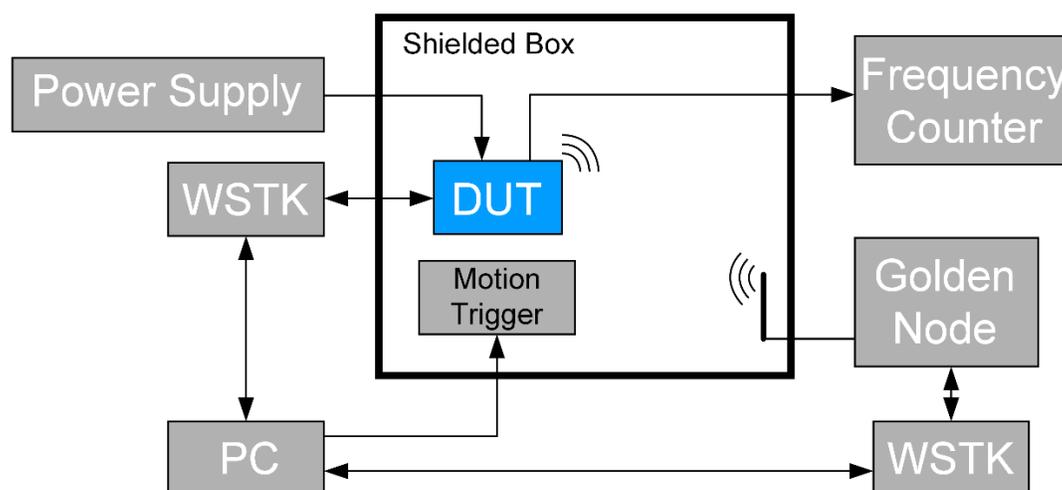
Figure 8. Test System Diagram

10.4 Test System Connection Procedure

Wire the system as indicated above. All WSTK adapters are connected to the PC through USB. One adapter connects to the DUT to both reprogram and interact with the DUT, and the second one connects to the Golden Node for testing network functionality. The WSTK adapters may need to be powered off during some tests. This can be done by adding a power switch to the test jig, or by using the power selection switch already on the WSTK.

Within the shielded box, the only items protruding inward should be the connectors linked to the DUT, the motion trigger, and connectors for 50 Ω antennas. If desired, a spectrum analyzer may be procured and integrated into the system to measure frequency offset.

A controllable varying temperature element can be used to verify correct installation of the PIR motion sensor.

10.5   Configuring the WSTKs

For this setup, the PC will communicate to the WSTK with a serial connection. The WSTK enumerate as a "COM Port" device when connected to the PC via USB without any additional setup.

10.6   Communicating with Targets

For testing purposes, the NodeTest firmware image is used on the target DUT. The PC communicates with the DUT's UART interface through the WSTK's COM port. Use a serial connection tool, such as PuTTY, to start a serial console connection to the WSTK's enumerated COM port with the following settings:

- Speed (baud): 115200
- Hardware flow control: enabled
- Data bits: 8
- Parity: none
- Stop bits: 1

Once connected, pressing Enter or sending a "\n" character results in a prompt ending with a ">" character. Type "help" and then press Enter to display a list of available test functions in Test firmware. See *AN1019: Using the NodeTest Application* for more information.
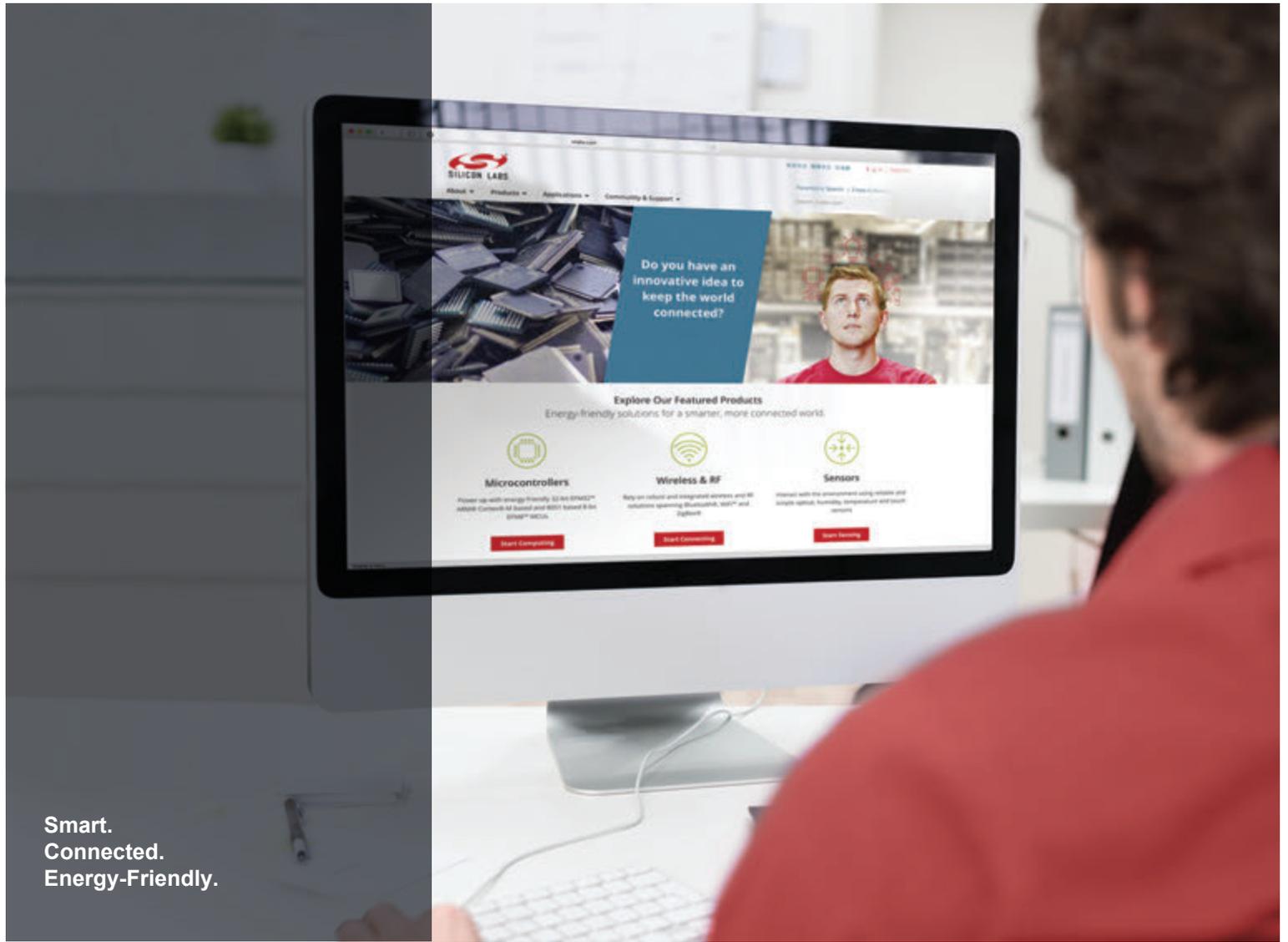
**Smart.
Connected.
Energy-Friendly.**

Products
www.silabs.com/products
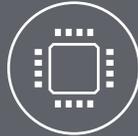
Quality
www.silabs.com/quality

Support and Community
community.silabs.com

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com