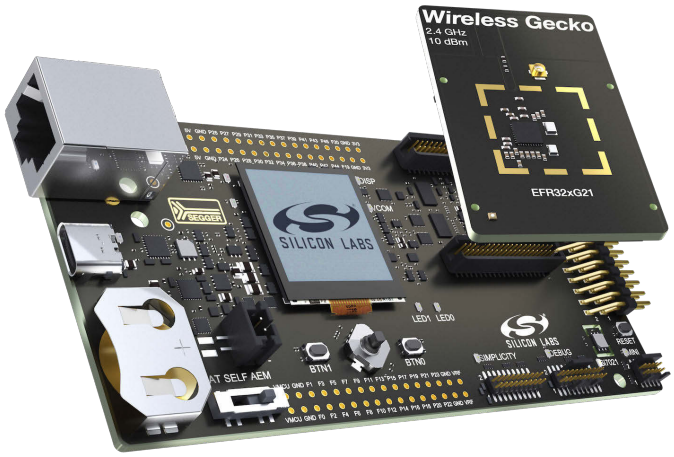
UG429: EFR32xG21 2.4 GHz 10 dBm Radio Board User's Guide

A Wireless Starter Kit with the BRD4181B Radio Board is an excellent starting point to get familiar with the EFR32 Wireless Gecko Wireless System-on-Chip. It also provides all necessary tools for developing a Silicon Labs wireless application.

BRD4181B is a plug-in board for the Wireless Starter Kit Mainboard (BRD4001A) and the Wireless Pro Kit Mainboard (BRD4002A). It is a complete reference design for the EFR32xG21 Wireless SoC, with matching network and a PCB antenna for 10 dBm output power in the 2.4 GHz band.

The mainboards contain an on-board J-Link debugger with a Packet Trace Interface and a Virtual COM port, enabling application development and debugging of the attached radio board as well as external hardware. The mainboards also contain sensors and peripherals for easy demonstration of some of the EFR32's many capabilities.

This document describes how to use the BRD4181B Radio Board together with a Wireless Starter Kit Mainboard or a Wireless Pro Kit Mainboard.



BRD4181B RADIO BOARD FEATURES

- EFR32xG21 Wireless Gecko Wireless SoC with 1024 kB Flash and 96 kB RAM (EFR32MG21A010F1024IM32).
- Inverted-F PCB antenna (2.4 GHz band)
- 2x user color LEDs (red and green)

MAINBOARD FEATURES

- Advanced Energy Monitor
- Packet Trace Interface
- Logic analyzer (BRD4002A only)
- Virtual COM port
- SEGGER J-Link on-board debugger
- External device debugging
- Ethernet and USB connectivity
- Low-power 128x128 pixel Memory LCD-TFT
- User LEDs / pushbuttons
- Joystick (BRD4002A only)
- 20-pin 2.54 mm EXP header
- Breakout pads for Wireless SoC I/O
- CR2032 coin cell battery support

SOFTWARE SUPPORT

- Simplicity Studio
- Energy Profiler
- Network Analyzer

ORDERING INFORMATION

- SLWSTK6006A
- SLWSTK6023A
- SLWRB4181B

 Copyright © 2022 by Silicon Laboratories

Downloaded from Arrow.com.

Table of Contents

1. Introduction . . . 4

1.1 Radio Boards . . . 4

1.2 Mainboards . . . 4

1.3 Ordering Information . . . 5

1.4 Getting Started . . . 5

2. Hardware Overview . . . 6

2.1 Hardware Layout . . . 6

2.2 Block Diagram . . . 7

3. Connectors . . . 8

3.1 J-Link USB Connector . . . 8

3.2 Ethernet Connector . . . 8

3.3 Breakout Pads . . . 9

3.4 EXP Header . . . 11

3.4.1 EXP Header Pinout . . . 12

3.5 Logic Analyzer Connector . . . 13

3.6 Debug Connector . . . 14

3.7 Simplicity Connector . . . 15

3.8 Mini Simplicity Connector . . . 16

3.9 Debug Adapter . . . 17

4. Power Supply and Reset . . . 18

4.1 Radio Board Power Selection . . . 18

4.2 Kit Power . . . 19

4.2.1 Board Controller Power . . . 19

4.2.2 AEM Power . . . 19

4.3 EFR32 Reset . . . 19

4.4 Battery Holder . . . 20

5. Peripherals . . . 21

5.1 Push Buttons and LEDs . . . 21

5.2 Joystick . . . 22

5.3 Memory LCD-TFT Display . . . 23

5.4 Virtual COM Port . . . 24

5.4.1 Host Interfaces . . . 25

5.4.2 Serial Configuration . . . 25

5.4.3 Hardware Handshake . . . 26

6. Board Controller . . . 27

6.1 Introduction . . . 27

Downloaded from Arrow.com.

6.2 Admin Console 27
6.2.1 Connecting 27
6.2.2 Built-in Help 28
6.2.3 Command Examples 28
6.3 Virtual UART 28
6.3.1 Target-to-Host 28
6.3.2 Host-to-Target 29
6.3.3 Limitations 29
6.3.4 Troubleshooting 29
7. Advanced Energy Monitor 30
7.1 Introduction 30
7.2 Code Correlation 30
7.3 AEM Circuit 30
7.3.1 AEM Details 31
8. On-Board Debugger 32
8.1 Host Interfaces 32
8.1.1 USB Interface 32
8.1.2 Ethernet Interface 32
8.1.3 Serial Number Identification 32
8.2 Debug Modes 33
8.3 Debugging During Battery Operation 34
9. Kit Configuration and Upgrades 35
9.1 Firmware Upgrades 35
10. Schematics, Assembly Drawings, and BOM 36
11. Kit Revision History 37
11.1 SLWRB4181B Revision History 37
11.2 SLWSTK6023A Revision History 37
11.3 SLWSTK6006A Revision History 38
12. Document Revision History 39

Downloaded from Arrow.com.

1. Introduction

The EFR32xG21 Wireless Gecko Wireless SoC is featured on a radio board that plugs directly into a Wireless Starter Kit (Wireless STK) Mainboard or a Wireless Pro Kit Mainboard. The mainboards feature several tools for easy evaluation and development of wireless applications. An on-board J-Link debugger enables programming and debugging on the target device over USB or Ethernet. The Advanced Energy Monitor (AEM) offers real-time current and voltage monitoring. A virtual COM port interface (VCOM) provides an easy-to-use serial port connection over USB or Ethernet. The Packet Trace Interface (PTI) offers invaluable debug information about transmitted and received packets in wireless links. All debug functionality, including AEM, VCOM, and PTI, can also be used towards external target hardware instead of the attached radio board.

To further enhance its usability, the mainboard contains sensors and peripherals that demonstrate some of the many capabilities of the EFR32xG21. The mainboard also has a 20-pin EXP header which can be used for connecting EXP boards to the kit or for easy connection to I/Os on the radio board target IC.

1.1 Radio Boards

A Wireless Starter Kit consists of one or more mainboards and radio boards that plug into the mainboard. Different radio boards are available, each featuring different Silicon Labs devices with different operating frequency bands. Because the mainboards are designed to work with different radio boards, the actual pin mapping from a device pin to a mainboard feature is done on the radio board. This means that each radio board has its own pin mapping to the Wireless STK features, such as buttons, LEDs, the display, the EXP header, and the breakout pads. Because this pin mapping is different for every radio board, it is important to consult the correct document, which shows the kit features in context of the radio board plugged in.

1.2 Mainboards

The EFR32xG21 2.4 GHz 10 dBm Radio Board (BRD4181B) can be used with either a Wireless Starter Kit Mainboard (BRD4001A) or a Wireless Pro Kit Mainboard (BRD4002A). The Wireless Pro Kit Mainboard is the successor to the Wireless Starter Kit Mainboard, which comes with some improvements and added features including increased AEM measurement range and sample rate, variable VMCU voltage, joystick, a logic analyzer, and a Mini Simplicity Connector. Kit features, such as the Si7021 sensor and the EXP header, are available on the same EFR32xG21 pins regardless of the mainboard being used, but the pinout to the breakout pads differs. The combination of the EFR32xG21 2.4 GHz 10 dBm Radio Board with either one of these mainboards is hereby referred to as a Wireless Starter Kit as the figure below illustrates.

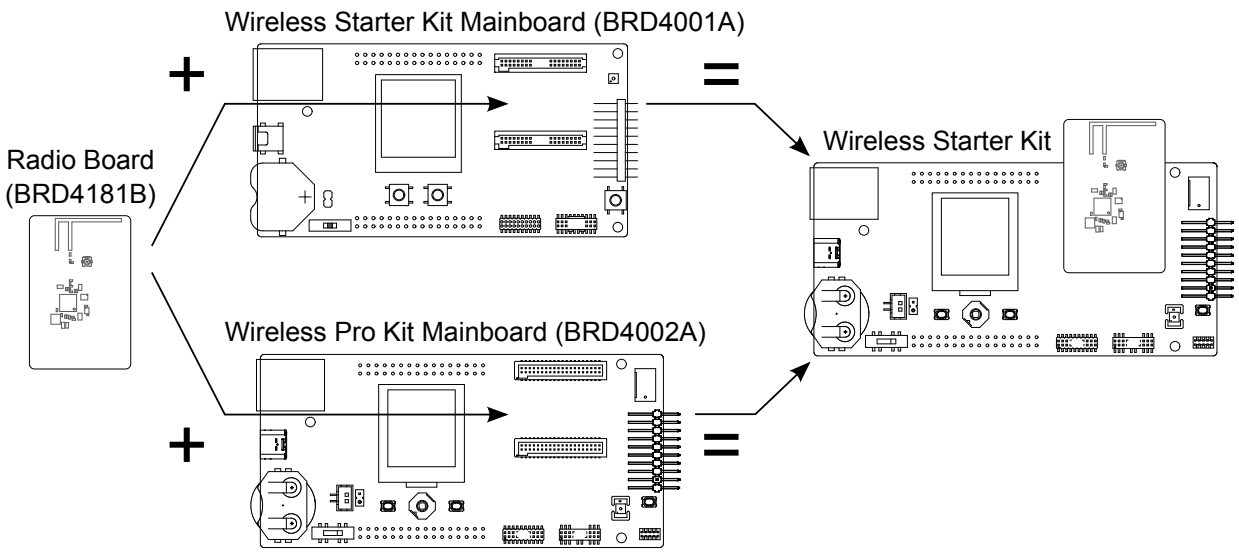


Figure 1.1. Wireless STK Combinations

Note: This document explains how to use the Wireless STK when the EFR32xG21 2.4 GHz 10 dBm Radio Board (BRD4181B) is combined with either a Wireless Starter Kit Mainboard (BRD4001A) or a Wireless Pro Kit Mainboard (BRD4002A). Since some of the functionality of the kit depends on the type of mainboard used, it is important to consult the right information in the user guide whenever there are discrepancies.

Downloaded from Arrow.com.

1.3 Ordering Information

BRD4181B can be obtained as part of SLWSTK6006A EFR32xG21 2.4 GHz Mesh Networking Starter Kit, SLWSTK6023A EFR32xG21 Bluetooth Starter Kit, or as a separate radio board, SLWRB4181B.

Table 1.1. Ordering Information

Part Number	Description	Contents
SLWSTK6006A	EFR32xG21 2.4 GHz Mesh Networking Starter Kit	3x BRD4002A Wireless Pro Kit Mainboard 3x BRD4180B EFR32xG21 2.4 GHz 20 dBm Radio Board 3x BRD4181B EFR32xG21 2.4 GHz 10 dBm Radio Board 3x AA battery holders 1x 10-pin 1.27 mm IDC debug cable
SLWSTK6023A	EFR32xG21 Bluetooth Starter Kit	1x BRD4002A Wireless Pro Kit Mainboard 1x BRD4181B EFR32xG21 2.4 GHz 10 dBm Radio Board 1x 10-pin 1.27 mm IDC debug cable
SLWRB4181B	EFR32xG21 2.4 GHz 10 dBm Radio Board	1x BRD4181B EFR32xG21 2.4 GHz 10 dBm Radio Board

Note: Kit content in the table refers to SLWSTK6006A Rev. C00, SLWSTK6023A Rev. B00, and SLWRB4181B Rev. A04 and may vary between revisions. For information about kit revision changes, see Section 11. Kit Revision History. The type of mainboard (BRD4002A or BRD4001A) included in SLWSTK6006A and SLWSTK6023A depends on the kit revision.

1.4 Getting Started

Detailed instructions for how to get started can be found on the Silicon Labs web pages: http://www.silabs.com/dev-tools.

Downloaded from Arrow.com.

2. Hardware Overview

2.1 Hardware Layout

The layout of the EFR32xG21 2.4 GHz 10 dBm Wireless STK when the radio board is combined with a Wireless Pro Kit Mainboard (BRD4002A) or a Wireless STK Mainboard (BRD4001A) is shown below.

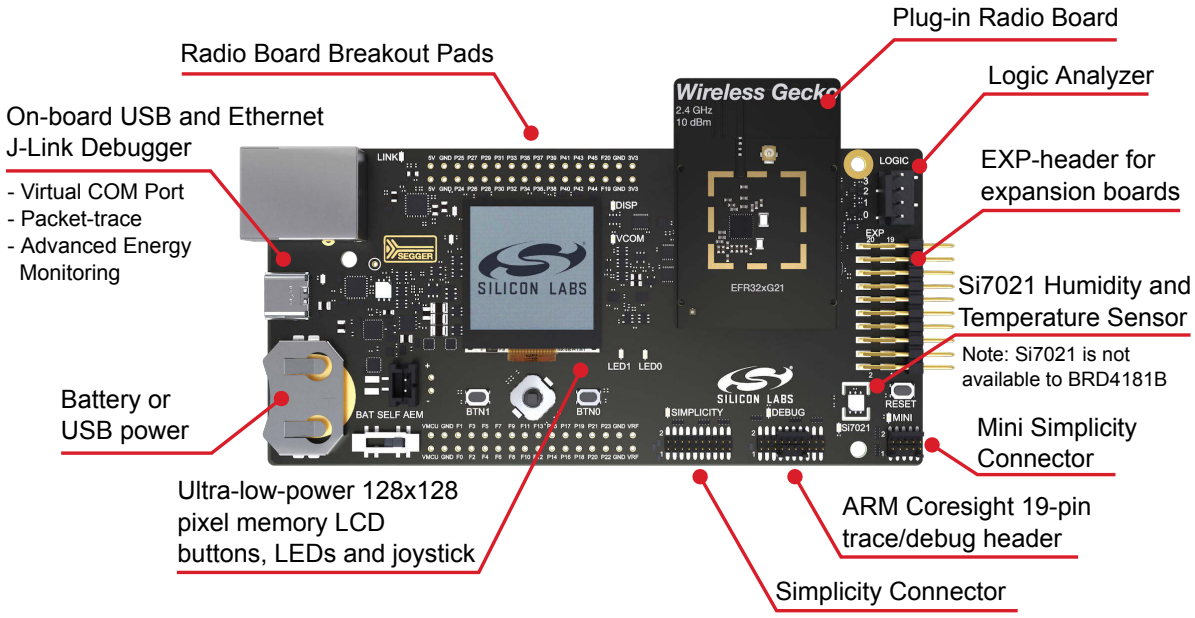


Figure 2.1. Hardware Layout With A Wireless Pro Kit Mainboard (BRD4002A)

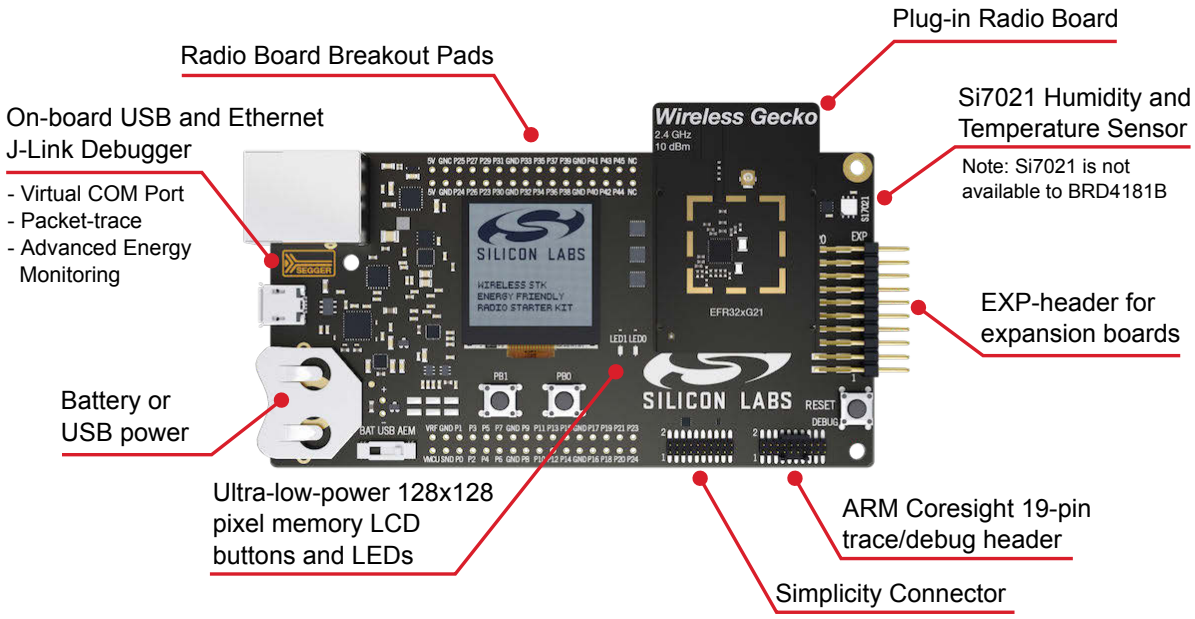


Figure 2.2. Hardware Layout With A Wireless STK Mainboard (BRD4001A)

Downloaded from Arrow.com.

2.2 Block Diagram

An overview of the EFR32xG21 2.4 GHz 10 dBm Wireless STK is shown in the figure below.

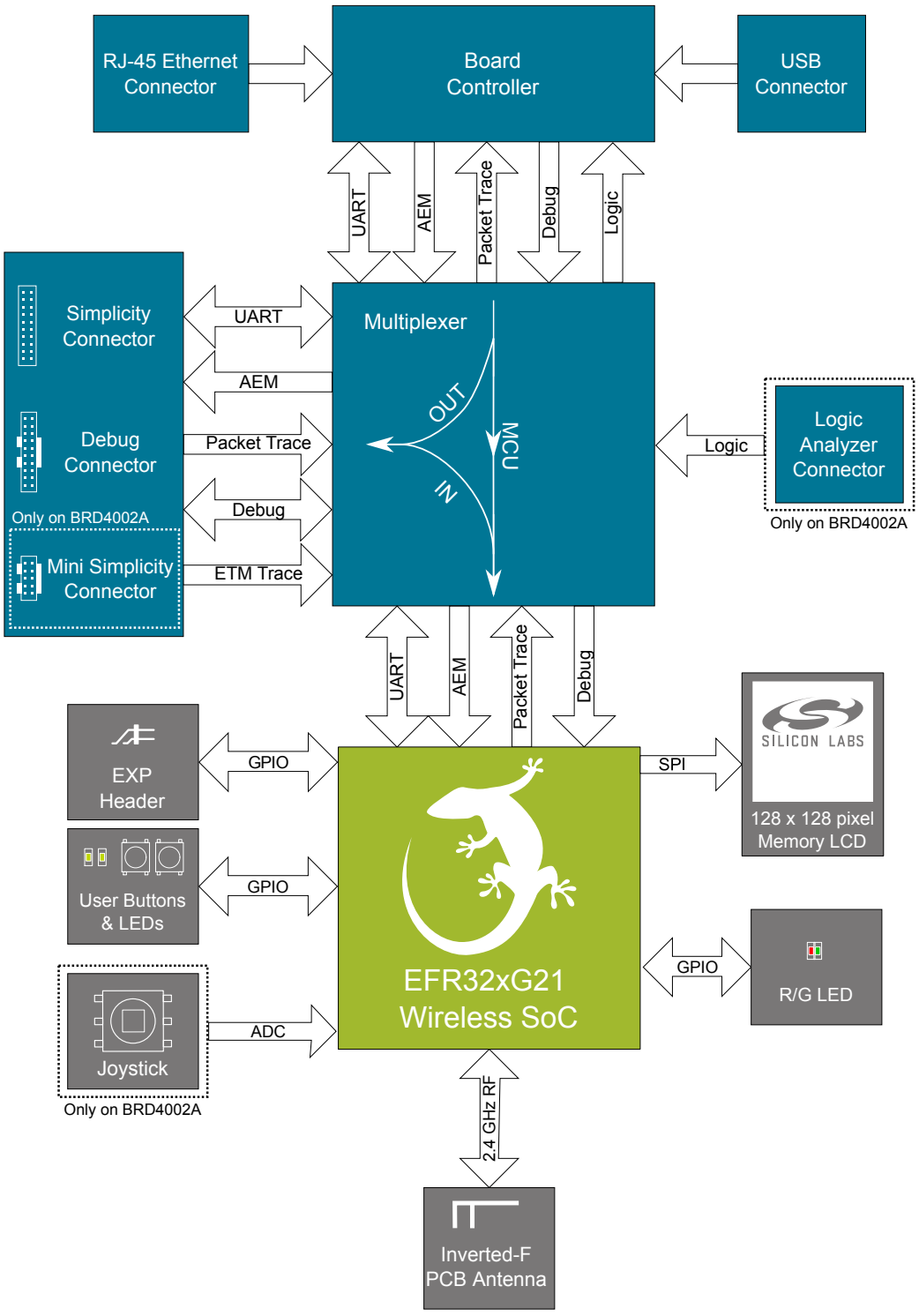


Figure 2.3. Kit Block Diagram

Downloaded from Arrow.com.

3. Connectors

This chapter gives you an overview of the mainboard connectivity. The placement of the connectors on the Wireless Pro Kit Mainboard (BRD4002A) and the Wireless STK Mainboard (BRD4001A) is shown below.

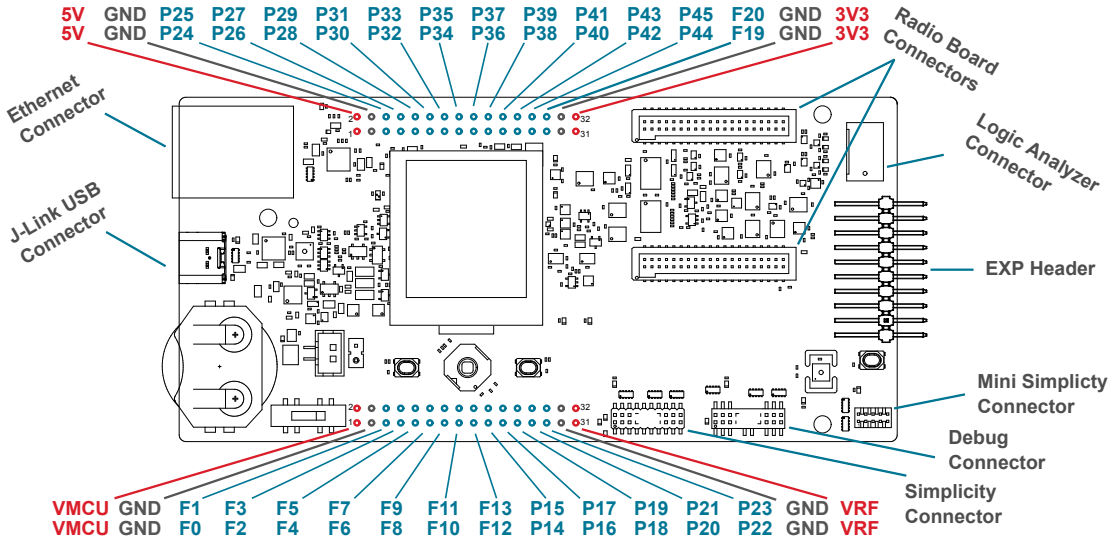


Figure 3.1. Wireless Pro Kit Mainboard (BRD4002A) Connector Layout

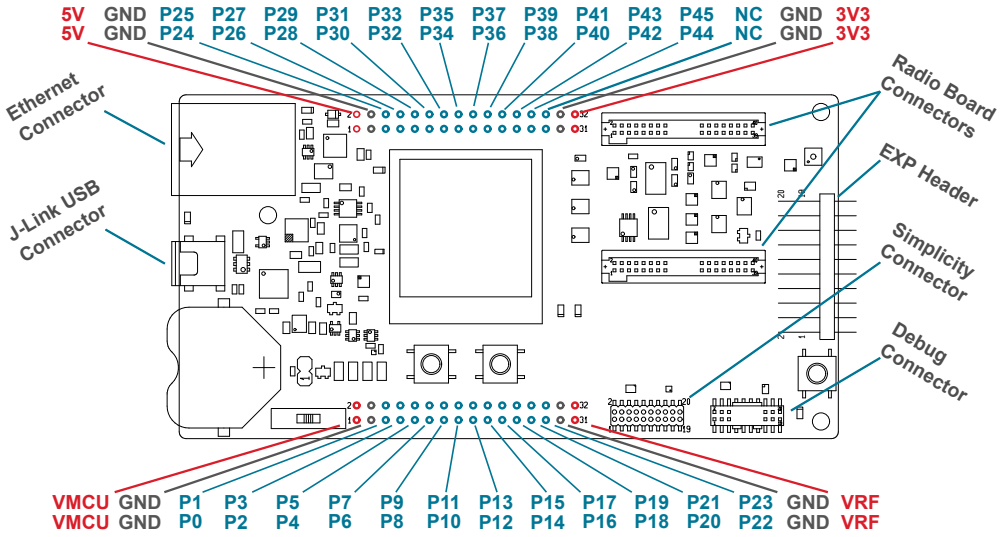


Figure 3.2. Wireless STK Mainboard (BRD4001A) Connector Layout

3.1 J-Link USB Connector

The J-Link USB connector is situated on the left side of the mainboard and provides access to the kit features described in Section 6. Board Controller through the USB interface. In addition to providing access to development features of the kit, this USB connector is also the main power source for the kit powering both the board controller and the AEM as described in Section 4. Power Supply and Reset.

3.2 Ethernet Connector

The Ethernet connector is situated on the left side of the mainboard and provides access to the kit features described in Section 6. Board Controller over TCP/IP. The J-Link USB connector must be connected while using this interface to provide power to the Wireless STK as power is not supplied over the Ethernet connector.

Downloaded from Arrow.com.

3.3 Breakout Pads

Most of the EFR32 pins are routed from the radio board to breakout pads at the top and bottom edges of the mainboard. A 2.54 mm pitch pin header can be soldered on for easy access to the pins. The figures below show how the pins of the EFR32 map to the pin numbers printed on the breakout pads on the Wireless Pro Kit Mainboard (BRD4002A) and the Wireless STK Mainboard (BRD4001A). To see the available functions on each pin, refer to the data sheet for EFR32MG21A010F1024IM32.

Note: Pinout to the breakout pads depends on the mainboard being used.

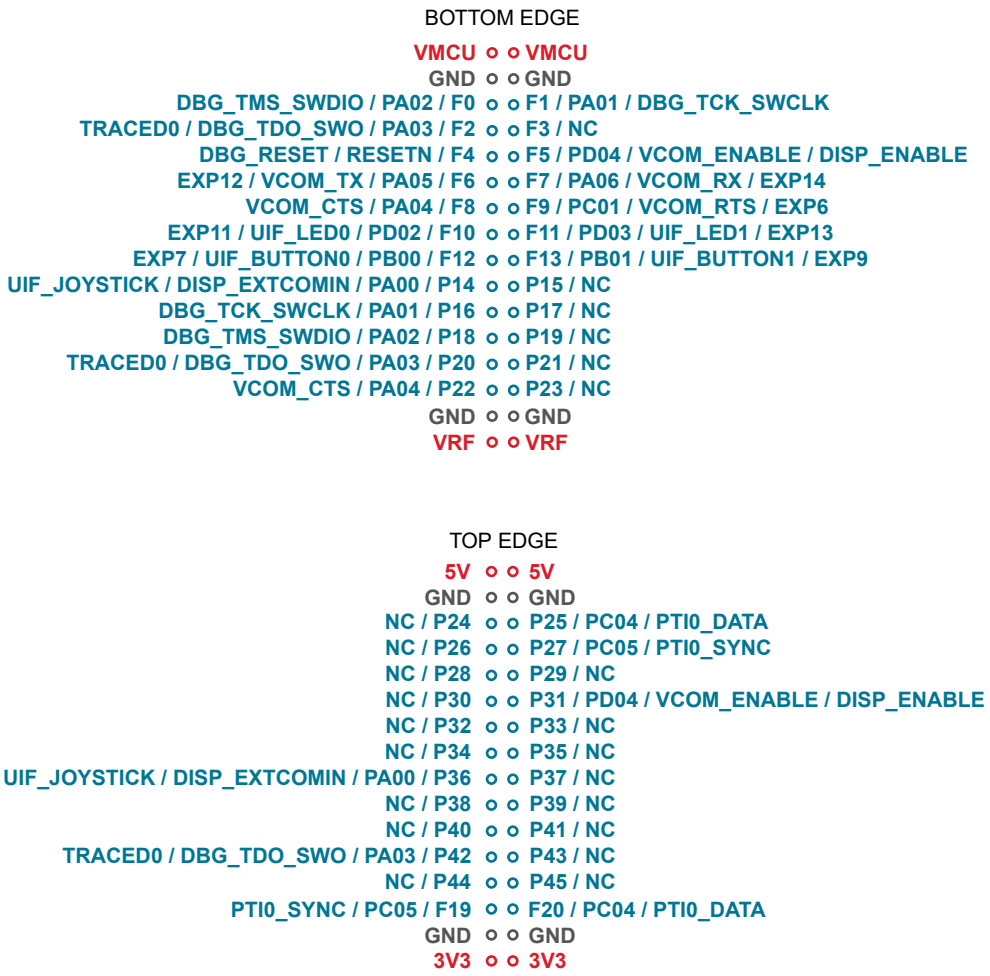


Figure 3.3. Wireless Pro Kit Mainboard (BRD4002A) Breakout Pad Pin Mapping

Downloaded from Arrow.com.



Figure 3.4. Wireless STK Mainboard (BRD4001A) Breakout Pad Pin Mapping

Downloaded from Arrow.com.

3.4 EXP Header

The EXP header is an angled, 20-pin expansion header that allows connection of peripherals or plugin boards to the kit. It is located on the right-hand side of the mainboard and contains several I/O pins that can be used with most of the EFR32 Wireless Gecko's features. Additionally, the VMCU, 3V3, and 5V power rails are also exposed.

The connector follows a standard which ensures that commonly used peripherals, such as a SPI, a UART, and an I2C bus, are available on fixed locations in the connector. The rest of the pins are used for general purpose IO. This allows the definition of expansion boards (EXP boards) that can plug into several different Silicon Labs Starter Kits.

The figure below shows the pin assignment of the EXP header. Because of limitations in the number of available GPIO pins, some of the EXP header pins are shared with kit features.

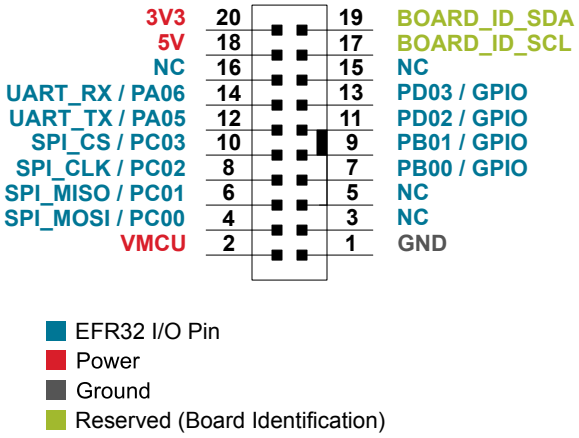


Figure 3.5. EXP Header

Downloaded from Arrow.com.

3.4.1 EXP Header Pinout

The pin-routing on the EFR32 is very flexible, so most peripherals can be routed to any pin. However, many pins are shared between the EXP header and other functions on the mainboard. The table below includes an overview of the mainboard features that share pins with the EXP header.

Table 3.1. EXP Header Pinout

Pin	Connection	EXP Header Function	Shared Feature	Peripheral Mapping
20	3V3	Board controller supply		
18	5V	Board USB voltage		
16	NC	I2C_SDA		
14	PA06	UART_RX	VCOM_RX	USART0_RX
12	PA05	UART_TX	VCOM_TX	USART0_TX
10	PC03	SPI_CS	DISP_SCS	USART0_CS
8	PC02	SPI_SCLK	DISP_SCLK	USART0_SCLK
6	PC01	SPI_MISO	VCOM_RTS	USART0_RTS
4	PC00	SPI_MOSI	DISP_MOSI	USART0_MOSI
2	VMCU	EFR32 voltage domain, included in AEM measurements.		
19	BOARD_ID_SDA	Connected to board controller for identification of add-on boards.		
17	BOARD_ID_SCL	Connected to board controller for identification of add-on boards.		
15	NC	I2C_SCL		
13	PD03	GPIO	UIF_LED1	
11	PD02	GPIO	UIF_LED0	
9	PB01	GPIO	UIF_BUTTON1	
7	PB00	GPIO	UIF_BUTTON0	
5	NC	GPIO		
3	NC	GPIO		
1	GND	Ground		

Downloaded from Arrow.com.

3.5 Logic Analyzer Connector

The Wireless Pro Kit Mainboard includes an on-board, eight-channel logic analyzer. It enables four digital signals to be sampled and displayed in Simplicity Studio, in addition to the state of the on-board user interface LEDs and buttons. The logic analyzer is a good tool for correlating specific events to the AEM energy profile and packet trace data as these are time-synchronized and can be visualized together. The sampling rate of 100 kHz limits its use in decoding digital protocols like I^2C or SPI.

The logic analyzer connector is situated on the top right side of the Wireless Pro Kit Mainboard. Four signals (channel 0-3) can be connected to the logic analyzer using this connector and the test probes that are obtainable through the "Si-DA001A Pro Kit Mainboard Accessory Kit". The test probes can be connected to the kit itself or on an external board connected to the Wireless Pro Kit Mainboard. Note that in both cases the connected signals must be digital, and the voltages referenced to ground and VMCU on the Wireless Pro Kit Mainboard. The table below gives an overview of the logic analyzer signals.

Note: The logic analyzer is only available on the Wireless Pro Kit Mainboard (BRD4002A). Using the external signals requires test probes which are obtainable through the "Si-DA001A Pro Kit Mainboard Accessory Kit".

Table 3.2. Logic Analyzer Signal Description

Type	Channel	Description
External signal	0	Connector (ch0)
	1	Connector (ch1)
	2	Connector (ch2)
	3	Connector (ch3)
Internal signal	4	LED0
	5	LED1
	6	BTN0
	7	BTN1

Downloaded from Arrow.com.

3.6 Debug Connector

The debug connector serves multiple purposes based on the "debug mode" setting which can be configured in Simplicity Studio. When the debug mode is set to "Debug IN", the debug connector can be used to connect an external debugger to the EFR32 on the radio board. When set to "Debug OUT", this connector allows the kit to be used as a debugger towards an external target. When set to "Debug MCU" (default), the connector is isolated from both the on-board debugger and the radio board target device.

Because this connector is electronically switched between the different operating modes, it can only be used when the board controller is powered (i.e., J-Link USB cable connected). If debug access to the target device is required when the board controller is unpowered, connect directly to the appropriate breakout pins.

The pinout of the connector follows that of the standard ARM Cortex Debug+ETM 19-pin connector. The pinout is described in detail below. Even though the connector has support for both JTAG and ETM Trace, it does not necessarily mean that the kit or the on-board target device supports these features.

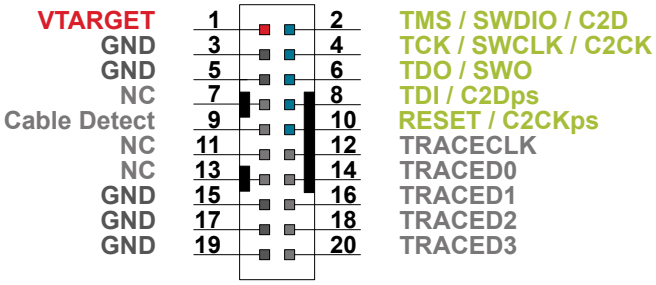


Figure 3.6. Debug Connector

Note: The pinout matches the pinout of an ARM Cortex Debug+ETM connector, but these are not fully compatible because pin 7 is physically removed from the Cortex Debug+ETM connector. Some cables have a small plug that prevents them from being used when this pin is present. If this is the case, remove the plug or use a standard 2x10 1.27 mm straight cable instead.

Table 3.3. Debug Connector Pin Descriptions

Pin Number(s)	Function	Description
1	VTARGET	Target reference voltage. Used for shifting logical signal levels between target and debugger.
2	TMS / SDWIO / C2D	JTAG test mode select, Serial Wire data, or C2 data
4	TCK / SWCLK / C2CK	JTAG test clock, Serial Wire clock, or C2 clock
6	TDO/SWO	JTAG test data out or Serial Wire Output
8	TDI / C2Dps	JTAG test data in or C2D "pin sharing" function
10	RESET / C2CKps	Target device reset or C2CK "pin sharing" function
12	TRACECLK	Not connected
14	TRACED0	PA03
16	TRACED1	Not connected
18	TRACED2	Not connected
20	TRACED3	Not connected
9	Cable detect	Connect to ground
7, 11, 13	NC	Not connected
3, 5, 15, 17, 19	GND	Ground

Downloaded from Arrow.com.

3.7 Simplicity Connector

The Simplicity Connector enables the advanced debugging features, such as the AEM, the virtual COM port, and the Packet Trace Interface, to be used towards an external target. The pinout is illustrated in the figure below.

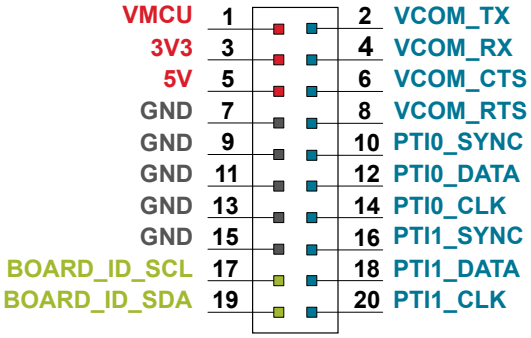


Figure 3.7. Simplicity Connector

Note: Current drawn from the VMCU voltage pin is included in the AEM measurements, while the 3V3 and 5V voltage pins are not. When monitoring the current consumption of an external target with the AEM, unplug the radio board from the mainboard to avoid adding the radio board's current consumption to the measurements.

Table 3.4. Simplicity Connector Pin Descriptions

Pin Number(s)	Function	Description
1	VMCU	3.3 V power rail, monitored by the AEM
3	3V3	3.3 V power rail
5	5V	5 V power rail
2	VCOM_TX	Virtual COM Tx
4	VCOM_RX	Virtual COM Rx
6	VCOM_CTS	Virtual COM CTS
8	VCOM_RTS	Virtual COM RTS
10	PTI0_SYNC	Packet Trace 0 Sync
12	PTI0_DATA	Packet Trace 0 Data
14	PTI0_CLK	Packet Trace 0 Clock
16	PTI1_SYNC	Packet Trace 1 Sync
18	PTI1_DATA	Packet Trace 1 Data
20	PTI1_CLK	Packet Trace 1 Clock
17	BOARD_ID_SCL	Board ID SCL
19	BOARD_ID_SDA	Board ID SDA
7, 9, 11, 13, 15	GND	Ground

Downloaded from Arrow.com.

3.8 Mini Simplicity Connector

The Mini Simplicity Connector on the Wireless Pro Kit Mainboard offers advanced debugging features on a 10-pin connector to be used towards an external target. The Mini Simplicity Connector offers the following features:

- Serial Wire Debug (SWD) with SWO
- Packet Trace Interface (PTI)
- Virtual COM port (VCOM)
- AEM monitored voltage rail

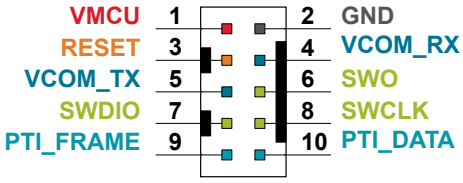


Figure 3.8. Mini Simplicity Connector

Note: Current drawn from the VMCU voltage pin is included in the AEM measurements. When monitoring the current consumption of an external target with the AEM, unplug the radio board from the Wireless Pro Kit Mainboard to avoid adding the radio board's current consumption to the measurements.

Table 3.5. Mini Simplicity Connector Pin Descriptions

Pin Number(s)	Function	Description
1	VMCU	Target voltage on the debugged application. Supplied and monitored by the AEM when power selection switch is in the "AEM" position.
2	GND	Ground
3	RST	Target device reset
4	VCOM_RX	Virtual COM Rx
5	VCOM_TX	Virtual COM Tx
6	SWO	Serial Wire Output
7	SWDIO	Serial Wire Data
8	SWCLK	Serial Wire Clock
9	PTI_FRAME	Packet Trace Frame Signal
10	PTI_DATA	Packet Trace Data Signal

Note: Mini Simplicity Connector pin-out is referenced from the device target side.

Downloaded from Arrow.com.

3.9 Debug Adapter

The BRD8010A STK/WSTK Debug Adapter is an adapter board which plugs directly into the debug connector and the Simplicity Connector on the mainboard. It combines selected functionality from the two connectors to a smaller footprint 10-pin connector, which is more suitable for space-constrained designs.

For versatility, the debug adapter features three different 10-pin debug connectors:

- Silicon Labs Mini Simplicity Connector
- ARM Cortex 10-pin Debug Connector
- Silicon Labs ISA3 Packet Trace

The ARM Cortex 10-pin Debug Connector follows the standard Cortex pinout defined by ARM and allows the Wireless STK to be used to debug hardware designs that use this connector.

The ISA3 connector follows the same pinout as the Packet Trace connector found on the Silicon Labs Ember Debug Adapter (ISA3). This enables using the Wireless STK to debug hardware designs that use this connector.

The Mini Simplicity Connector is designed to offer advanced debug features from the kit on a 10-pin connector. The connector has the same pinout and functionality as described in 3.8 Mini Simplicity Connector. It is only necessary to use the debug adapter to get access to the Mini Simplicity Connector when using the Wireless STK Mainboard (BRD4001A). If using the Wireless Pro Kit Mainboard (BRD4002A), use the Mini Simplicity Connector on the mainboard instead.

Downloaded from Arrow.com.

4. Power Supply and Reset

4.1 Radio Board Power Selection

The EFR32 on a Wireless STK can be powered by one of these sources:

- The debug USB cable
- A 3 V coin cell battery
- A USB regulator on the radio board (for devices with USB support only)

The power source for the radio board is selected with the slide switch in the lower left corner of the Wireless STK Mainboard or the Wireless Pro Kit Mainboard. The figure below shows how the different power sources can be selected with the slide switch.

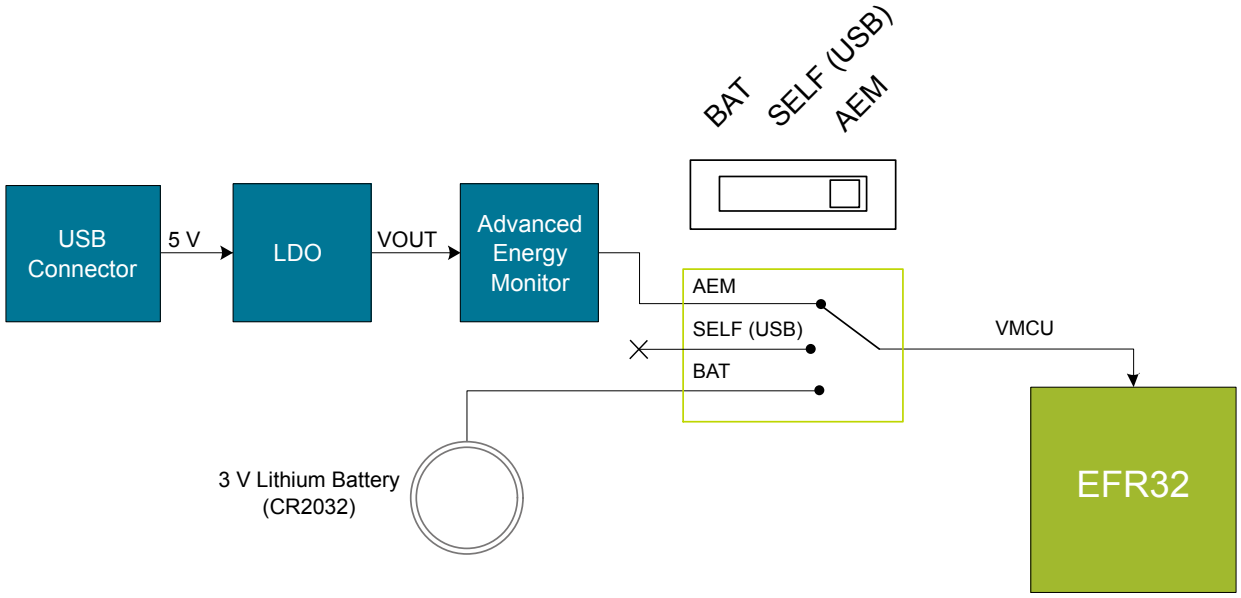


Figure 4.1. Power Switch

Note: The middle position is denoted by "USB" on the Wireless STK Mainboard, while it is denoted by "SELF" on the Wireless Pro Kit Mainboard. The slide switch functions the same on both mainboards.

Note: The AEM can only measure the current consumption of the EFR32 when the power selection switch is in the **AEM** position.

AEM position: With the switch in the AEM position, a low noise LDO on the mainboard is used to power the radio board. This LDO is again powered from the debug USB cable. The AEM is now also connected in series, allowing accurate high speed current measurements and energy debugging/profiling.

USB position: With the switch in the USB position, radio boards with USB-support can be powered by a regulator on the radio board itself. BRD4181B does not contain a USB regulator, and setting the switch in the **USB** position will cause the EFR32 to be unpowered.

BAT position: With the switch in the BAT position, a 20 mm coin cell battery in the CR2032 socket can be used to power the device. With the switch in this position, no current measurements are active. This is the switch position that should be used when the radio board is powered with an external power source. The Wireless Pro Kit Mainboard (BRD4002A) features an additional 2-pin JST connector connected in paralell to the CR2032 socket that can be used with an external power source between 1.8 V and 3.6 V instead of a coin cell. The coin cell battery is not protected from reverse current, and it is therefore important to remove the coin cell battery from the CR2032 socket if applying external power.

Note: The current sourcing capabilities of a coin cell battery might be too low to supply certain wireless applications.

Downloaded from Arrow.com.

4.2 Kit Power

There are normally two main contributions to the power consumption from the mainboard USB connector, i.e., two main current paths:

- One being monitored by the AEM that goes to the target power domain (VMCU)
- One that goes to the board controller power domain

While the current consumption of the board controller section is fairly deterministic and stable, the current consumption connected to the target's power domain (VMCU) varies widely depending on the application and the slide switch position. Typically, the board controller power domain draws 200 mA on the Wireless Starter Kit Mainboard (BRD4001A) and 250 mA on the Wireless Pro Kit Mainboard (BRD4002A). The mainboards use linear regulators, and the recommended input voltage is 4.4 - 5.25 V. Use a USB host or power supply and cables that can deliver at least the total amount of current required by the kit.

The 5V net exposed on the breakout pads, EXP header, and radio board is also sourced from the mainboard USB connector when the power select switch is in the AEM position. The 3V3 net exposed on the same peripherals is always sourced from the mainboard USB connector. The current consumption of these nets must be included in the total current consumption of the kit if these are utilized.

4.2.1 Board Controller Power

The board controller is responsible for important features, such as the debugger and the AEM, and is powered exclusively through the USB port in the top left corner of the board. This part of the kit resides on a separate power domain, so a different power source can be selected for the target device while retaining debugging functionality. This power domain is also isolated to prevent current leakage from the target power domain when power to the board controller is removed.

The board controller power domain is not influenced by the position of the power switch.

The kit has been carefully designed to keep the board controller and the target power domains isolated from each other as one of them powers down. This ensures that the target EFR32 device will continue to operate in the **BAT** mode.

4.2.2 AEM Power

The supply for the target power domain (VMCU) is a linear regulator integrated with the AEM described in Section 7. Advanced Energy Monitor when the power select switch is in the AEM position. The output voltage of the regulator is fixed to 3.3 V on the Wireless STK Mainboard (BRD4001A), while it can be adjusted between 1.8 V and 3.6 V on the Wireless Pro Kit Mainboard (BRD4002A) using the admin console.

The output current on the Wireless Pro Kit Mainboard (BRD4002A) is limited by an overcurrent protection (OCP) function, which depends on the programmed VMCU voltage: OCP (A) ≈ $VMCU_{SET}$ (V) x 0.2 (A/V). Approaching or exceeding the OCP limit is not recommended as the output voltage will be pulled low, which causes loss of function.

The maximum recommended output current on the Wireless STK Mainboard (BRD4001A) is 300 mA.

4.3 EFR32 Reset

The EFR32 Wireless SoC can be reset by a few different sources:

- A user pressing the RESET button
- The on-board debugger pulling the #RESET pin low
- An external debugger pulling the #RESET pin low

In addition to the reset sources mentioned above, a reset to the EFR32 will also be issued during board controller boot-up. This means that removing power to the board controller (unplugging the J-Link USB cable) will not generate a reset but plugging the cable back in will as the board controller boots up.

Downloaded from Arrow.com.

4.4 Battery Holder

In radio applications with high output power, peak current consumption will exceed the current sourcing capacity of a coin-cell battery. To support evaluation of the EFR32 Wireless Gecko in situations where powering the kit from a wired USB connection is impractical, for instance during range-tests, the kit is supplied with a battery holder for 2 AA batteries.

To use the battery holder, first set the power switch in the BAT position and remove any battery power sources (coin-cell or other sources) that may be connected to the BAT power net. Then attach the cable to pin 1 and 2 on the expansion header, orienting the connector so the black cable goes down towards pin 1, and the red cable up towards pin 2.

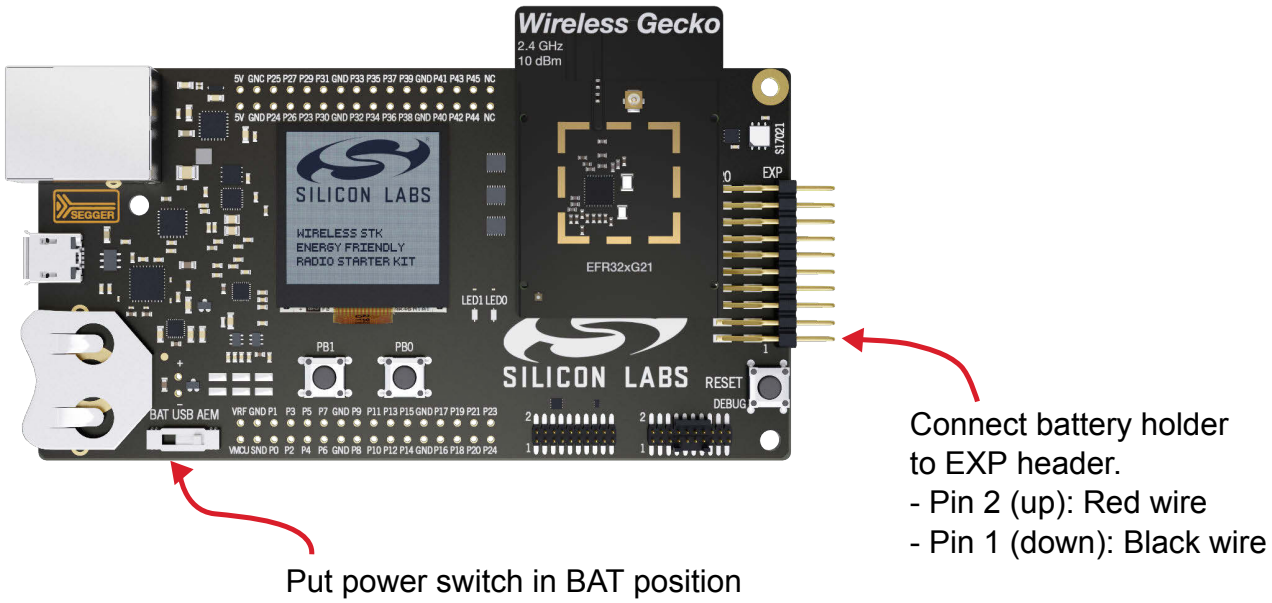


Figure 4.2. Battery Holder Connection

Warning: There is no reverse voltage protection on the VMCU pin! Ensure that the battery holder is connected the right way. Failure to do so may result in damage to the radio board and its components.

Downloaded from Arrow.com.

5. Peripherals

The Wireless STK has a set of peripherals that showcase some of the EFR32 features.

Note that most EFR32 I/Os routed to peripherals are also routed to the breakout pads or the EXP header, which must be taken into consideration when using these I/Os.

5.1 Push Buttons and LEDs

The kit has two user push buttons marked BTN0 and BTN1. They are connected directly to the EFR32 and are debounced by RC filters with a time constant of 1 ms. The buttons are connected to pins PB00 and PB01.

The kit features a total of four LEDs controlled by two GPIO pins on the EFR32. Two yellow LEDs, marked LED0 and LED1, are located on the mainboard. A red/green LED is located on the BRD4181B itself. The LEDs are connected such that PD02 controls LED0 on the mainboard and the red LED on BRD4181B. PD03 controls LED1 on the mainboard and the green LED on BRD4181B. The LEDs are all controlled in an active-high configuration.

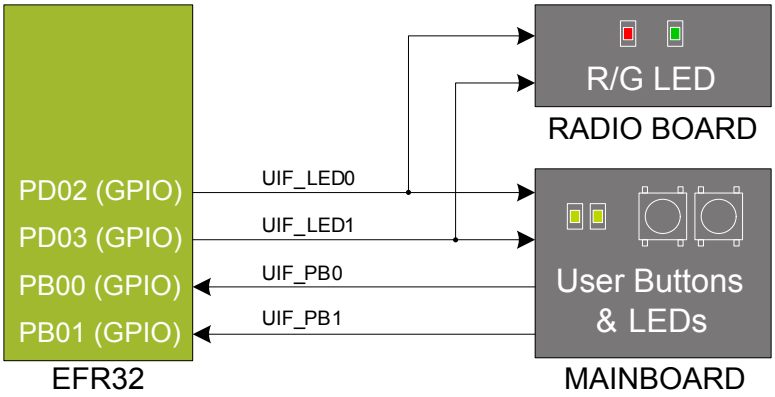


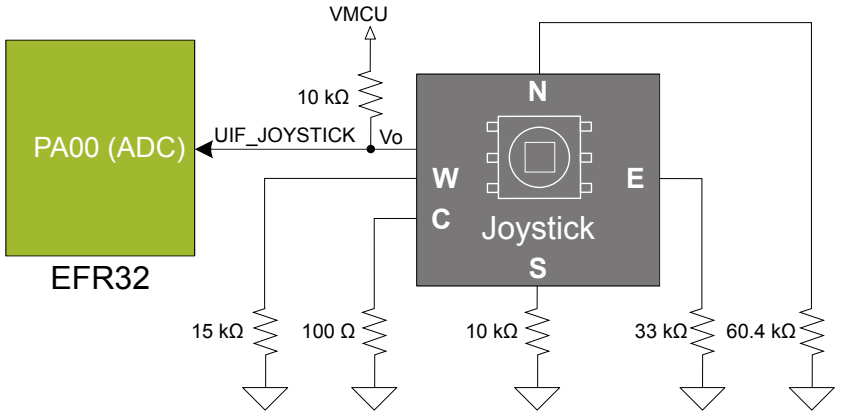
Figure 5.1. Buttons and LEDs

Downloaded from Arrow.com.

5.2 Joystick

The kit has an analog joystick connected to the EFR32 on pin PA00 when using a Wireless Pro Kit Mainboard (BRD4002A). The Wireless STK Mainboard (BRD4001A) does not feature a joystick. Moving the joystick around connects different pull-down resistors to the joystick output, which together with the pull-up resistor on VMCU, creates different output voltages, Vo, that can be read using the ADC on the EFR32.

Note: The PA00 pin on the EFR32 is also used for the display (DISP_EXTCOMIN) on this kit. Both the joystick and the display (DISP_EXTCOMIN) can be used, but not at the same time.



Direction	Divider Ratio	Vo (VMCU = 3.3 V)
Center press (C)	$\frac{100\ \Omega}{100\ \Omega + 10\ k\Omega}$	0.03 V
Up (N)	$\frac{60.4\ k\Omega}{60.4\ k\Omega + 10\ k\Omega}$	2.83 V
Left (W)	$\frac{15\ k\Omega}{15\ k\Omega + 10\ k\Omega}$	1.98 V
Down (S)	$\frac{10\ k\Omega}{10\ k\Omega + 10\ k\Omega}$	1.65 V
Right (E)	$\frac{33\ k\Omega}{33\ k\Omega + 10\ k\Omega}$	2.53 V

Figure 5.2. Joystick

Downloaded from Arrow.com.

5.3 Memory LCD-TFT Display

A 1.28-inch SHARP Memory LCD-TFT is available on the kit to enable interactive applications to be developed. The display has a high resolution of 128 x 128 pixels and consumes very little power. It is a reflective monochrome display, so each pixel can only be light or dark, and no backlight is needed in normal daylight conditions. Data sent to the display is stored in the pixels on the glass, which means no continuous refreshing is required to maintain a static image.

The display interface consists of a SPI-compatible serial interface and some extra control signals. Pixels are not individually addressable, instead data is sent to the display one line (128 bits) at a time.

The Memory LCD-TFT display is shared with the kit's board controller, allowing the board controller application to display useful information when the user application is not using the display. The user application always controls ownership of the display with the DISP_ENABLE signal:

- DISP_ENABLE = LOW: The board controller has control of the display
- DISP_ENABLE = HIGH: The user application (EFR32) has control of the display

Power to the display is sourced from the target application power domain when the EFR32 controls the display and from the board controller's power domain when the DISP_ENABLE line is low. Data is clocked in on DISP_SI when DISP_CS is high, and the clock is sent on DISP_SCLK. The maximum supported clock speed is 1.1 MHz.

DISP_EXTCOMIN is the "COM Inversion" line. It must be pulsed periodically to prevent static build-up in the display itself. Refer to the LS013B7DH03 documentation for more information on driving the display.

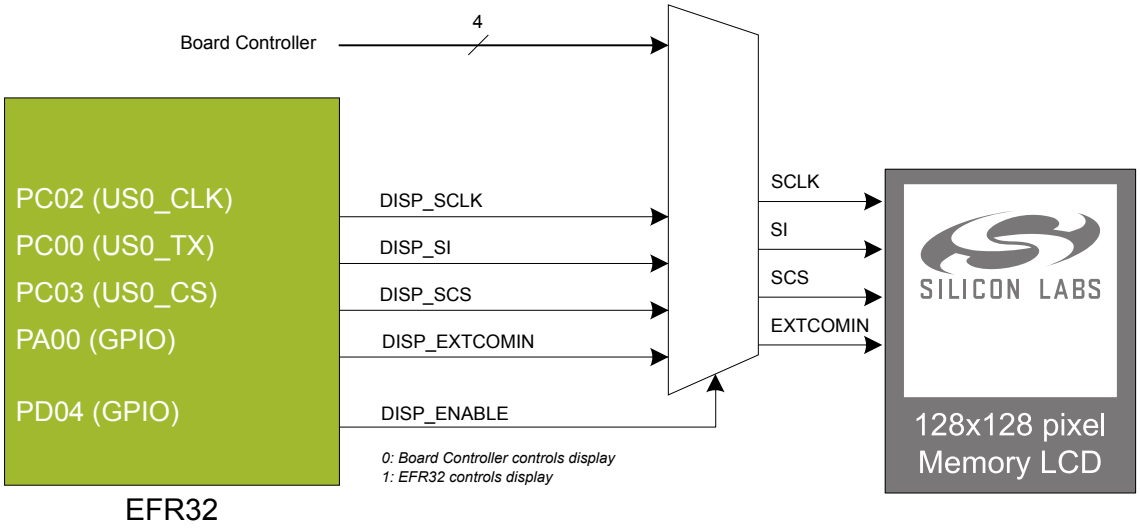


Figure 5.3. 128x128 Pixel Memory LCD

Downloaded from Arrow.com.

5.4 Virtual COM Port

An asynchronous serial connection to the board controller is provided for application data transfer between a host PC and the target EFR32. This eliminates the need for an external serial port adapter.

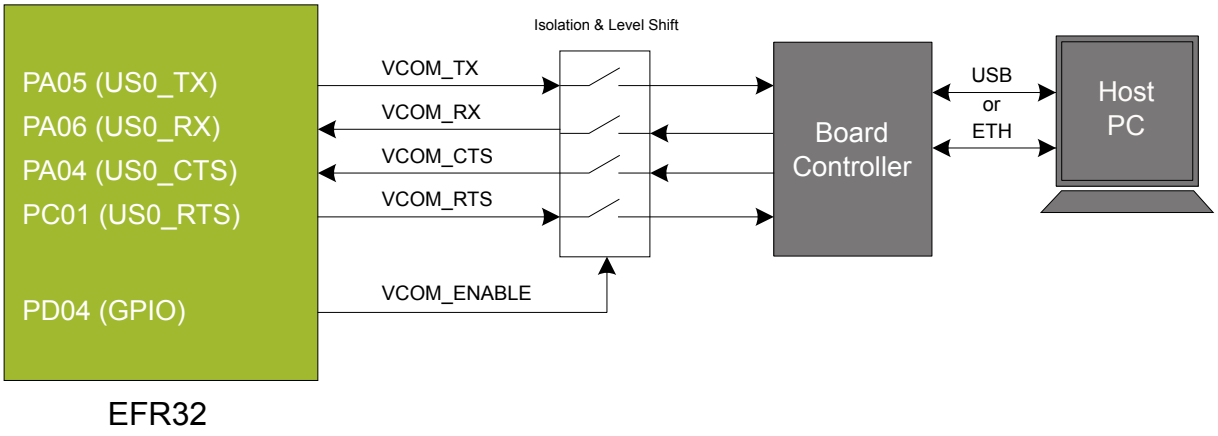


Figure 5.4. Virtual COM Port Interface

The virtual COM port consists of a physical UART between the target device and the board controller and a logical function in the board controller that makes the serial port available to the host PC over USB or Ethernet. The UART interface consists of four pins and an enable signal.

Table 5.1. Virtual COM Port Interface Pins

Signal	Description
VCOM_TX	Transmit data from the EFR32 to the board controller
VCOM_RX	Receive data from the board controller to the EFR32
VCOM_CTS	Clear to Send hardware flow control input, asserted by the board controller when it is ready to receive more data
VCOM_RTS	Request to Send hardware flow control output, asserted by the EFR32 when it is ready to receive more data
VCOM_ENABLE	Enables the VCOM interface, allowing data to pass through to the board controller

The parameters of the serial port, such as baud rate or flow control, can be configured using the admin console. The default settings depend on which radio board is used with the mainboard.

Note: The VCOM port is only available when the board controller is powered, which requires the J-Link USB cable to be inserted.

Note: There may be slight differences on the terminal prompt and settings between the Wireless Starter Kit Mainboard and the Wireless Pro Kit Mainboard.

Downloaded from Arrow.com.

5.4.1 Host Interfaces

Data can be exchanged between the board controller and the target device through the VCOM interface, which is then available to the user in two different ways:

- Virtual COM port using a standard USB-CDC driver
- TCP/IP by connecting to the Wireless STK on TCP/IP port 4901 with a Telnet client

When connecting via USB, the device should automatically show up as a COM port. The actual device name that is associated with the kit depends on the operating system and how many devices are or have been connected previously. The following are examples of what the device might show up as:

- JLink CDC UART Port (COM5) on Windows hosts
- /dev/cu.usbmodem1411 on macOS
- /dev/ttyACM0 on Linux

Data sent by the target device into the VCOM interface can be read from the COM port, and data written to the port is transmitted to the target device. Connecting to the Wireless STK on port 4901 gives access to the same data over TCP/IP. Data written into the VCOM interface by the target device can be read from the socket, and data written into the socket is transmitted to the target device.

Note: Only one of these interfaces can be used at the same time, with the TCP/IP socket taking priority. This means that if a socket is connected to port 4901, no data can be sent or received on the USB COM port.

5.4.2 Serial Configuration

By default, the VCOM serial port is configured to use 115200 8N1 (115.2 kbit/s, 8 data bits, 1 stop bit), with flow control disabled/ignored. The configuration can be changed using the admin console:

```
WPK> serial vcom config
Usage: serial vcom config [--nostore] [handshake <rts/cts/rtscts/disable/auto>] [speed <9600,921600>]
```

Using this command, the baud rate can be configured between 9600 and 921600 bit/s, and hardware handshake can be enabled or disabled on either or both flow control pins.

Downloaded from Arrow.com.

5.4.3 Hardware Handshake

The VCOM peripheral supports basic RTS/CTS flow control.

VCOM_CTS (target clear to send) is a signal that is output from the board controller and input to the target device. The board controller de-asserts this pin whenever its input buffer is full and it is unable to accept more data from the target device. If hardware handshake is enabled in the target firmware, its UART peripheral will halt when data is not being consumed by the host. This implements end-to-end flow control for data moving from the target device to the host.

VCOM_CTS is connected to the RTS pin on the board controller and is enabled by setting handshake to either RTS or RTSCTS using the "serial vcom config" command.

VCOM_RTS (target request to send) is a signal that is output from the target device and input to the board controller. The board controller will halt transmission of data towards the target if the target device de-asserts this signal. This gives the target firmware a means to hold off incoming data until it can be processed. Note that de-asserting RTS will not abort the byte currently being transmitted, so the target firmware must be able to accept at least one more character after RTS is de-asserted.

VCOM_RTS is connected to the CTS pin of the board controller. It is enabled by setting handshake to either CTS or RTSCTS using the "serial vcom config" command in the admin console. If CTS flow control is disabled, the state of VCOM_RTS will be ignored and data will be transmitted to the target device anyway.

Table 5.2. Hardware Handshake Configuration

Mode	Description
disabled	RTS (VCOM_CTS) is not driven by the board controller and CTS (VCOM_RTS) is ignored.
rts	RTS (VCOM_CTS) is driven by the board controller to halt target from transmitting when input buffer is full. CTS (VCOM_RTS) is ignored.
cts	RTS (VCOM_CTS) is not driven by the board controller. Data is transmitted to the target device if CTS (VCOM_RTS) is asserted and halted when de-asserted.
rtscts	RTS (VCOM_CTS) is driven by the board controller to halt target when buffers are full. Data is transmitted to the target device if CTS (VCOM_RTS) is asserted and halted when de-asserted.

Note: Enabling CTS flow control without configuring the VCOM_RTS pin can result in no data being transmitted from the host to the target device.

Downloaded from Arrow.com.

6. Board Controller

6.1 Introduction

The Wireless STK Mainboard and the Wireless Pro Kit Mainboard contain a dedicated microcontroller for some of the advanced kit features provided. This microcontroller is referred to as the board controller and is not programmable by the user. The board controller acts as an interface between the host PC and the target device on the radio board, as well as handling some housekeeping functions on the board.

Note: This chapter describes the board controller on both the Wireless Starter Kit Mainboard and the Wireless Pro Kit Mainboard. There might be slight differences between these two boards, such as the exact menu and format on the admin console, not highlighted in this chapter. The logic analyzer is furthermore only available on BRD4002A.

Some of the kit features actively managed by the board controller are:

- The on-board debugger, which can flash and debug both on-board and external targets.
- The Advanced Energy Monitor, which provides real-time energy profiling of the user application.
- The Packet Trace Interface, which is used in conjunction with PC software to provide detailed insight into an active radio network.
- The logic analyzer, which can capture digital signals time-synchronized to the energy profiling and packet trace data.
- The Virtual COM Port and Virtual UART interfaces, which provide ways to transfer application data between the host PC and the target processor.
- The admin console, which provides configuration of the various board features.

Silicon Labs publishes updates to the board controller firmware in the form of firmware upgrade packages. These updates may enable new features or fix issues. See Section 9.1 Firmware Upgrades for details on firmware upgrade.

6.2 Admin Console

The admin console is a command line interface to the board controller on the kit. It provides functionality for configuring the kit behavior and retrieving configuration and operational parameters.

6.2.1 Connecting

The admin console is available when the Wireless STK is connected to Ethernet using the Ethernet connector in the top left corner of the mainboard. See Section 8.1.2 Ethernet Interface for details on the Ethernet connectivity.

Connect to the admin console by opening a telnet connection to the kit's IP address, port number 4902.

When successfully connected, a `WPK>` prompt is displayed.

Downloaded from Arrow.com.

6.2.2 Built-in Help

The admin console has a built-in help system which is accessed by the `help` command. The `help` command will print a list of all top level commands:

```
WPK> help
*************** Root commands ****************
aem             AEM Configuration and Information Commands    [ avg, calibrate, calinfo ]
boardid         Commands for board ID probe.    [ list, probe ]
dbg             Debug interface status and control    [ info, mode ]
dch             Datachannel control and info commands    [ info, message ]
discovery       Discovery service commands.    [ key ]
net             Network commands.    [ dnslookup, ip, mac ]
pti             Packet trace interface status and control    [ config, disable, dump, ... ]
quit            Exit from shell
serial          Serial channel commands    [ vcom ]
sys             System commands    [ crashlog, nickname, reset, ... ]
target          Target commands.    [ button, go, halt, ... ]
time            Time Sync Service commands    [ client, disable, info, ... ]
user            User management functions    [ login,]
```

The `help` command can be used in conjunction with any top level command to get a list of sub-commands with descriptions. For example, `pti help` will print a list of all available sub-commands of `pti`:

```
WPK> pti help
*************** pti commands ****************
config          Configure packet trace
disable         Disable packet trace
dump            Dump PTI packets to the console as they come
enable          Enable packet trace
info            Packet trace state information
```

This means that running `pti enable` will enable packet trace.

6.2.3 Command Examples

PTI Configuration

```
pti config 0 efruart 1600000
```

Configures PTI to use the "EFRUART" mode at 1.6 Mb/s.

Serial Port Configuration

```
serial config vcom handshake enable
```

Enables hardware handshake on the VCOM UART connection.

6.3 Virtual UART

The Virtual UART (VUART) interface provides a high-performance application data interface that does not require additional I/O pins apart from the debug interface.

The Wireless STK makes the VUART interface available on TCP/IP port 4900.

6.3.1 Target-to-Host

Target-to-host communication utilizes the SWO-pin of the debug interface through the ITM debug peripheral. This approach allows a sleepy target device to enter all energy modes and still wake up intermittently to send debug information. The baud rate of the SWO data is locked to 875 kHz.

VUART utilizes ITM stimulus port 0 for general purpose printing. Silicon Labs' networking stacks utilize ITM stimulus port 8 for debug printing. The data on port 8 is encapsulated in additional framing and will also appear in the Simplicity Studio Network Analyzer.

Downloaded from Arrow.com.

6.3.2 Host-to-Target

Host-to-target communication utilizes SEGGER's Real Time Transfer (RTT) technology. A full explanation of how this works can be found in *J-Link/J-Trace User Guide (UM08001)*. Briefly summarized, RTT consists of a structure called the RTT Control Block, which is located in RAM. This control block points to circular buffers that the debugger can write data into. The target application can then read data out of this circular buffer.

The board controller will start searching for the RTT Control Block upon receiving data on TCP/IP port 4900. If the board controller is unable to locate the RTT Control Block, it will return an error message on the same connection. For the board controller to be able to locate the RTT Control Block, it has to be aligned on a 1024-byte boundary in RAM.

After initializing the RTT connection, the target will only enter emulated EM2 and EM3 where the power consumption remains similar to EM1. This is because RTT utilizes the debug interface, which requires use of high-frequency oscillators. Energy modes EM4S and EM4H will work as normal. When debugging energy consumption, it is therefore important to not send data on TCP/IP port 4900 as not to instantiate the RTT connection.

6.3.3 Limitations

- Because the SWO-connection can be disabled by the debugger at will, it is important for the target application to verify that SWO is enabled and configured before each transmission on the interface.
- After initializing host-to-target communication over RTT by sending data on TCP/IP port 4900, the target application will be unable to enter EM2 and EM3. This is because RTT utilizes the debug connection of the target.
- VUART might not work reliably during an active debugging session. This is because there is contention over the target's debug interface. The board controller will defer accessing the target until it is made available by the host debugger.
- VUART is designed with the assumption that only the board controller will access the RTT control block. If the target application uses RTT for other purposes, such as Segger SystemView, refrain from using VUART.

6.3.4 Troubleshooting

Problem	Solution
No data received after ending a debug session.	After certain debugger operations, the host computer manually disables SWO on the target to conserve power. This might cause SWO data to not appear if the target application initialized SWO before the debugger has disconnected. Either press the RESET button on the Wireless Starter Kit to reset the target application or make sure that the target application verifies that SWO is enabled and configured before sending any data.
No data received after flashing a new application.	
Other issues	Disconnect from TCP port 4900, press the RESET button on the kit, then reconnect to 4900. If this does not fix the issue, try to restart the kit by unplugging and replugging the USB cable.

Downloaded from Arrow.com.

7. Advanced Energy Monitor

7.1 Introduction

Any embedded developer seeking to make their embedded code spend as little energy as the underlying architecture supports needs tools to easily and quickly discover inefficiencies in the running application. This is what the Simplicity Energy Profiler is designed to do. In real-time, the Energy Profiler will graph and log current as a function of time while correlating this to the actual target application code running on the EFR32. There are multiple features in the profiler software that allow for easy analysis, such as markers and statistics on selected regions of the current graph or aggregate energy usage by different parts of the application. The Energy Profiler is available through Simplicity Studio.

7.2 Code Correlation

By using the Energy Profiler, current consumption and voltage can be measured and linked to the actual code running on the EFR32 in realtime. The Energy Profiler gets its data from the board controller on the mainboard through the Advanced Energy Monitor (AEM). The current signal is combined with the target processor's Program Counter (PC) sampling by utilizing a feature of the ARM CoreSight debug architecture, and the Instrumentation Trace Macrocell (ITM) block can be programmed to sample the MCU's PC at periodic intervals and output these over SWO pin ARM devices. When these two data streams are fused and correlated with the running application's memory map, an accurate statistical profile can be built that shows the energy profile of the running application in real-time.

7.3 AEM Circuit

The AEM circuit on the Wireless Pro Kit Mainboard (BRD4002A) and the Wireless STK Mainboard (BRD4001A) measures the current through a sense resistor inside the feedback loop of a low-dropout regulator (LDO). The output voltage of this LDO powers the EFR32 when the power slide switch is in the AEM position. AEM usage on both mainboards is similar, but the implementation and perfomance on the Wireless Pro Kit Mainboard (BRD4002A) has some key differences, including the utilization of two sense resistors instead of one, and a different LDO, which is explained in Section 7.3.1 AEM Details. The AEM implementation on the Wireless Pro Kit Mainboard (BRD4002A) is shown in the figure below.

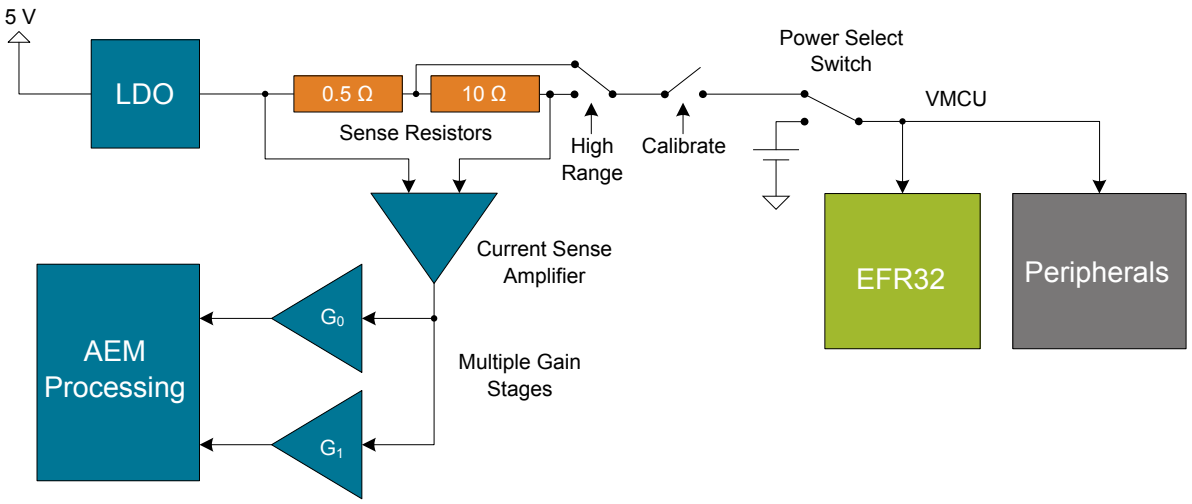


Figure 7.1. Advanced Energy Monitor On The Wireless Pro Kit Mainboard (BRD4002A)

Note: The VMCU regulator feedback point is after the sense resistor to ensure that the VMCU voltage is kept constant when the output current changes. Series resistances in the current path will, however, cause some IR drop on VMCU.

Note: The AEM circuit only works when the kit is powered and the power switch is in the AEM position.

Downloaded from Arrow.com.

7.3.1 AEM Details

The main differences between the AEM on the Wireless Pro Kit Mainboard (BRD4002A) and the Wireless STK Mainboard (BRD4001A) is summarized in the table below with more in-depth information given in the text to follow.

Table 7.1. Advanced Energy Monitor Parameters

Parameter	BRD4002A	BRD4001A
Voltage	1.8 - 3.6 V	3.3 V
Sample Rate	100 kHz	10 kHz
Sense Resistor	10.5 Ω / 0.5 Ω	2.35 Ω
Measurement Range[1]	0 - 495 mA	0 - 95 mA

Note:
1. The current sourcing capabilities of the LDO may be different than the measurement range.

Wireless Pro Kit Mainboard (BRD4002A) AEM Design Details

The AEM circuitry on the Wireless Pro Kit Mainboard is capable of measuring current signals in the range of approximately 0.1 µA to 495 mA. This is accomplished through a combination of a highly capable current sense amplifier, multiple sense resistors and gain stages, and signal processing within the kit's board controller before the current sense signal is read by a host computer with 100 kHz sample rate for display and/or storage. Averaging on the output data may be required to achieve sufficient accuracy in some situations, such as low currents, which can be traded for lower bandwidth. High current applications require that the regulator is able to supply enough current as described in Section 4.2 Kit Power.

At low currents the current sense amplifier measures the voltage drop over a 10.5 Ω resistive path. The gain stage further amplifies this voltage with two different parallel gain settings to obtain two current ranges. The transition between these two ranges occurs around 150 µA. When the current exceeds a threshold, which is typically between 10 and 30 mA, the AEM circuitry switches from the 10.5 Ω resistive path to a 0.5 Ω sense resistor and is now capable of measuring currents up to approximately 495 mA. Should the current drop below the threshold again, the sense resistor is changed back to the 10.5 Ω resistive path and the AEM is back to using two different gain stages depending on whether the current is above or below 150 µA.

The expected typical accuracy of the AEM on the Wireless Pro Kit Mainboard is within 1 %, except for currents in the low tens of micro-amps where offset errors start to dominate. In this low current region, the expected typical accuracy is some hundred nanoamps. At kit power-up or on a power-cycle, an automatic AEM calibration is performed which compensates for offset errors in the current sense amplifiers. To achieve the stated accuracy, averaging of the AEM output data is required in certain situations (typically at low currents and close to the bottom of the measurement ranges) to reduce noise. Averaging can be applied in Energy Profiler to suit different requirements during or after the acquisition. The analog bandwidth of the measurement circuit depends on multiple factors, such as output current and capacitance on the VMCU net, and may be lower than the output data rate. Generally, higher output current and lower capacitance on VMCU gives a higher analog bandwidth.

Wireless STK Mainboard (BRD4001A) AEM Design Details

The AEM circuitry on the Wireless STK Mainboard works conceptually in a similar way to the implementation on the Wireless Pro Kit Mainboard except for two key differences: it uses only one 2.35 Ω sense resistor and the low-dropout regulator (LDO) is different. For details about the two implementations, the reader is encouraged to see the schematics.

The AEM on the Wireless STK Mainboard is capable of measuring currents in the range of 0.1 µA to 95 mA. The second stage amplifier amplifies the signal with two different gain settings with the transition occurring around 250 µA. For currents above 250 µA, the AEM is accurate within 0.1 mA. When measuring currents below 250 µA, the accuracy increases to 1 µA. Even though the absolute accuracy is 1 µA in the sub 250 µA range, the AEM can detect changes in the current consumption as small as 0.1 µA. It is possible to source currents above the measurement range as decribed in Section 4.2 Kit Power. The board controller outputs the AEM data with 10 kHz sample rate.

Downloaded from Arrow.com.

8. On-Board Debugger

The Wireless Pro Kit Mainboard and the Wireless STK Mainboard contain an integrated debugger, which can be used to download code and debug the EFR32. In addition to programming a target on a plug-in radio board, the debugger can also be used to program and debug external Silicon Labs EFM32, EFM8, EZR32, and EFR32 devices connected through the debug connector.

The debugger supports three different debug interfaces for Silicon Labs devices:

- Serial Wire Debug is supported by all EFM32, EFR32, and EZR32 devices
- JTAG is supported by EFR32 and some EFM32 devices
- C2 Debug is supported by EFM8 devices

For debugging to work properly, make sure the selected debug interface is supported by the target device. The debug connector on the board supports all three of these modes.

8.1 Host Interfaces

The Wireless STK supports connecting to the on-board debugger using either Ethernet or USB.

Many tools support connecting to a debugger using either USB or Ethernet. When connected over USB, the kit is identified by its J-Link serial number. When connected over Ethernet, the kit is normally identified by its IP address. Some tools also support using the serial number when connecting over Ethernet; however, this typically requires the computer and the kit to be on the same subnet for the discovery protocol (using UDP broadcast packets) to work.

8.1.1 USB Interface

The USB interface is available whenever the USB connector on the left-hand side of the mainboard is connected to a computer.

8.1.2 Ethernet Interface

The Ethernet interface is available when the mainboard Ethernet connector in the top left corner is connected to a network. Normally, the kit will receive an IP address from a local DHCP server, and the IP address is printed on the LCD display. If your network does not have a DHCP server, you need to connect to the kit via USB and set the IP address manually using Simplicity Studio, Simplicity Commander, or J-Link Configurator.

For the Ethernet connectivity to work, the kit must still be powered through the mainboard USB connector.

8.1.3 Serial Number Identification

All Silicon Labs kits have a unique J-Link serial number which identifies the kit to PC applications. This number is 9 digits and is normally on the form `44xxxxxxx`.

The J-Link serial number is normally printed at the bottom of the kit LCD display.

Downloaded from Arrow.com.

8.2 Debug Modes

The kit can be used in various debug modes as explained in this chapter. The on-board debugger can be used to debug the EFR32 on the radio board, or it can be used to debug a supported external target board using either the debug connector or the Mini Simplicity Connector. An external debugger can furthermore be used to debug the EFR32 on the radio board using the debug connector. Selecting the active debug mode is done in Simplicity Studio.

Note: The Wireless Starter Kit Mainboard (BRD4001A) does not feature a Mini Simplicity Connector; therefore, debugging an external target board directly over the Mini Simplicity Connector is not supported on this mainboard. However, it is possible to debug an external target that uses a Mini Simplicity Connector from the Wireless Starter Kit Mainboard by using a BRD8010A STK/WSTK Debug Adapter.

Debug MCU: In this mode, the on-board debugger is connected to the EFR32 on the kit. To use this mode, set the debug mode to [**MCU**].

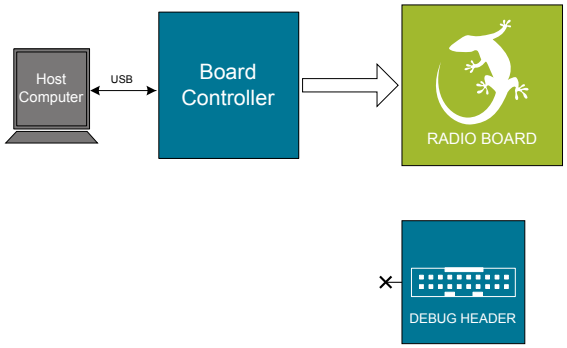


Figure 8.1. Debug MCU

Debug OUT: In this mode, the on-board debugger can be used to debug a supported Silicon Labs device mounted on a custom board using the debug connector. To use this mode, set the debug mode to [**Out**].

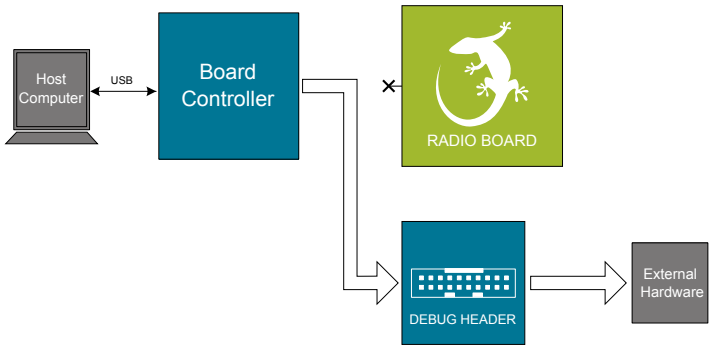


Figure 8.2. Debug OUT

Debug IN: In this mode, the on-board debugger is disconnected and an external debugger can be used to debug the EFR32 on the kit over the debug connector. To use this mode, set the debug mode to [**In**].

Downloaded from Arrow.com.

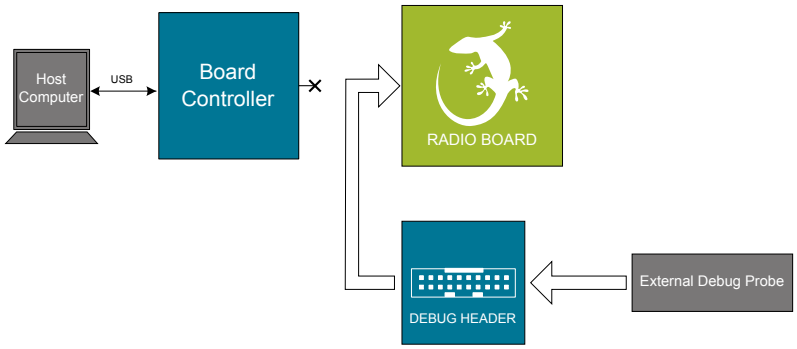


Figure 8.3. Debug IN

Note: For "Debug IN" to work, the kit board controller must be powered through the Debug USB connector.

Debug MINI: The Wireless Pro Kit mainboard features a dedicated Mini Simplicity Connector on the board. In this mode, the on-board debugger can be used to debug a supported Silicon Labs device mounted on a custom board over Serial Wire Debug. Virtual COM port and Packet Trace Interface is also available in this mode. To use this mode, set the debug mode to [**Mini**].

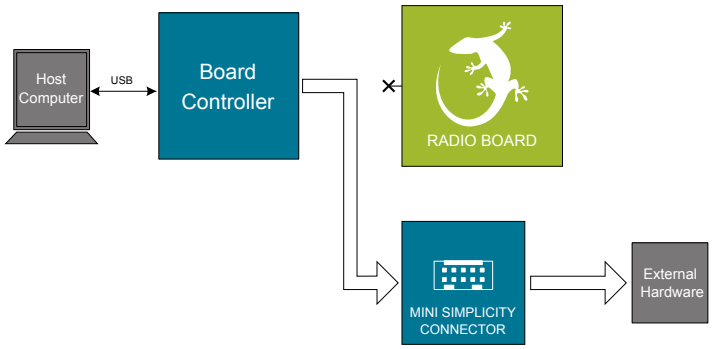


Figure 8.4. Mini Out

8.3 Debugging During Battery Operation

When the EFR32 is battery-powered and the J-Link USB is still connected, the on-board debug functionality is available. If the USB power is disconnected, the Debug IN mode will stop working.

If debug access is required when the target is running off another energy source, such as a battery, and the board controller is powered down, make direct connections to the GPIOs used for debugging, which are exposed on the breakout pads.

Downloaded from Arrow.com.

9. Kit Configuration and Upgrades

The kit configuration dialog in Simplicity Studio allows you to change the J-Link adapter debug mode, upgrade its firmware, and change other configuration settings. To download Simplicity Studio, go to silabs.com/simplicity.

In the main window of the Simplicity Studio's Launcher perspective, the debug mode and firmware version of the selected J-Link adapter are shown. Click the [**Change**] link next to any of these settings to open the kit configuration dialog.

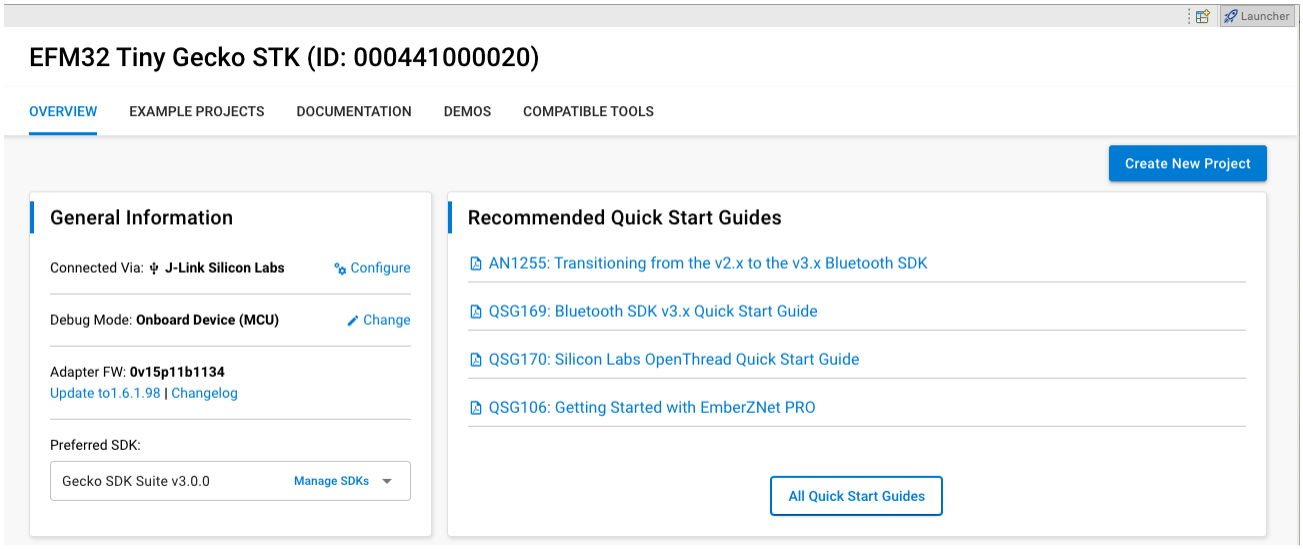


Figure 9.1. Simplicity Studio Kit Information

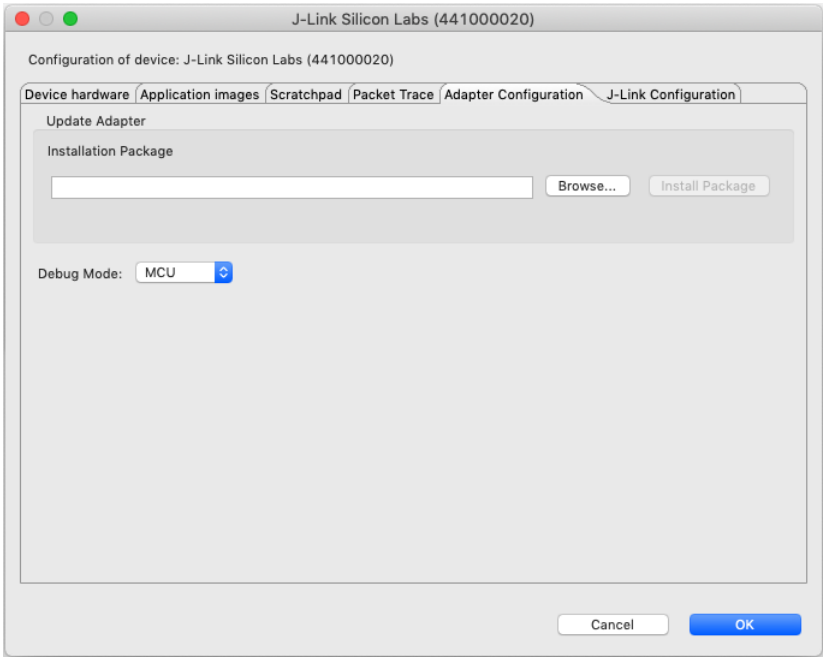


Figure 9.2. Kit Configuration Dialog

9.1 Firmware Upgrades

You can upgrade the kit firmware through Simplicity Studio. Simplicity Studio will automatically check for new updates on startup.

You can also use the kit configuration dialog for manual upgrades. Click the [**Browse**] button in the [**Update Adapter**] section to select the correct file ending in `.emz`. Then, click the [**Install Package**] button.

Downloaded from Arrow.com.

10. Schematics, Assembly Drawings, and BOM

Schematics, assembly drawings, and bill of materials (BOM) are available through Simplicity Studio when the kit documentation package has been installed. They are also available from the kit page on the Silicon Labs website: silabs.com.

Downloaded from Arrow.com.

11. Kit Revision History

The kit revision can be found printed on the kit packaging label, as outlined in the figure below. The revision history given in this section may not list every kit revision. Revisions with minor changes may be omitted.

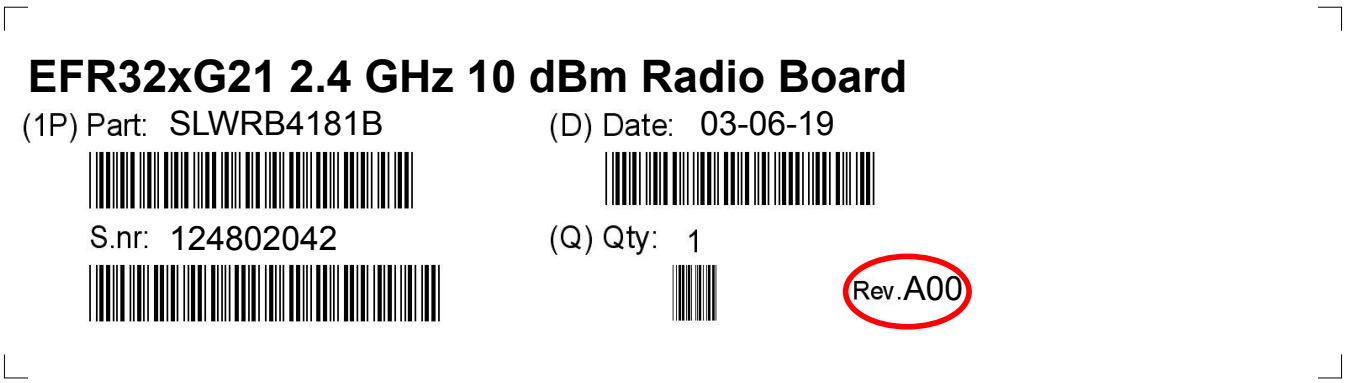


Figure 11.1. Kit Label

11.1 SLWRB4181B Revision History

Kit Revision	Released	Description
A04	9 December 2021	Changed box. Removed getting started card.
A03	6 July 2021	Updated BRD4181B to Rev. A03.
A02	13 April 2021	Updated BRD4181B to Rev. A02.
A01	6 July 2020	Updated BRD4181B to Rev. A01.
A00	3 June 2019	Initial release.

11.2 SLWSTK6023A Revision History

Kit Revision	Released	Description
B00	28 March 2022	Changed BRD4001A Rev. A02 with BRD4002A Rev. A06. Removed USB cable and added 10-pin 1.27 mm IDC debug cable.
A03	6 July 2021	Updated BRD4181B to Rev. A03.
A02	14 April 2021	Updated BRD4181B to Rev. A02.
A01	8 July 2020	Replaced BRD4181A with BRD4181B Rev. A01.
A00	11 October 2019	Initial release.

Downloaded from Arrow.com.

11.3 SLWSTK6006A Revision History

Kit Revision	Released	Description
C00	24 March 2022	Replaced BRD4001A Rev. A02 with BRD4002A Rev. A06. Removed BRD8010A. Removed USB cables.
B06	6 July 2021	Updated BRD4180B to Rev. A03 and BRD4181B to Rev. A03.
B05	14 April 2021	Updated BRD4180B to Rev. A02 and BRD4181B to Rev. A02.
B04	17 March 2021	Updated BRD8010A to Rev. A03.
B03	5 August 2020	Replaced BRD4180A with BRD4180B Rev. A01. Replaced BRD4181A with BRD4181B Rev. A01.
B02	8 July 2020	Replaced BRD4180B with BRD4180A Rev. A02. Replaced BRD4181B with BRD4181A Rev. A02. Updated BRD4001A to Rev. A02.
B01	6 November 2019	Added Thread Certified sticker to box.
B00	3 June 2019	Replaced BRD4180A with BRD4180B Rev. A00. Replaced BRD4181A with BRD4181B Rev. A00.
A00	18 January 2019	Initial release with BRD4180A Rev. A01 and BRD4181A Rev. A01.

Downloaded from Arrow.com.

12. Document Revision History

Revision 2.0

September 2022

- Major update to document content.
  - The user guide now describes how the kit works with the Wireless Starter Kit Mainboard and the Wireless Pro Kit Mainboard.
- Updated SLWSTK6006A kit revision history and kit content.
- Updated SLWSTK6023A kit revision history and kit content.
- Updated SLWRB4180B kit revision history.
- Changed document title from "UG429: EFR32xG21 2.4 GHz 10 dBm Wireless Starter Kit User's Guide" to "UG429: EFR32xG21 2.4 GHz 10 dBm Radio Board User's Guide".

Revision 1.0

April 2020

- Initial document version.

Downloaded from Arrow.com.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

Disclaimer
Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the information supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon Labs product in such unauthorized applications.
Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit www.silabs.com/about-us/inclusive-lexicon-project

Trademark Information
Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals®, WiSeConnect , n-Link, ThreadArch®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, Gecko OS, Gecko OS Studio, Precision32®, Simplicity Studio®, Telegesis, the Telegesis Logo®, USBXpress® , Zentri, the Zentri logo and Zentri DMS, Z-Wave®, and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.

SILICON LABS

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

www.silabs.com

Downloaded from Arrow.com.