

UG474: PG22 Dev Kit User's Guide

The PG22 Dev Kit is a low cost, small form factor prototype and development platform for the EFM32PG22 microcontroller.

The board is a small and cost effective, feature rich, prototype and development platform based on the EFM32™ microcontroller. The PG22 Dev Kit is an ideal platform for developing energy-friendly electronic devices.

A built in SEGGER J-Link debugger ensures easy debugging through the USB Micro-B connector.

Most of the EFM32PG22 I/Os are routed to breakout pads on the sides of the board making it easy to expand on the board and evaluate the EFM32PG22's features with custom hardware



TARGET DEVICE

- EFM32PG22 microcontroller (EFM32PG22C200F512IM40-C)
 - 32-bit ARM® Cortex®-M33 with 76.8 MHz maximum operating frequency
 - 512 kB flash and 32 kB RAM
 - Energy-efficient MCU core with low active and sleep currents
 - Secure Boot with Root of Trust and Secure Loader (RTSL)

KIT FEATURES

- Power control of on-board peripherals for ultra low power operation
- Relative humidity and temperature sensor
- Ambient light sensor
- Hall effect sensor
- 6-axis inertial sensor
- PDM stereo microphones
- 8 Mbit serial flash
- User LED and push button
- 20-pin 2.54 mm breakout pads
- SEGGER J-Link on-board debugger
- Virtual COM port
- Mini Simplicity connector for AEM and virtual COM using external Silicon Labs debugger
- USB or coin cell battery powered.

SOFTWARE SUPPORT

- Simplicity Studio™

ORDERING INFORMATION

- PG22-DK2503A

Table of Contents

1. Introduction	3
1.1 Kit Contents	3
1.2 Hardware Content	3
1.3 Kit Hardware Layout	4
2. Specifications	5
2.1 Recommended Operating Conditions	5
2.2 Current Consumption	6
3. Hardware	8
3.1 Block Diagram	8
3.2 Power Supply	9
3.3 EFM32PG22 Reset	9
3.4 Peripherals	10
3.4.1 Si7021 Relative Humidity and Temperature Sensor	11
3.4.2 Si7210 Hall Effect Sensor	11
3.4.3 VEML6035 Ambient Light Sensor	12
3.4.4 PDM Stereo Microphones	12
3.4.5 ICM-20648 6-Axis Inertial Sensor	13
3.4.6 Push Button and LED	14
3.4.7 External Memory	14
3.5 On-board Debugger	15
3.6 Connectors	16
3.6.1 Breakout Pads	17
3.6.2 Mini Simplicity Connector	18
3.6.3 Debug USB Micro-B Connector	18
4. Debugging	19
4.1 On-board Debugger	20
4.2 External Debugger	20
4.3 Virtual COM Port	20
5. Schematics, Assembly Drawings, and BOM	21
6. Kit Revision History and Errata	22
6.1 Revision History	22
6.2 Errata	22
7. Board Revision History and Errata	23
7.1 Revision History	23
7.2 Errata	23
8. Document Revision History	24

1. Introduction

The PG22 Dev Kit has been designed to allow customers getting familiar with the Silicon Labs EFM32PG22 microcontrollers. The highlights of the board include four different environmental sensors and stereo PDM microphones accessible to the EFM32PG22 MCU. An 8 Mbit serial flash, also embedded on this board, can be used for general purpose non-volatile memory.

The peripherals have been grouped into power domains that can be turned on and off by the application code as needed. A coin cell battery holder makes this kit like a small self contained system, easing development of battery operated products based on EFM32PG22.

Programming the PG22 Dev Kit is easily done using a USB Micro-B cable and the on-board J-Link debugger. A USB virtual COM port provides a serial connection to the target application. The PG22 Dev Kit is supported in Simplicity Studio™, and a Board Support Package (BSP) is provided to give application developers a flying start.

Energy profiling and advanced debugging tools are available through the provided Mini Simplicity Connector using an external Silicon Labs debugger.

Connecting external hardware to the PG22 Dev Kit can be done using the 20 breakout pads, which present peripherals from the EFM32PG22 such as I²C, SPI, UART and GPIOs. The breakout pads follow the same pinout as the EXP headers on other Silicon Labs Starter Kits.

1.1 Kit Contents

The following items are included in the box:

- 1x PG22 Dev Kit board (BRD2503A)
- 1x USB Type A to Micro-B cable

1.2 Hardware Content

The following key hardware elements are included on the PG22 Dev Kit:

- EFM32PG22 MCU with 76.8 MHz operating frequency, 512 kB flash and 32 kB RAM
- Silicon Labs Si7021 relative humidity and temperature sensor
- Silicon Labs Si7210 hall effect sensor
- Ambient light sensor (Vishay VEML6035)
- 6-axis inertial sensor (TDK InvenSense ICM-20648)
- Two MEMS microphones (Knowles SPK0641HT4H-1)
- Ultra low power 8 Mbit SPI flash (Macronix MX25R8035F)
- One LED and one push button
- On-board SEGGER J-Link debugger for easy programming and debugging, which also includes a USB virtual COM port
- Mini Simplicity connector for access to energy profiling and other advanced debugging features
- Breakout pads for GPIO access and connection to external hardware
- Reset button
- Automatic switchover between USB and battery power
- CR2032 coin cell holder

1.3 Kit Hardware Layout

The layout of the PG22 Dev Kit is shown below.

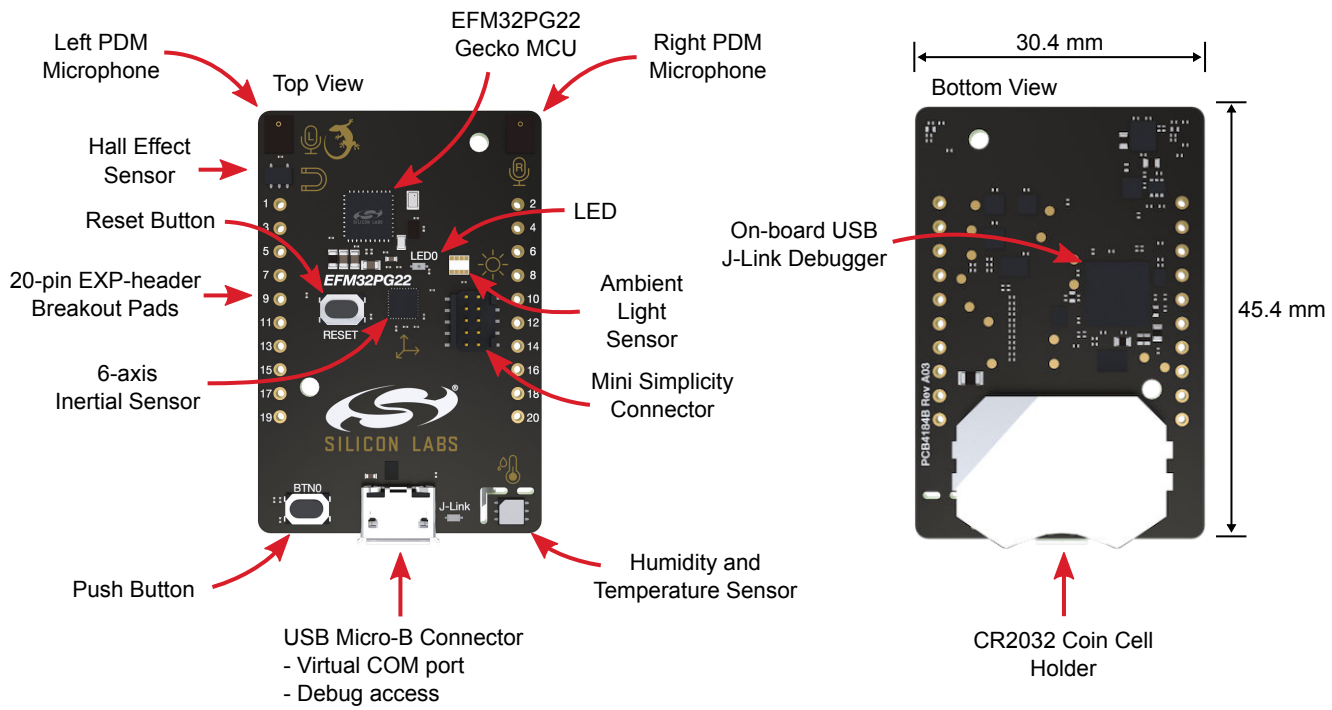


Figure 1.1. PG22 Dev Kit Hardware Layout

2. Specifications

2.1 Recommended Operating Conditions

The following table is intended to serve as guideline for a correct use of PG22 Dev Kit, indicating typical operating conditions and some design limits.

Table 2.1. Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit
USB Supply Input Voltage	V_{USB}	-	5.0	-	V
Battery Supply Input Voltage ^{1,2}	V_{VBAT}	2.0	3.0	3.3	V
Supply Input Voltage (VMCU supplied externally) ^{1,2,3}	V_{VMCU}	2.0	3.0	3.6	V
Operating Temperature	T_{OP}	-	20	-	°C

Note:

1. Brightness of LED0 will vary with the supply voltage. Due to manufacturing tolerances, functionality is not guaranteed over the entire working range
2. To optimize efficiency, EFM32PG22's internal DC-DC converter should be set in BYPASS mode when the input voltage approaches the output voltage, typically for supply voltages lower than 2.2 V. More details are found in EFM32PG22's datasheet
3. Not recommended for use with rechargeable Lithium batteries. Most Li-Ion and Li-Po cells exceed 3.6 V when fully charged

2.2 Current Consumption

The operating current of the board greatly depends on the application. The table below summarizes the datasheet current consumption of the sensors and peripherals on the board. In many cases the given conditions differ from the operating conditions on the PG22 Dev Kit, but the table can still be used as an indication of how much each feature contribute to the total current consumption. More details can be found in the specific datasheet for each device.

Table 2.2. Current Consumption

Parameter	Symbol	Condition ¹	Typ	Unit
EFM32PG22 Current Consumption ²	I _{EM4}	EM4, no BURTC, no LF oscillator	0.17	μA
	I _{EM2}	EM2 mode with Full RAM retention and RTC running from LFXO (VREGVDD = 3.0 V, AVDD = DVDD = IOVDD = RFVDD = PAVDD = 1.8 V from DCDC)	1.40	μA
	I _{EM0}	EM0 mode with all peripherals disabled (VREGVDD = 3.0 V, AVDD = DVDD = IOVDD = RFVDD = PAVDD = 1.8 V from DCDC, VSCALE1, 38.4 MHz crystal, CPU running Prime from flash) ³	1.1	mA
Serial Flash Memory Current Consumption ⁴	ISB2	Deep power-down at 1.8 V, chip select HIGH	0.007	μA
	ISB1	Standby at 1.8 V, chip select HIGH	5	μA
	ICC3	Write Status Register at 1.8 V, chip select HIGH	3.1	mA
Hall Effect Magnetic Sensor Current Consumption ⁵	I _{DD}	Sleep mode at 3.3 V	50	nA
		Average current for periodic activation at 3.3 V, sleep timer enabled, 200 msec sleep time	0.4	μA
		Conversion in progress at 3.3 V	5	mA
Ambient Light Sensor Current Consumption ⁶	I _{SD}	Shutdown at 1.8 V	0.5	μA
	I _{DD}	Operation mode at 1.8 V (ALS only)	170	μA
RH/Temp Sensor Current Consumption ⁷	I _{DD}	Standby	0.06	μA
		RH conversion in progress	150	μA
		Temperature conversion in progress	90	μA
		Peak current during I ² C operations	3.5	mA
Microphone Current Consumption ⁸	I _{sleep}	Sleep mode at 3.6 V, Fclock = 0 Hz	26	μA
	I _{dd}	Performance mode at 3.6 V, Fclock = 2.4 MHz	700	μA
IMU Current Consumption ⁹	-	Full-chip sleep mode at 1.8 V supply	8	μA
		Gyroscope Only, 102.3 Hz update rate at 1.8 V supply	1.23	mA
		Accelerometer only, 102.3 Hz update rate at 1.8 V supply	68.9	μA
		Gyroscope + Accelerometer, 102.3 Hz update rate at 1.8 V supply	1.27	mA
On-board Debugger Sleep Current Consumption ¹⁰	I _{EM4S}	On-board debugger current consumption when USB cable is not inserted (EFM32GG12 EM4S mode current consumption)	80	nA
Total board Quiescent Current Consumption	I _{TOT}	Typ value for VMCU = 3.0V, Target in EM4S, the USB cable disconnected, all peripherals in sleep mode, sensors disconnected	320	nA

Parameter	Symbol	Condition ¹	Typ	Unit
Note: 1. at +25°C, unless otherwise specified 2. From EFM32PG22 datasheet 3. Derived from datasheet number (28 μA/MHz) 4. From MX25R8035F datasheet 5. From Si7210 datasheet 6. From VEML6035 datasheet 7. From Si7021-A20 datasheet 8. From SPK0641HT4H-1 datasheet 9. From ICM-20648 datasheet 10. From EFM32GG12 datasheet				

3. Hardware

The core of the PG22 Dev Kit is the EFM32PG22 microcontroller. The board also contains several peripherals connected to the EFM32PG22. Placement and layout of hardware components is described in section [1.3 Kit Hardware Layout](#).

3.1 Block Diagram

An overview of the PG22 Dev Kit is illustrated in the figure below.

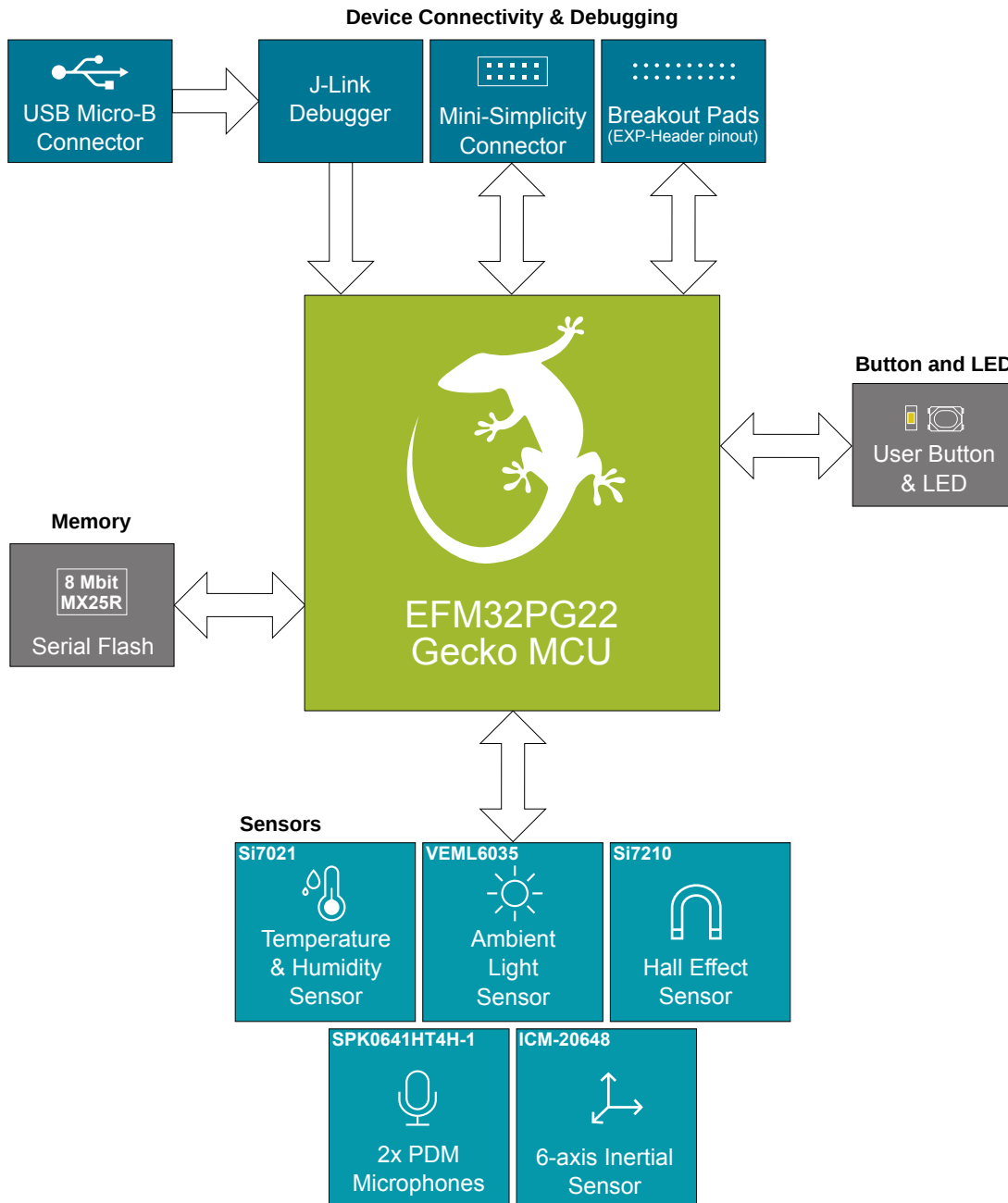


Figure 3.1. Kit Block Diagram

3.2 Power Supply

The kit can be powered through one of these interfaces:

- USB Micro-B
- Battery
- Mini Simplicity connector

The figure below shows the power options available on the kit and illustrates the main system power architecture.

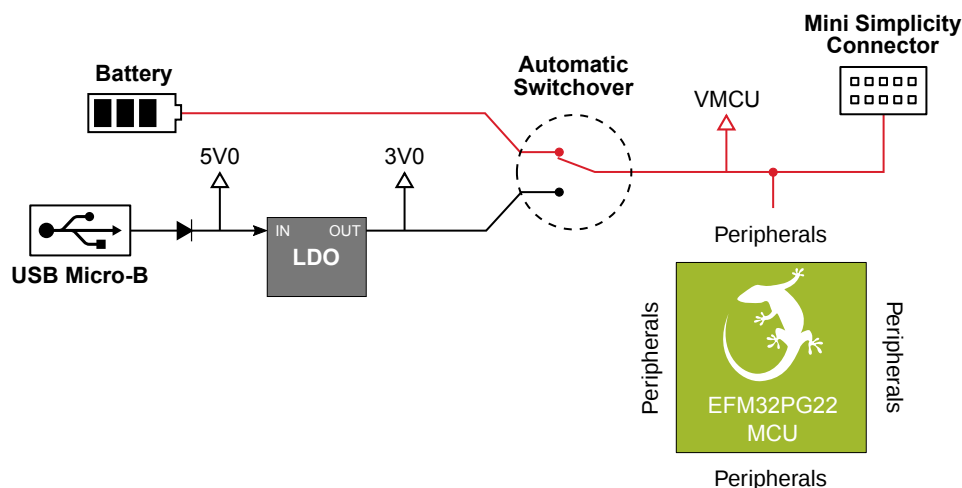


Figure 3.2. PG22 Dev Kit Power Architecture

Power is normally applied either through USB cable or a CR2032 battery. When the USB cable is connected, VBUS is regulated down to 3.0 V. An automatic switchover circuit switches the main system power from battery power to USB power when the USB cable is inserted, and protects the battery from reverse current.

Power can also be applied through the Mini Simplicity connector. This requires that no other power sources are present on the kit, as power is injected directly to the VMCU net. This is important to avoid power conflicts and backfeeding the battery. Powering the PG22 Dev Kit through the Mini Simplicity connector allows current measurements using the Advanced Energy Monitoring (AEM) as described in section [4.2 External Debugger](#).

Important: When powering the board through the Mini Simplicity connector, the USB and battery power sources must be removed.

The power supply options are summarized in the table below.

Table 3.1. PG22 Dev Kit Power Options

Supply mode	Typical input voltage	VMCU Source	3V0	5V
USB power	5.0 V	On-board regulator	On-board regulator	USB VBUS
CR2032 battery	3.0 V	Battery voltage	Disconnected	No voltage present
Mini Simplicity	3.3 V	Debugger dependent	Disconnected	No voltage present

3.3 EFM32PG22 Reset

The EFM32PG22 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.

3.4 Peripherals

The PG22 Dev Kit contains a set of peripherals that can be accessed from the EFM32PG22. All the peripherals have enable signals which can be used to completely turn off the peripherals that are not in use, or they can be put into a state that draws very small amounts of power. This allows for the lowest possible power consumption in every application. The following peripherals are accessible to the EFM32PG22:

- One Silicon Labs Si7021 relative humidity & temperature sensor
- One Silicon Labs Si7210 hall effect sensor
- One Vishay VEML6035 ambient light sensor
- One TDK InvenSense ICM-20648 6-axis inertial measurement sensor
- Two Knowles SPK0641HT4H-1 MEMS microphones with PDM output
- One Macronix MX25R8035F ultra low power 8 Mbit SPI flash
- One LED and one push button

The figure below gives an overview of the peripherals that are connected to the EFM32PG22. Note that some of the peripherals share the same interface and enable signals. As the enable signals does not have external pull-down resistors on the board, it is recommended to actively drive the signals either low or high in the application code to prevent the lines from floating.

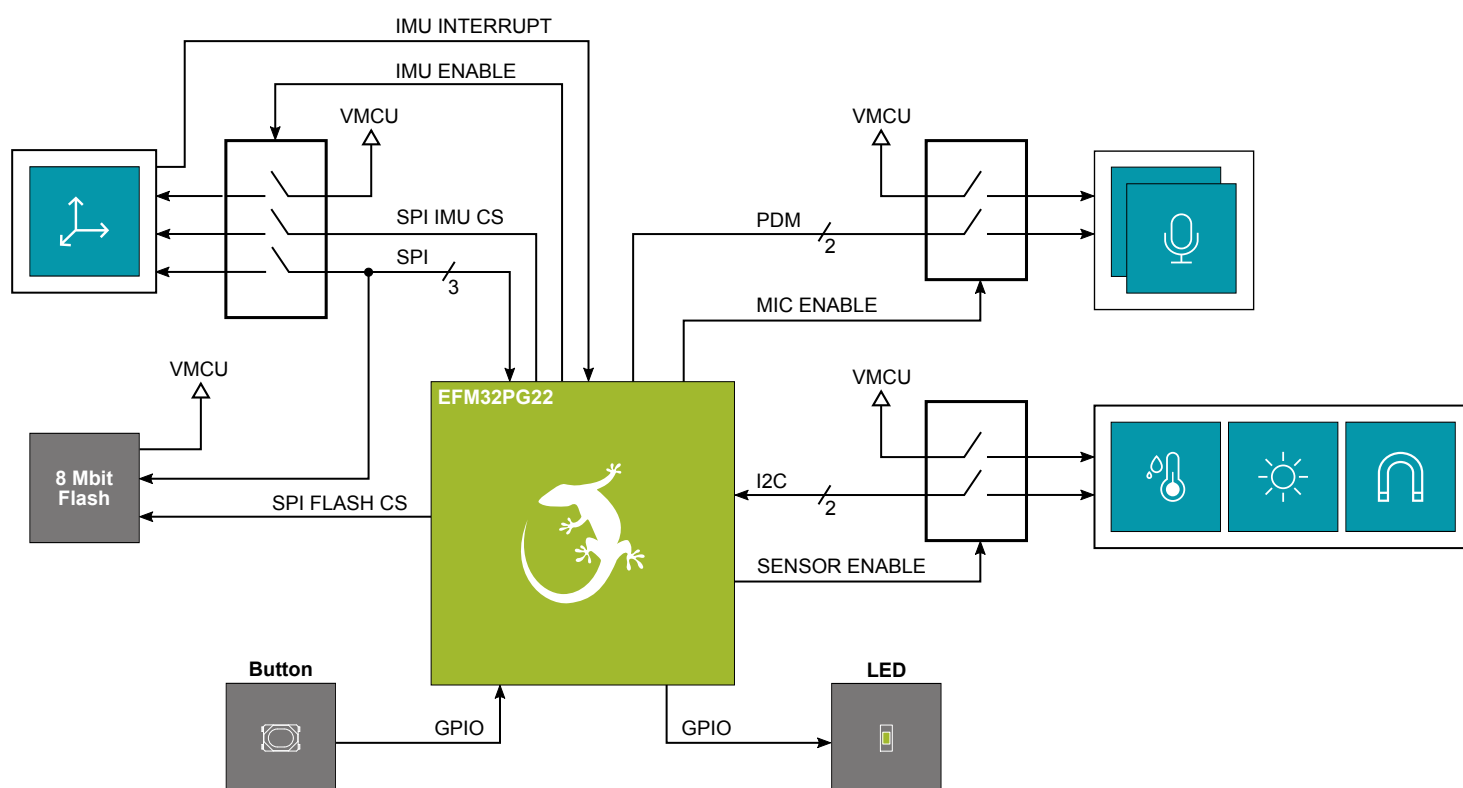


Figure 3.3. Peripherals

Note: Make sure the MCU pins connected to the switch enable inputs are always defined as output.

3.4.1 Si7021 Relative Humidity and Temperature Sensor

The Si7021 I²C relative humidity and temperature sensor is a monolithic CMOS IC integrating humidity and temperature sensor elements, an analog-to-digital converter, signal processing, calibration data, and an I²C interface. The patented use of industry-standard, low-K polymeric dielectrics for sensing humidity enables the construction of low-power, monolithic CMOS Sensor ICs with low drift and hysteresis, and excellent long term stability. The Si7021 offers an accurate, low-power, factory-calibrated digital solution ideal for measuring humidity, dew-point, and temperature, in applications ranging from HVAC/R and asset tracking to industrial and consumer platforms.

On PG22 Dev Kit, the Si7021 is connected through a switch. The switch must therefore be enabled by setting `SENSOR_ENABLE` high before it can be used by the application. This enables power to the Si7021 and connects the I²C lines used for the sensor to the EFM32PG22 I²C bus. The figure below shows how the Si7021 is connected to the EFM32PG22.

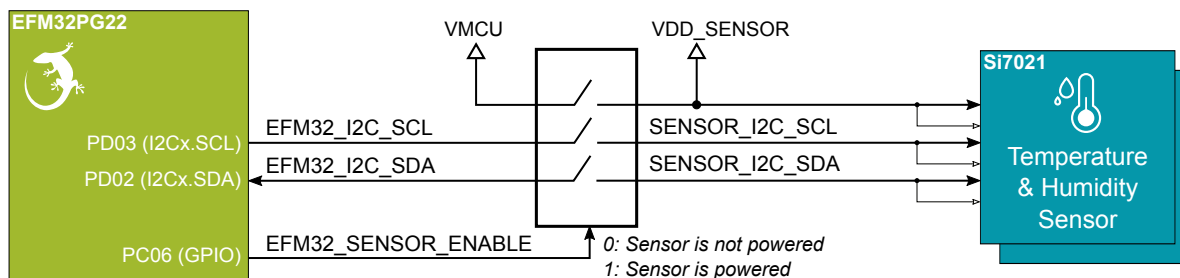


Figure 3.4. Si7021 Relative Humidity and Temperature Sensor

Note: Ensure that `SENSOR_ENABLE` is defined.

Although measures have been taken to thermally isolate the sensor from the board, temperature readings will be influenced when power is dissipated on the board. More accurate temperature measurements are achieved when powering the board with a battery or through the Mini Simplicity connector as self-heating from the on-board LDO is eliminated and the on-board debugger is put in a low power state.

3.4.2 Si7210 Hall Effect Sensor

The Si7210 family of Hall effect sensors from Silicon Labs combines a chopper-stabilized Hall element with a low-noise analog amplifier, 13-bit analog-to-digital converter, and an I²C interface. Leveraging Silicon Labs' proven CMOS design techniques, the Si7210 family incorporates digital signal processing to provide precise compensation for temperature and offset drift. The 13-bit magnetic field strength can be read through the I²C interface at any time. Applications for the Si7210 include mechanical position sensing in consumer, industrial and automotive applications, reed switch replacement, fluid level measurement, speed sensing and control knobs and switches.

On PG22 Dev Kit, the Si7210 is connected through a switch. The switch must therefore be enabled by setting `SENSOR_ENABLE` high before it can be used by the application. This enables power to the Si7210 and connects the I²C lines used for the sensor to the EFM32PG22 I²C bus. The figure below shows how the Si7210 is connected to the EFM32PG22.

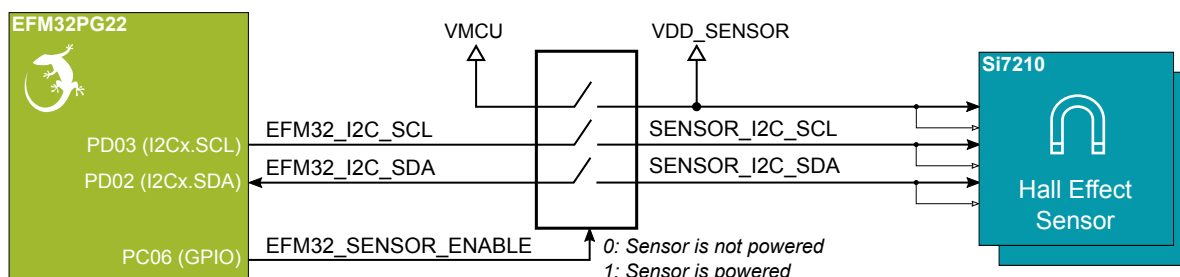


Figure 3.5. Hall Effect Sensor

Note: Ensure that `SENSOR_ENABLE` is defined.

3.4.3 VEML6035 Ambient Light Sensor

The VEML6035 is an ambient light sensor with I²C digital interface.

On PG22 Dev Kit, the VEML6035 is connected through a switch. The switch must therefore be enabled by setting `SENSOR_ENABLE` high before it can be used by the application. This enables power to the VEML6035 and connects the I²C lines used for the sensor to the EFM32PG22 I²C bus. The figure below shows how the VEML6035 is connected to the EFM32PG22.

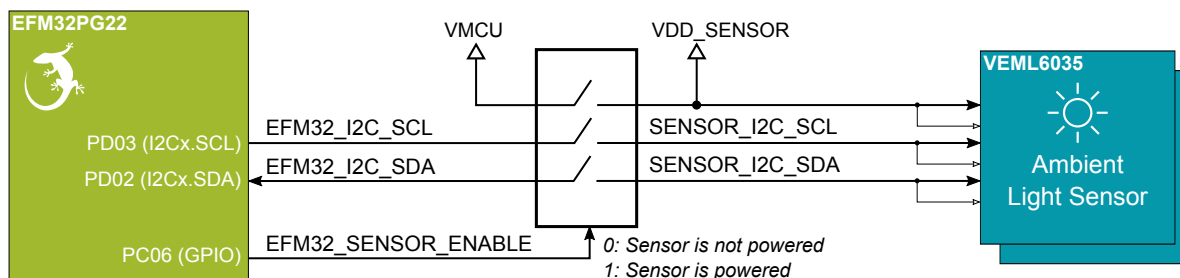


Figure 3.6. VEML6035 Ambient Light Sensor

Note: Ensure that `SENSOR_ENABLE` is defined.

3.4.4 PDM Stereo Microphones

The PG22 Dev Kit features two Knowles SPK0641HT4H-1 digital MEMS microphones with PDM output. The microphones are configured to form a stereo sound input device using only a single PDM data line. The clock to the microphones are fed from a pin on EFM32PG22 with PDM clock support. The output from both microphones are connected to the same line and connected to a pin on the EFM32PG22 supporting PDM data input.

On PG22 Dev Kit, the microphones are connected through a switch. The switch must therefore be enabled by setting `SENSOR_ENABLE` high before it can be used by the application. This enables power to the microphones and connects the PDM lines used for the sensor to the EFM32PG22. The figure below shows how the microphones are connected to the EFM32PG22.

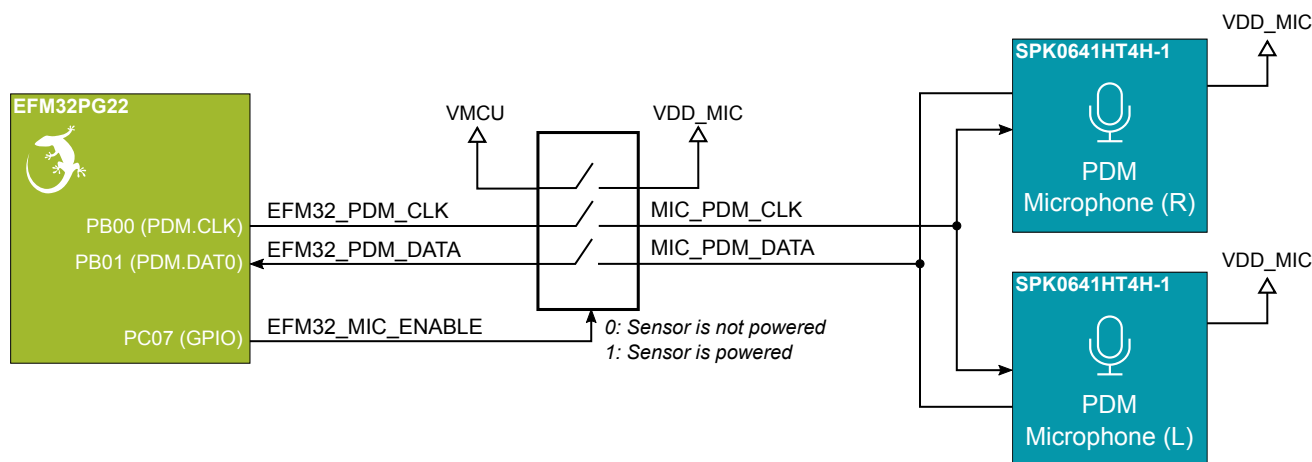


Figure 3.7. Digital Stereo Microphones

Note: Ensure that `SENSOR_ENABLE` is defined.

3.4.5 ICM-20648 6-Axis Inertial Sensor

The ICM-20648 is a 6-axis inertial sensor consisting of a 3-axis gyroscope and a 3-axis accelerometer. The sensor detects acceleration and angular rate in and around the X-, Y- and Z-axes with integrated 16-bit ADCs and programmable digital filters.

On PG22 Dev Kit, the ICM-20648 is connected through a switch. The switch must be enabled by setting `SENSOR_ENABLE` high before it can be used by the application. This enables power to the ICM-20648 and connects the SPI lines used for the sensor to the EFM32PG22 SPI bus. Note the presence of the external pull-up resistor on the interrupt line as this can cause back powering if not handled correctly in software. The figure below shows how the ICM-20648 is connected to the EFM32PG22.

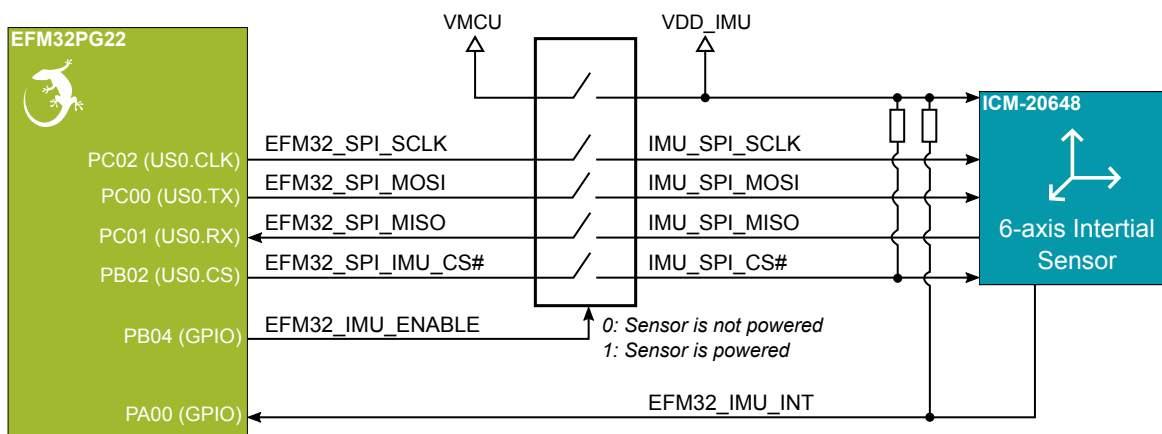


Figure 3.8. ICM-20648 Six-axis Inertial Sensor

Note: Ensure that `SENSOR_ENABLE` is defined.

The inertial sensor is located close to the geometrical center of the board. The coordinate system and rotation of the sensor follows the right-hand rule, and the spatial orientation of the board is shown in the figure below.

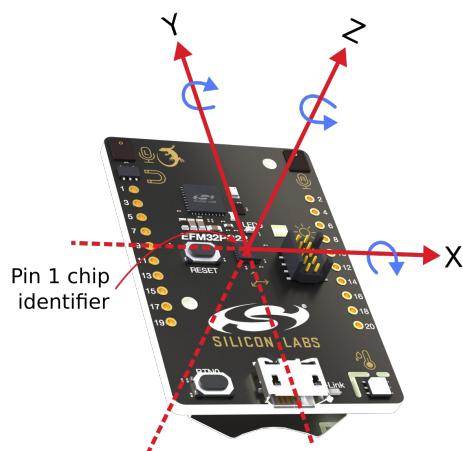


Figure 3.9. PG22 Dev Kit Spatial Orientation

3.4.6 Push Button and LED

The kit has one user push button, marked BTN0, that is connected to a GPIO on the EFM32PG22. The button is connected to pin PB03 and it is debounced by an RC filter with a time constant of 1 ms. The logic state of the button is high while the button is not being pressed, and low when the button is pressed.

The kit also features one yellow LED, marked LED0, that is controlled by a GPIO pin on the EFM32PG22. The LED is connected to pin PA04 in an active-high configuration.

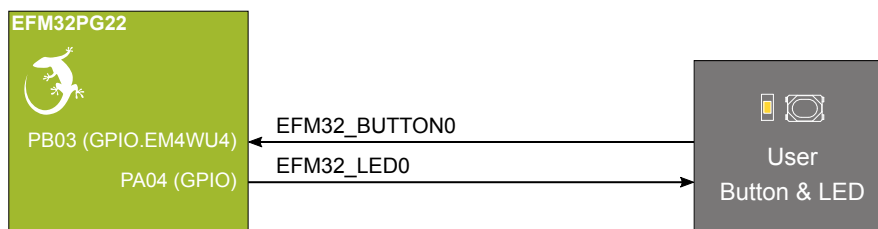


Figure 3.10. Button and LED

3.4.7 External Memory

The PG22 Dev Kit includes an 8 Mbit Macronix SPI Flash that is connected directly to the EFM32PG22. The MX25R series are ultra low power serial flash devices, so there is no need for a separate enable switch to keep current consumption down. However, it is important that the flash is always put in deep power down mode when not used. This is done by issuing a command over the SPI interface. In deep power down, the MX25R typically adds approximately 100 nA to the current consumption. The figure below shows how the serial flash is connected to the EFM32PG22.

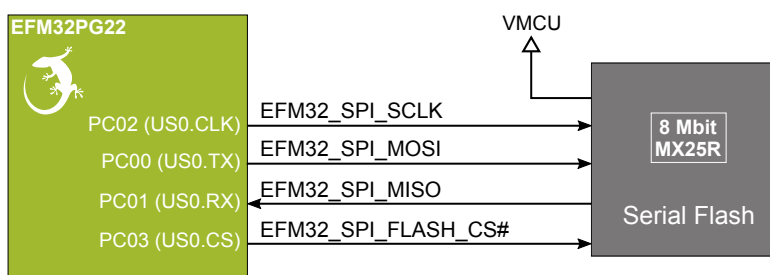


Figure 3.11. Serial Flash

3.5 On-board Debugger

The PG22 Dev Kit contains a microcontroller separate from the EFM32PG22 that provides the user with an on-board J-Link debugger through the USB Micro-B port. This microcontroller is referred to as the "on-board debugger", and is not programmable by the user. When the USB cable is removed, the on-board debugger goes into a very low power shutoff mode (EM4S).

In addition to providing code download and debug features, the on-board debugger also presents a virtual COM port for general purpose application serial data transfer.

The figure below shows the connections between the target EFM32PG22 device and the on-board debugger. The figure also shows the presence of the Mini Simplicity Connector, and how this is connected to the same I/O pins.

Please refer to chapter 4. [Debugging](#) for more details on debugging.

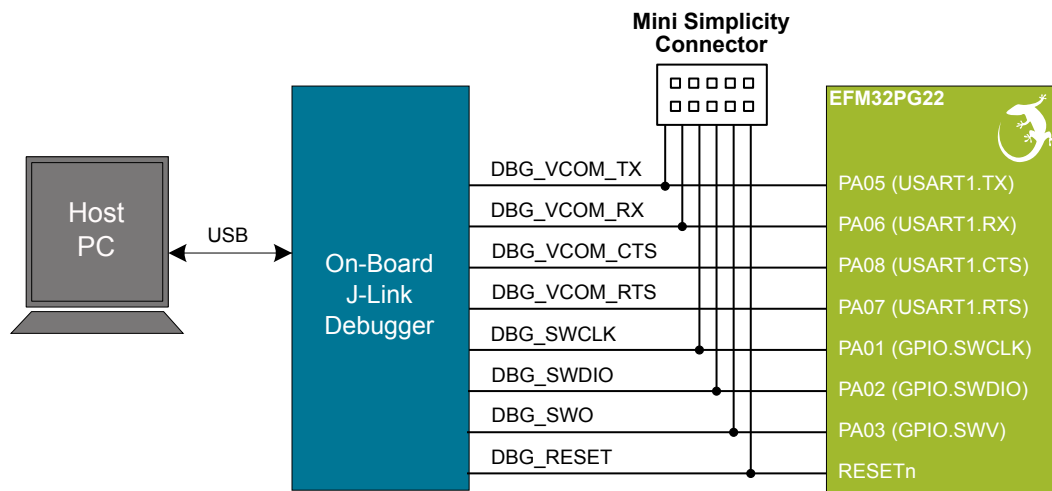


Figure 3.12. On-Board Debugger Connections

3.6 Connectors

Featured on the PG22 Dev Kit is a Mini Simplicity Connector, a USB Micro-B connector and 20 breakout pads that follow the EXP header pinout. The connectors are placed on the top side of the board, and their placement and pinout can be seen in the figure below. For additional information on the connectors see the following sub chapters.

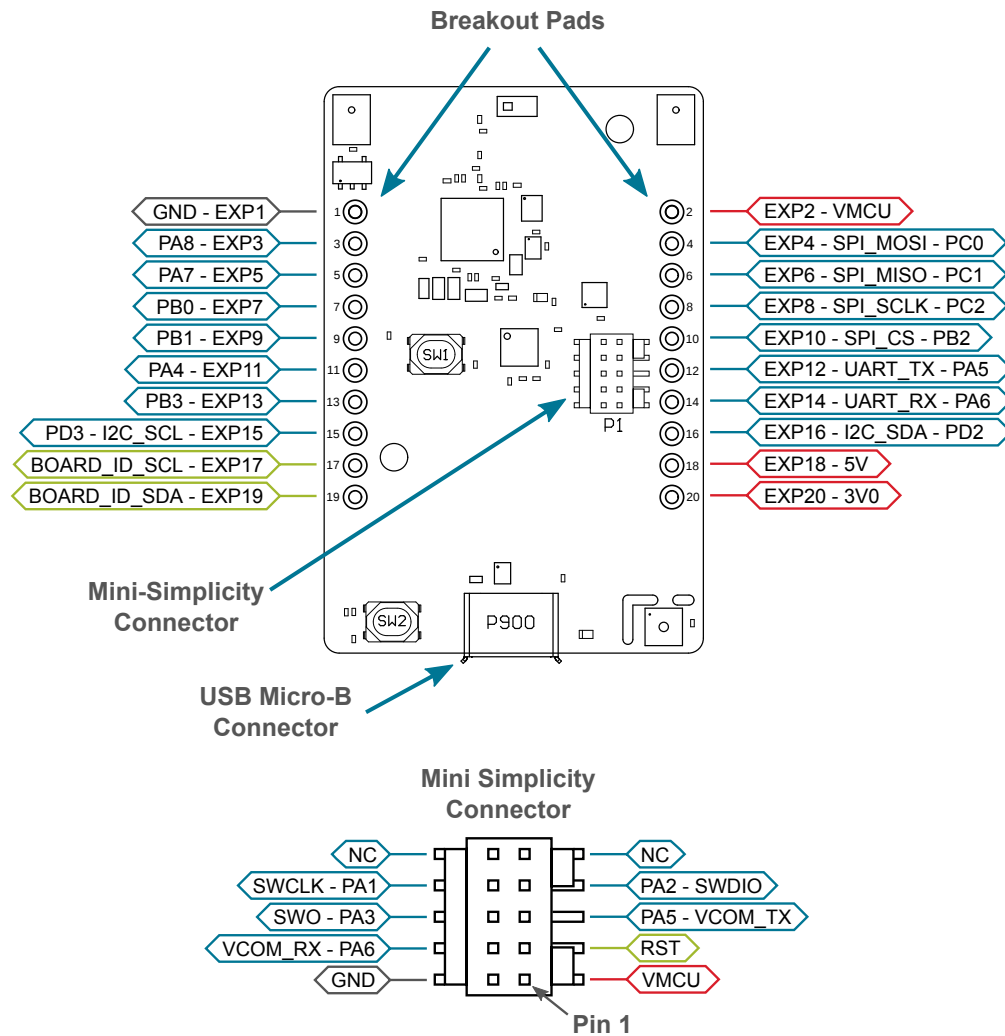


Figure 3.13. PG22 Dev Kit Connectors

3.6.1 Breakout Pads

20 breakout pads, which follow the EXP header pinout, are provided, and allow connection of peripherals or add-on boards. Ten of the pads are located along the left side of the board and ten are located on the right side. The breakout pads expose I/O pins that can be used with most of the EFM32PG22's features. Additionally, the VMCU (main power rail), 3V0 (LDO regulator output) and 5V power rails are also exposed.

The breakout pads are pinned out similar to the EXP header found on other Silicon Labs Starter Kits, which ensures that commonly used peripherals such as SPI, UART and I²C buses are available on fixed locations. The rest of the pins are used for general purpose IO. This allows the definition of EXP boards that can plug into a number of different Silicon Labs starter kits.

The pin-routing on EFM32 is very flexible, so most peripherals can be routed to any pin. However, pins may be shared between the breakout pads and other functions on the PG22 Dev Kit. The table below includes an overview of the EXP header and functionality that is shared with the kit.

Table 3.2. Expansion Header Pinout

Pin	Connection	EXP Header Function	Shared Feature	Peripheral Mapping
Right Side Breakout Pins				
2	VMCU	EFM32PG22 voltage domain, included in AEM measurements.		
4	PC00	SPI_MOSI	IMU & Flash MOSI	US0_TX
6	PC01	SPI_MISO	IMU & Flash MISO	US0_RX
8	PC02	SPI_SCLK	IMU & Flash SCLK	US0_CLK
10	PB02	SPI_CS	IMU CS	US0_CS
12	PA05	UART_TX	VCOM Tx	US1_TX
14	PA06	UART_RX	VCOM Rx	US1_RX
16	PD02	I2C_SDA	Sensor I2C bus SDA	I2Cx_SDA
18	5V	Board USB voltage		
20	3V0	Board controller supply		
Left Side Breakout Pins				
1	GND	Ground		
3	PA08	GPIO	VCOM CTS	
5	PA07	GPIO	VCOM RTS	
7	PB00	GPIO	PDM Microphones CLK	
9	PB01	GPIO	PDM Microphones DATA	
11	PA04	GPIO	LED	
13	PB03	GPIO	Button	
15	PD03	I2C_SCL	Sensor I2C bus SCL	I2Cx_SCL
17	BOARD_ID_SCL	Connected to Board Controller for identification of add-on boards.		
19	BOARD_ID_SDA	Connected to Board Controller for identification of add-on boards.		

3.6.2 Mini Simplicity Connector

The Mini Simplicity connector is a 10-pin 1.27 mm pitch connector that allows the use of an external debugger such as the one found on a Silicon Labs Wireless Starter Kit (WSTK) mainboard. See section [4.2 External Debugger](#) for more details. The pinout of the connector on the board is described in the table below with the names being referenced from the EFM32PG22.

Table 3.3. Mini Simplicity Connector Pin Descriptions

Pin number	Connection	Function	Description
1	VMCU	VAEM	Target voltage on the debugged application. Supplied and monitored by the AEM when powered by a WSTK with its power selection switch in the "AEM" position.
2	GND	GND	Ground.
3	EFM32PG22 reset pin	DBG_RST	Reset.
4	PA06	VCOM_RX	VCOM Rx.
5	PA05	VCOM_TX	VCOM Tx.
6	PA03	DBG_SWO	Serial Wire Output.
7	PA02	DBG_SWDIO	Serial Wire Data.
8	PA01	DBG_SWCLK	Serial Wire Clock.
9	NC	None	Not Connected.
10	NC	None	Not Connected.

3.6.3 Debug USB Micro-B Connector

The debug USB port can be used for uploading code, debugging, and as a Virtual COM port. More information is available in section [4. Debugging](#).

4. Debugging

The PG22 Dev Kit contains an on-board SEGGER J-Link Debugger that interfaces to the target EFM32PG22 using the Serial Wire Debug (SWD) interface. The debugger allows the user to download code and debug applications running in the target EFM32PG22. Additionally, it also provides a VCOM port to the host computer that is connected to the target device's serial port, for general purpose communication between the running application and the host computer. The Packet Trace Interface (PTI) is also supported by the on-board debugger which offers invaluable debug information about transmitted and received packets in wireless links. The on-board debugger is accessible through the USB Micro-B connector.

An external debugger can be used instead of the on-board debugger by connecting it to the Mini Simplicity Connector. This allows advanced debugging features as described in section 4.2 [External Debugger](#). When using an external debugger it is very important to make sure that there is no power source present on the PG22 Dev Kit, as the external debugger might source a voltage on the target power domain (VMCU).

Important: When connecting an external debugger that sources voltage to the VMCU net, the USB cable and battery must be removed from the PG22 Dev Kit. Failure to do so will create power conflicts.

The figure below shows the possible debug options.

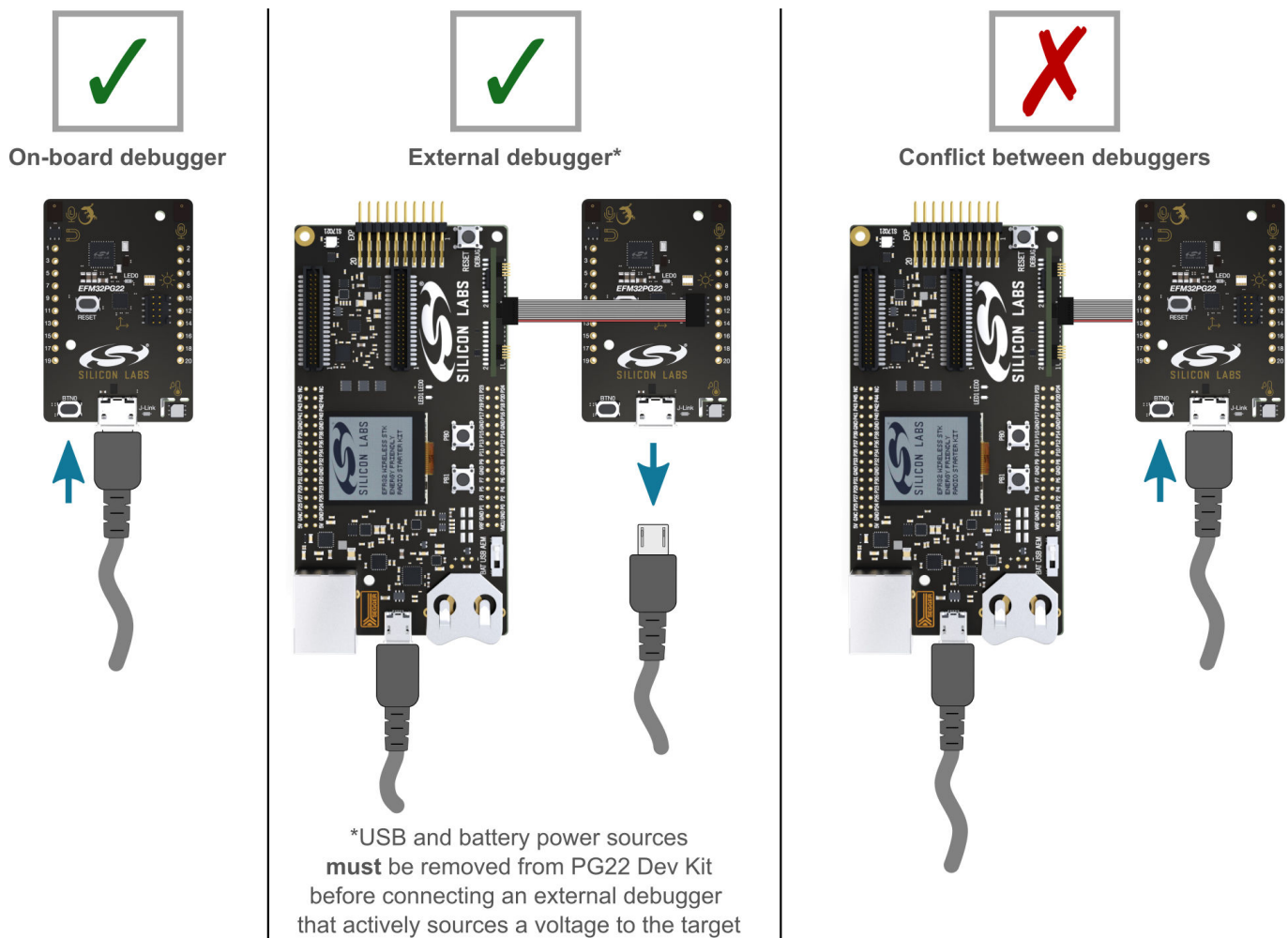


Figure 4.1. PG22 Dev Kit Debugging Possibilities

4.1 On-board Debugger

The on-board debugger is a SEGGER J-Link debugger running on an EFM32 Giant Gecko. The debugger is directly connected to the debug and VCOM pins of the target EFM32PG22.

When the debug USB cable is inserted, the on-board debugger is automatically active, and takes control of the debug and VCOM interfaces. This means that debug and communication will *not* work with an external debugger connected at the same time. The on-board LDO is also activated, providing power to the board.

When the USB cable is removed, the board might still be running on battery power, as described in section [3.2 Power Supply](#). In this case, the on-board debugger goes into a very low power shutoff mode (EM4S), consuming about 80 nA. This means that battery life-time will not be affected too much by the on-board debugger power consumption. Since the I/O voltage rail of the debugger remains powered in the battery operated mode, the pins connected to the debug and VCOM interfaces maintain proper isolation and prevent leakage currents.

4.2 External Debugger

A Wireless Starter Kit (Wireless STK) mainboard with a debug adapter board from Silicon Labs can be connected to the Mini Simplicity Connector and used for debugging instead of the on-board debugger. For instruction on how this can be done see "AN958: Debugging and Programming Interfaces for Custom Designs". Debugging with an external Wireless STK mainboard gives access to the following debugging features:

- Debugging of the target device through SWD
- Communication using the VCOM port
- Packet Trace Interface (for wireless devices only)
- Advanced Energy Monitor

Note that the Mini Simplicity Connector *cannot* be used at the same time as the on-board debugger is active (USB cable is plugged in). For information on how to correctly connect to the kit, see [Figure 4.1 PG22 Dev Kit Debugging Possibilities on page 19](#).

Powering the board when using the Mini Simplicity Connector with a Wireless STK mainboard and adapter board can be done using the AEM voltage supply of the Wireless STK mainboard. When doing this, remove both the USB cable and the coin cell battery from the PG22 Dev Kit before connecting the Wireless STK mainboard to the Mini Simplicity Connector. The power switch on the Wireless STK mainboard should be set in "AEM". Power-cycling of the board, if necessary, can easily be done by flipping the power switch on the Wireless STK to "BAT" and back to "AEM" assuming a battery is not inserted in the Wireless STK mainboard.

It is possible to have the PG22 Dev Kit powered by a battery, and still use the Mini Simplicity Connector with a Wireless STK mainboard for debugging and communication. In this case the power switch on the Wireless STK mainboard must be set to the "BAT" position and the coin cell battery on the Wireless STK mainboard must be removed. In this case level shifters on the Wireless STK itself takes care of interfacing to different voltage levels on the PG22 Dev Kit. Connecting the board to an external debugger in other ways than those described above might create power conflicts, compromise the ability to monitor power consumption and might hazardously feed power back to the on-board battery.

Important: Always remove the battery if you are not sure whether the external debugger is sourcing voltage to PG22 Dev Kit.

4.3 Virtual COM Port

The virtual COM port (VCOM) is a connection to a UART on the EFM32PG22, and allows serial data to be sent and received from the device. The on-board debugger presents this as a virtual COM port on the host computer that shows up when the USB cable is inserted.

Data is transferred between the host computer and the debugger through the USB connection, which emulates a serial port using the USB Communication Device Class (CDC). From the debugger, the data is passed on to the target device through a physical UART connection.

The serial format is 115200 bps, 8 bits, no parity, and 1 stop bit by default.

Note: Changing the baud rate for the COM port on the PC side does not influence the UART baud rate between the debugger and the target device. However, it is possible to change the VCOM Baudrate through the kits Admin Console available through Simplicity Studio.

Alternatively, the VCOM port can also be used through the Mini Simplicity Connector with an external Wireless STK. Using the VCOM port through the Mini Simplicity Connector with an external Wireless STK works in a similar way, but requires that the USB cable to the on-board debugger is unplugged. The board controller on the Wireless STK then makes the data available over USB (CDC) or an IP socket. Note that flow control is not available over the Mini Simplicity Connector.

5. 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: <http://www.silabs.com/>

6. Kit Revision History and Errata

6.1 Revision History

The kit revision can be found printed on the box label of the kit, as outlined in the figure below. The kit revision history is summarised in the table below.

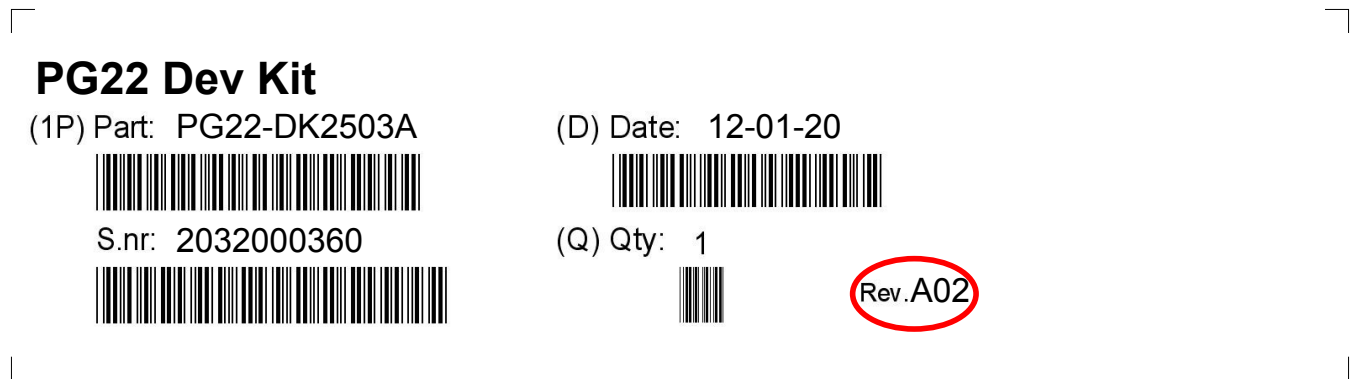


Figure 6.1. Revision Info

Table 6.1. Kit Revision History

Kit Revision	Released	Description
A02	2020-12-01	Initial kit release with BRD2503A Rev. A02.

6.2 Errata

There are no known errata at present.

7. Board Revision History and Errata

7.1 Revision History

The board revision can be found laser printed on the board, and the board revision history is summarised in the table below.

Table 7.1. Board Revision History

Revision	Released	Description
A02	2020-11-23	Initial version.

7.2 Errata

There are no known errata at present.

8. Document Revision History

Revision 1.0

December, 2020

- Initial document release.

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.

Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, ClockBuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, Gecko OS, Gecko OS Studio, ISOModem®, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, 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 Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

<http://www.silabs.com>