Sidekiq Z2 Hardware User's Manual

Version 1.1

EPIQ SOLUTIONS

Disclaimer

Epiq Solutions is disclosing this document (“Documentation”) as a general guideline for development. Epiq Solutions expressly disclaims any liability arising out of your use of the Documentation. Epiq Solutions reserves the right, at its sole discretion, to change the Documentation without notice at any time. Epiq Solutions assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Epiq Solutions expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU “AS IS” WITH NO WARRANTY OF ANY KIND. EPIQ SOLUTIONS MAKES NO OTHER WARRANTIES, WHETHER EXPRESSED, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT WILL EPIQ SOLUTIONS BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

Revision History

Date	Revision	Description
04/02/2018	0.1	Initial version
05/16/2018	0.2	Added additional details for GPIO usage, re-flashing Z2 over DFU, power consumption specs, and general usage
07/12/2018	0.3	Added additional details for USB networking options, simple carrier card usage, memory partition info, and I/O expansion card info. Added statement of volatility.
10/10/2018	1.0	Updated Tables 3, 5, 6, 7, 8, and 9 Added Section 10.5.3 regarding the JTAG jumper for JTAG_BOOT_EN access Added temperature and IMU sensor info to Figure 2 Added Section 8.9 Added Rev C change summary
12/05/2018	1.1	Updated Figure 1, Table 13, Sections 8.7 and 10.8.2

Table of Contents

Table of Figures

Table of Tables

1 Introduction

This document provides an overview of Epiq Solutions' Sidekiq Z2 software defined radio (SDR) card [1], a highly integrated wideband RF transceiver plus Linux computer integrated on to a MiniPCIe card measuring 30mm x 51mm x 5mm. The following topics will be discussed:

- Overview of the Sidekiq Z2 hardware and available interfaces
- Sidekiq Z2 usage/integration options
- Sidekiq development/debug board usage

Sidekiq Z2 is available in an Evaluation Kit (EVK) as well as with a Platform Development Kit (PDK) upgrade. For customers procuring the EVK, Sidekiq Z2 is pre-loaded with a reference design based on Analog Devices' Industrial I/O (IIO) software & FPGA reference design, an open source board support package available at [4]. All support-related questions are managed through Analog Devices' web-based support forum available at [3].

For customers procuring the PDK upgrade, Sidekiq Z2 is pre-loaded with Epiq Solutions standard Sidekiq reference design, supporting the libsidekiq API and an optimized FPGA reference design for customers wanting to enable more FPGA-based processing. All support-related questions are managed through Epiq Solutions' private web-based support forum available at [2]. Please note that it is necessary to register prior to accessing the relevant information for the PDK. Additional details comparing the EVK and PDK options can be found in section 9 of this manual.

2 Legal Considerations

Sidekiq Z2 is distributed all over the world. Each country has its own laws governing reception and transmission of radio frequencies. The user of Sidekiq Z2 and associated software is solely responsible for insuring that it is used in a manner consistent with the laws of the jurisdiction in which it is used. Many countries, including the United States, prohibit the transmission and reception of certain frequency bands, or receiving certain transmissions without proper authorization. Again, the user is solely responsible for the user's own actions.

3 Proper Care and Handling

Each Sidekiq Z2 unit is fully tested by Epiq Solutions before shipment, and is guaranteed functional at the time it is received by the customer, and ONLY AT THAT TIME. Improper use of Sidekiq can cause it to become non-functional. In particular, a list of actions that may cause damage to the hardware include the following:

- Handling the unit without proper static precautions (ESD protection) when the housing is removed or opened up
- Inserting or removing Sidekiq from a host system when power is applied to the host system
- Connecting a transmitter to the RX port without proper attenuation – see the Specifications section for details on maximum RF signal input levels
- Executing custom software and/or an FPGA bitstream that was not developed according to guidelines

The above list is not comprehensive, and experience with the appropriate measures for handling electronic devices is required.

4 References

[1] **Sidekiq Z2 Product Page**
https://epiqsolutions.com/modules/sidekiq-z2/

[2] **Epiq Solutions Support Portal**
https://www.epiqsolutions.com/support

[3] **Analog Devices' Support Portal**
https://ez.analog.com/

[4] **Analog Devices' IIO Reference Design & Board Support Package for Sidekiq Z2**
https://github.com/epiqsolutions/plutosdr-fw

[5] **GPIO Sysfs Interface for Userspace**
https://www.kernel.org/doc/Documentation/gpio/sysfs.txt

[6] **Zynq-7000 Technical Reference Manual**
https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf

[7] **DFU Utility**
http://dfu-util.sourceforge.net/

[8] **Berquist Thermal Gap Pad Solution**
https://www.bergquistcompany.com/

5 Terms and Definitions

Term	Definition
A/D	Analog to Digital converter
BSP	Board Support Package
D/A	Digital to Analog converter
dB	Decibel
DFU	Device Firmware Update
ESD	ElectroStatic Discharge
EVK	Evaluation Kit
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input / Output
GPS	Global Positioning System
IIO	Industrial Input / Output
IMU	Inertial Measurement Unit
I/Q	In-Phase / Quadrature Phase
LED	Light Emitting Diode
MHz	Megahertz
OTG	On The Go, a variant of the USB protocol supporting both host and device operation
PDK	Platform Development Kit
PL	Programmable Logic
PPS	Pulse Per Second
PS	Processing System
RF	Radio Frequency
Rx	Receive
SCC	Simple Card Carrier
SDK	Software Development Kit
SDR	Software Defined Radio
SSH	Secure SHell
TCVCXO	Temperature Compensated Voltage Controlled Crystal Oscillator
Tx	Transmit
UART	Universal Asynchronous Receiver Transmitter
U.FL	Miniature RF connector manufactured by Hirose
USB	Universal Serial Bus
W.FL	Micro-Miniature RF connector manufactured by Hirose

Table 1: Terms and Definitions

6 System Overview

Sidekiq Z2 is a small form factor software defined radio card that provides a wideband RF transceiver plus a Linux computer in a standard MiniPCIe card module measuring 30mm x 51mm x 5mm. With the integrated Linux computer, Sidekiq Z2 is targeted for applications where the necessary signal/protocol processing tasks can execute directly on the card itself, without the necessity of a host computer platform. Simple carrier cards can be developed for Sidekiq Z2 to customize the I/O available in the system. Alternately, Sidekiq Z2 can be integrated in to any host system with a full size MiniPCIe slot, where supplemental processing could take place on the host system if needed.

A high level summary of Sidekiq Z2 features is shown below:

- Leverages Analog Devices' AD9364 to provide a 1x1 RF transceiver covering 70 MHz to 6 GHz, with independent Tx and Rx frequencies
- Integrated four band Rx pre-select filter
- Configurable A/D and D/A sample rates up to 61.44 Msamples/sec
- Integrated Linux computer provided via Xilinx Zynq XC7Z010-2I System-on-Chip
- 512 MB of DDR3L RAM
- 32 MB of QSPI flash storage for uboot bootloader, Linux kernel, and root filesystem
- USB 2.0 OTG interface, serial console, and Zynq PS/PL GPIO available via edge connector
- Typical power consumption under 2W
- Achievable Linux boot time: ~2 seconds

A block diagram of Sidekiq Z2 is shown in Figure 1.

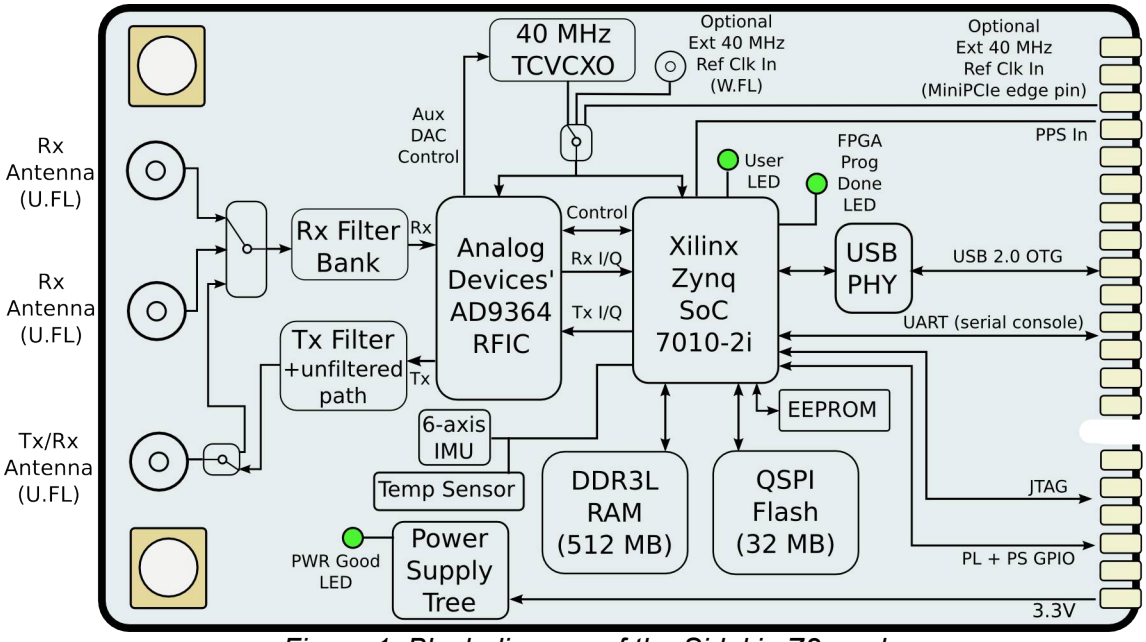


Figure 1: Block diagram of the Sidekiq Z2 card

7 Hardware Specification

7.1 RF Receiver Specification

RF Input	U.FL miniature coaxial connector (50 ohms); total of three U.FL connectors that can be used to switch between three external antennas (two dedicated Rx ports plus one dual-use Tx or Rx port)
Architecture	Zero-IF (direct conversion)
Tuning Range	70 MHz to 6 GHz
Rx Pre-Select Filter Bank	Band 1: 50 MHz to 435 MHz Band 2: 435 MHz to 900 MHz Band 3: 900 MHz to 1950 MHz Band 4: 1950 MHz to 6000 MHz
Tuning Step Size	~2.4 Hz
Tuning Time	~1 mS
Typical Noise Figure	8 dB
Typical IIP3	-10 dBm
Max RF Input Power Level (without damage)	+20 dBm
Gain Control Range	0 to 76 dB, 1 dB steps
Gain Control Mode	Manual or Automatic
A/D Converter Sample Rate	233 Ksamples/sec to 61.44 Msamples/sec
A/D Converter Sample Width	12 bits
Typical I/Q balance	> 50 dB
On-board Reference Clock	40 MHz, +/- 1PPM accuracy (shared with Tx)

7.2 RF Transmitter Specification

RF Input	U.FL miniature coaxial connector (50 ohms); dual-use connector supports operation as Tx or Rx port
Architecture	Zero-IF (direct conversion)
Tuning Range	70 MHz to 6 GHz
Tx Filter Bank	Band 1: 50 MHz to 3000 MHz Band 2: 3000 MHz to 6000 MHz
Tuning Step Size	~2.4 Hz
Tuning Time	~1 mS
Gain Control Range	0 to 80 dB, 0.25 dB steps
RF Output Power	+5 dBm
D/A Converter Sample Rate	233 Ksamples/sec to 61.44 Msamples/sec
D/A Converter Sample Width	12 bits
Typical I/Q balance	> 50 dB
On-board Reference Clock	40 MHz, +/- 1PPM accuracy (shared with Rx)

7.3 Digital Specification

System on Chip	Xilinx Zynq XC7Z010-2I Programmable Logic (PL) Specification: -28K Logic Cells -2.1 Mbits BlockRAM -80 DSP slices Processor System (PS) Specification: -Dual-core ARM Cortex A9 CPU running up to 733 MHz -Linux 4.11
RAM	512 MB DDR3L RAM
Flash	32 MB QSPI
User I/O at MiniPCIe Edge Connector	-USB 2.0 OTG high speed, supports host and device -Linux serial console -JTAG -GPIO -Three PL (FPGA controlled) general purpose input/output pins -Two PS (CPU controlled) general purpose input pins -Three PS (CPU controlled) open drain output pins -Two PS (CPU controlled) general purpose input/output pins
Temperature Sensor Texas Instruments TMP103AYFFR	Accuracy: -40 deg C to +125 deg C (+/- 1 deg C typ) Resolution: 1 deg C
Inertial Measurement Unit (IMU) Sensor 6-axis MotionTracking Device (3-axis gyroscope, 3-axis accelerometer) TDK / InvenSense ICM-20602	- Gyroscope sensitivity error: ±1% - Gyroscope noise: ±4 mdps/√Hz - Accelerometer noise: 100 µg/√Hz
Component Temperature Rating	*-30 deg C to + 85 deg C *Operation down to -40 deg C is supported, though the TCVCXO may operate outside of the +/- 1PPM accuracy specification.

7.4 Sidekiq Z2 Revision Changelog

- Sidekiq Z2 Rev C changes include:
  - PCB updates were made to remove some unused circuitry
  - Added IMU sensor - ICM-20602 - to I2C bus.
  - Minor changes to RX signal path to improve performance above 4 GHz
  - Added "JTAG BOOT" mode to recover the board from failed image loads.
  - Driving pin mPCIe pin 49 high (1.8 to 3.3V) before power-up will set Zynq to "JTAG BOOT" mode.

8 Hardware Interfaces

Sidekiq Z2 provides a variety of different hardware interfaces for use by an end user. Each of these hardware interfaces is shown in Figure 2, and defined below.

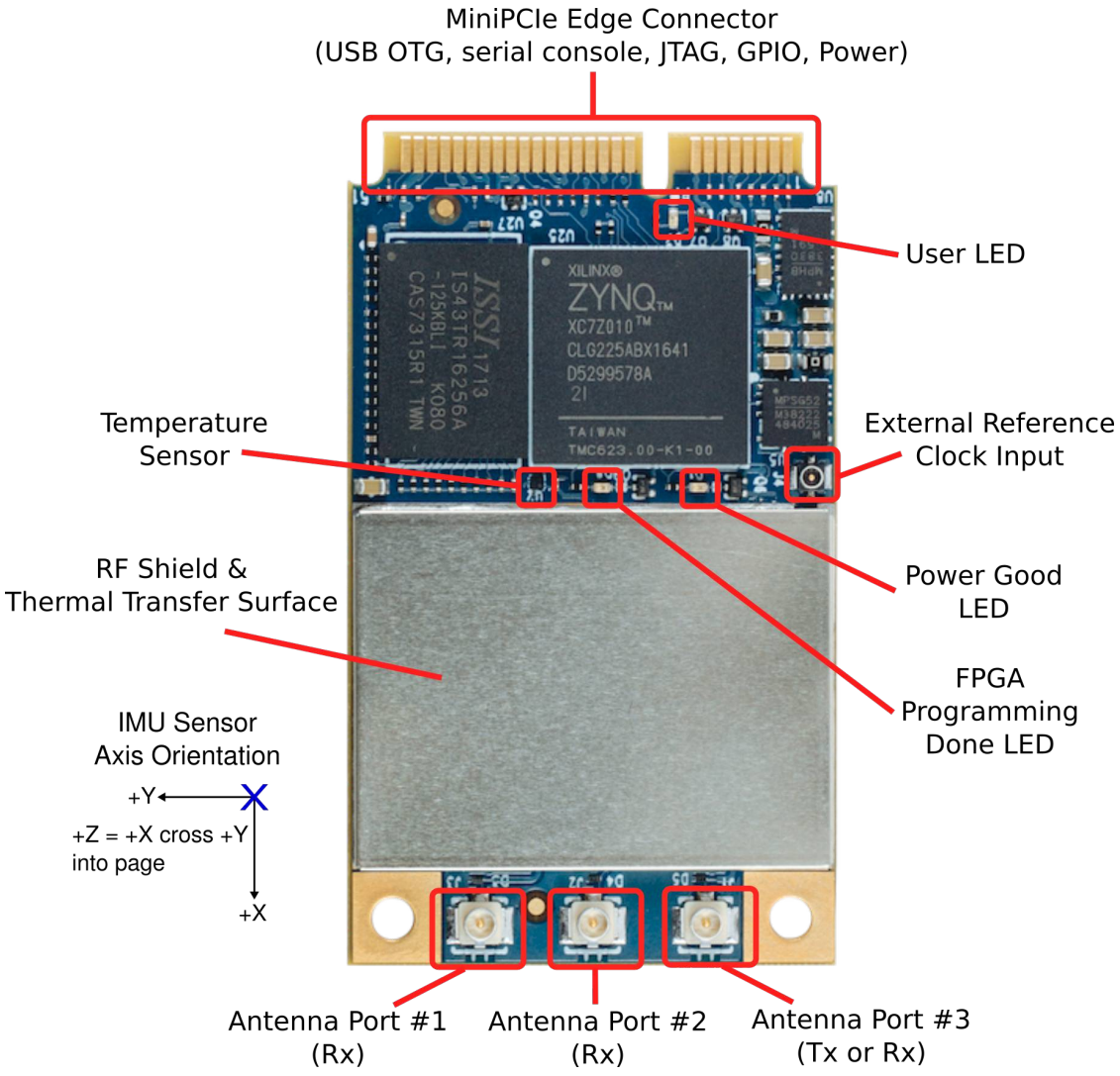


Figure 2: Annotated diagram of Sidekiq Z2 hardware I/O interfaces

8.1 Antenna Port #1 (Rx)

The Antenna Port #1 Rx interface is a U.FL jack connector that provides an antenna input path that can be switched in to the RF receiver in Sidekiq Z2, allowing a user-specified antenna to interface to the RF receiver. This Rx port supports RF input frequencies between 50 MHz and 6 GHz.

8.2 Antenna Port #2 (Rx)

The Antenna Port #2 Rx interface is a U.FL jack connector that provides an antenna input path that can be switched in to the RF receiver in Sidekiq Z2, allowing a user-specified antenna to interface to the RF receiver. This Rx port supports RF input frequencies between 50 MHz and 6 GHz.

8.3 Antenna Port #3 (Tx or Rx)

The Antenna Port #3 TRx interface is a U.FL jack connector that provides an antenna access path that can be switched to route to either the RF receiver or the RF transmitter in Sidekiq Z2, allowing a user-specified antenna to interface to the RF receiver or transmitter. This port supports RF frequencies between 50 MHz and 6 GHz. The selection of either transmit or receive mode for this antenna port can be switched on the fly at run time via software to support TDD applications.

8.4 User LED (Heartbeat LED)

The User LED #1 provides a visual status indicator that can be controlled through software. This LED is connected to a PS pin (PS_MIO_15) on the Zynq SoC. By default, this LED is used to provide a heartbeat indicator after the system has booted Linux and is running, blinking at a rate of ~2 Hz.

8.5 FPGA Programming Done LED

The FPGA Programming Done LED illuminates after a successful loading of an FPGA bitstream into the PL.

8.6 Power Good LED

The Power Good LED illuminates after the power supplies on the card have successfully brought up all the voltage rails needed by the card.

8.7 External Reference Clock Input

The External Reference Clock Input interfaces (software selectable) are available via a W.FL jack connector or the mPCIe edge-connector that allows an external 40 MHz reference clock to optionally be brought in to Sidekiq and utilized instead of the default on-board 40 MHz TCVCXO. This provides the facility to have multiple Sidekiq cards share a common external 40 MHz reference clock.

The electrical specification for this input signal is defined below.

Coupling	AC
Frequency	40 MHz
Peak-to-peak voltage swing	0.8V to 1.3V (on an open circuit; do not exceed 1.3V)
Waveform	squarewave or sinewave

Table 2: Electrical specification for external reference clock input

8.8 RF Shield & Thermal Transfer Surface

The RF shield (used to minimize the effects of RF noise entering the RF front end) serves as the primary thermal transfer path for heat dissipation in the system. Underneath the shield, thermal gap pad material is used to transfer heat from components to the shield itself, yielding a minimal thermal resistance. If no air flow is available in the host system where Sidekiq Z2 is being integrated, it is highly recommended that the user provide a thermal dissipation path from this shield to a thermally conductive surface in the host system, such as a metal back plate or other metal housing. The use of thermal gap pad material [8] can provide a flexible yet efficient thermal path between the RF shield and the host system.

In addition to the RF shield, the other primary component generating heat in Sidekiq is the Xilinx Zynq SoC (which is located right next to the RF shield). If an end user is developing a thermal dissipation path for the RF shield, it is also recommended to include the SoC in the thermal transfer path as well. Similar to the RF shield, use of a thermal gap pad material can be very effective in ensuring good thermal conductivity between this component and the hostmotionTracking system.

8.9 Temperature & Inertial Measurement Unit (IMU) Sensors

The Sidekiq Z2 is equipped with a temperature sensor for monitoring on-board temperature and a IMU sensor for detecting orientation and tracking rotation or twist.

- Texas Instruments Temperature Sensor TMP103AYFFR
- TDK / InvenSense High Performance 6-Axis MEMS MotionTracking™ Device ICM-20602

Please refer to Figure 2 for the IMU axis orientation & temperature sensor location.
The libsidekiq software API provides access to these peripherals and test applications such as *read_temp* (included with the libsidekiq software bundle) demonstrate how to read the sensor's value. Additional information can be found in the Sidekiq Software Development manual.

8.10 MiniPCIe Edge Connector

The MiniPCIe Edge Connector is used to route various signals between Sidekiq Z2 and the host carrier card, including the USB 2.0 OTG high speed interface, the Linux serial console, PS GPIO, PL GPIO, JTAG signals for the Zynq SoC, external reference clock input, and the 3.3V power input required to power the card.
A complete table enumerating the pins and their usage on Sidekiq Z2 is shown in Table 3.

Pin #	MiniPCIe Pin Name	Description as used in Sidekiq
1	WAKE_	No connection
3	COEX1	Console UART output from Z2, 1.8V logic
5	COEX2	Console UART input to Z2, 1.8V logic
7	CLKREQ_	No connection
9	GND	Ground
11	REFCLK-	No connection
13	REFCLK+	No connection
15	GND	Ground
17	UIM_IC_DM	PS_GPIO7: Zynq pin D8 / MIO0, 1.8V I/O
19	UIM_IC_DP	PS_GPIO6: Zynq pin B9 / MIO14, 1.8V I/O
21	GND	Ground
23	PERn0	No connection
25	PERp0	No connection
27	GND	Ground
29	GND	Ground
31	PETn0	No connection
33	PETp0	No connection
35	GND	Ground
37	GND	Ground
39	+3.3Vaux	+3.3V supply
41	+3.3Vaux	+3.3V supply
43	GND	Ground
45	RESERVED4	USB_OTG_ID: connect to ID pin of USB microAB connector
47	RESERVED3	USB_OTG_CPEN: driven by Z2 to enable 5V for USB device
49	RESERVED2	Rev-B: No Connect Rev-C: JTAG_BOOT_EN: Pull to 1.8V through 10K resistor on power-up to enable JTAG boot
51	W_DISABLE2_	40 MHz external clock input, 1.3V p-p max, designed as 50 ohm load. Contact Epiq Solutions for additional usage details.

Pin #	MiniPCIe Pin Name	Description as used in Sidekiq
2	+3.3Vaux	+3.3V supply
4	GND	Ground
6	1.5V	PL_GPIO3: Zynq pin F13, 1.8V I/O (in Epiq Solutions BSP, this is utilized as a UART0_RX signal by default)
8	UIM_PWR	DFU_BOOT_EN_N, hold low on power-up to enter DFU mode, internal pullup to 1.8V
10	UIM_DATA	JTAG TDI, 1.8V
12	UIM_CLK	JTAG TDO, 1.8V
14	UIM_RESET	JTAG TMS, 1.8V
16	UIM_SPU	JTAG TCK, 1.8V
18	GND	Ground
20	W_DISABLE1_	PS_GPIO2_IN: INPUT to Zynq pin D6 / MIO10, 3.3V tolerant, **inverted**
22	PERST_	PS_GPIO1_IN: INPUT to Zynq pin B10 / MIO11, 3.3V tolerant, **inverted**
24	+3.3Vaux	+3.3V supply
26	GND	Ground
28	1.5V	PL_GPIO2: Zynq pin F14, 1.8V I/O (in Epiq Solutions BSP reference design, this is utilized as an **inverted** UART0_TX signal by default)
30	SMB_CLK	I2C SCL 3.3V (without pullup resistor)
32	SMB_DATA	I2C SDA 3.3V (without pullup resistor)
34	GND	Ground
36	USB_D-	USB 2.0 high speed data, negative leg
38	USB_D+	USB 2.0 high speed data, positive leg
40	GND	Ground
42	LED_WWAN_	PS_GPIO3_OUT: Open drain output driven by Zynq pin B5 / MIO9 **inverted**
44	LED_WLAN_	PS_GPIO4_OUT: Open drain output driven by Zynq pin C13 / MIO53 **inverted**
46	LED_WPAN_	PS_GPIO5_OUT: Open drain output driven by Zynq pin D13 / MIO49 **inverted**
48	1.5V	PL_GPIO1: Zynq pin F15, 1.8V I/O
50	GND	Ground
52	+3.3Vaux	3.3V supply

Table 3: Sidekiq Z2 MiniPCIe edge connector signal descriptions

9 Linux Board Support Package Options for Sidekiq Z2

9.1 Overview

The Sidekiq Z2 board support package (BSP) consists of a collection of software & FPGA components needed to allow Sidekiq Z2 to boot Linux and execute a user's radio applications. These components include the uboot bootloader, Linux kernel, FPGA reference design, and other ancillary components such as the root filesystem and Linux device tree. There are two different BSP options available for Sidekiq Z2:

- **Analog Devices' BSP:** Open source (GPL) BSP based on Analog Devices' IIO framework and FPGA reference design, supported by Analog Devices (installed on the two Sidekiq Z2 units delivered in the EVK)
- **Epiq Solutions' BSP:** Commercial BSP based on Epiq Solutions' libsidekiq API and optimized FPGA reference design, supported by Epiq Solutions (installed on the two Sidekiq Z2 units delivered when a PDK upgrade is ordered)

The following section outlines some of the noteworthy differences between the two BSP options.

9.2 Sidekiq Z2 EVK vs Sidekiq Z2 PDK

The Sidekiq Z2 EVK includes two Z2 cards pre-loaded with Analog Devices' BSP supporting their IIO framework and FPGA reference design, whereas the Sidekiq Z2 PDK upgrade loads Epiq Solutions' BSP supporting libsidekiq and the standard Sidekiq FPGA reference design on the two Z2 cards. The hardware is otherwise identical between the EVK and the PDK.

9.3 Technical Support

All technical support for Sidekiq Z2 EVK customers and the corresponding Analog Devices' BSP is provided through Analog Devices' publicly accessible EngineerZone support forum [3]. Customers procuring the Sidekiq Z2 PDK upgrade receive technical support for Epiq Solutions' BSP utilizing the libsidekiq API and optimized FPGA reference design through a private web forum hosted by Epiq Solutions [2], with the support coming directly from Epiq's engineering team. Registration is required prior to accessing a customer's private support forum.

9.4 BSP Build Process

Analog Devices' BSP can be built for Sidekiq Z2 by following the instructions outlined in [4]. Epiq Solutions' BSP follows a similar procedure, with additional steps to integrate and build the Sidekiq FPGA reference design source code. Epiq Solutions' BSP also includes various test applications that utilize libsidekiq to serve as examples for using the API.

9.5 FPGA Resource Availability

Epiq Solutions' BSP includes an FPGA reference design that is optimized for resource efficiency to maximize the FPGA resources available to support the inclusion of custom processing blocks. A comparison of the FPGA resource utilization is shown below.

	Logic Cells	Block RAM	DSP48 Blocks
Total resources in Zynq 7010	28K	2.1 Mbits	80
% utilized in Analog Devices' stock FPGA reference design	40%	3%	90%
% utilized in Epiq Solutions' stock FPGA reference design	34%	18%*	0%

* Optional buffering included to support sustained high-rate Rx and Tx streaming between CPU and FPGA fabric in Zynq; can be reduced depending on CPU ↔ FPGA streaming requirements

Table 4: Analog Devices & Epiq Solutions FPGA Reference Design Resources

9.6 FPGA Feature Set and Customization

Epiq Solutions' BSP includes the Sidekiq FPGA reference design as its basis, which allows customers to migrate their designs from existing Sidekiq cards to Sidekiq Z2. Further, the Sidekiq FPGA reference design implements some key features not available in Analog Devices' FPGA reference design:

- **Timestamp management to support time-tagged receive and on-time transmit:** This allows a customer's application to accurately time tag each block of baseband I/Q data that is received so that all upstream processing blocks have a precise sense of time. This also provides a means to keep track of overflow / underflow conditions if the CPU can't keep up. Additionally, on-time transmit is supported to ensure that baseband I/Q data is upconverted and transmitted out at RF at precisely the requested time. This capability is key to enabling any sort of slotted time division multiple access (TDMA) communications system such as those found in cellular communications systems.
- **Pulse Per Second (PPS) integration:** This allows an externally generated PPS signal to be utilized by Sidekiq Z2 for the purpose of keeping track of time. This also provides a means to synchronize the time between physically separated Sidekiq Z2 units, where each Z2 unit receives a PPS signal.
- **Flexible buffering between the CPU (PS) and FPGA (PL):** In order to ensure sustained high transport streaming between the PL and the PS, it is necessary to include the appropriate amount of buffering to ensure no samples are dropped. Epiq Solutions' FPGA reference design allows the buffering scheme to be optimized to support the customer's requirements.
- **Well-defined “user app” space in the FPGA reference design to simplify custom IP block integration:** The standard Sidekiq FPGA reference design is architected to easily allow customers to integrate their own IP processing blocks into the signal processing chain with minimal effort. This is the same architecture used on all existing Sidekiq cards.

10 Basic Sidekiq Z2 Usage in a Carrier Card

10.1 Overview

Sidekiq Z2 includes a completely integrated Linux computer that controls the operation of the card. A host carrier system is only required to provide +3.3V via the MiniPCIe edge connector, and Sidekiq Z2 will power up, boot Linux, and remain in an idle state. From here, additional Linux userspace applications can be executed, and interfacing to the host carrier card via serial, USB, or GPIO can take place.

The following section outlines basic functionality of Sidekiq Z2 applicable to any host carrier system, with specific examples referenced back to the simple carrier card delivered as part of the Sidekiq Z2 Evaluation Kit (EVK) and Platform Development Kit (PDK). The same hardware consisting of the Sidekiq Z2 plus simple carrier card is used for both the EVK as well as the PDK, with the primary difference being the board support package of software/FPGA components utilized on Z2 (as outlined in section 9). The simple carrier card provides user accessible interfaces for Sidekiq Z2, such as SMA connectors for RF, a USB type C connector to access the USB 2.0 OTG port, a power supply input connector accepting DC voltage input between 6V and 17V, and additional signals (JTAG, serial console, and GPIO) on low profile headers.

10.2 Overview of Sidekiq Z2 & Simple Carrier Card (SCC)

Both the EVK and PDK are delivered with a simple carrier card that is capable of providing power and user accessible I/O to Sidekiq Z2. The simple carrier card and its associated I/O is shown in Figure 3, along with the Sidekiq Z2 card.

The following section describes all of the I/O associated with the simple carrier card.

10.2.1 USB-C Connector

The USB-C connector exposes a USB 2.0 compliant OTG interface to the USB 2.0 OTG pins available on Sidekiq Z2. By default, Sidekiq Z2 acts as a USB 2.0 device, and can accept 5V power from the USB host to power up the Z2 and simple carrier card, in addition to the standard D+/D- USB 2.0 signals. In this mode, no additional power source is required for Sidekiq Z2 (unless the total power consumption of the system exceeds 2.5W). When Sidekiq Z2 is configured as a USB OTG host (by using the proper USB-C OTG cable), then this USB-C connector can host USB devices such as Wi-Fi dongles, mass storage devices, USB-to-ethernet adapters, and more. In this mode of operation, power must be provided through the DC Power Input barrel jack.

10.2.2 DC Power Input

The DC Power Input barrel jack is an optional connector used to provide power to the Sidekiq Z2 and simple carrier card for use cases where an external power supply is desired or required (such as when using the USB-C port as an OTG host). The part number for this DC power input barrel jack is PJ-038-SMT manufactured by CUI, Inc, with the center pin carrying DC voltage and the ring serving as ground. The mating connector is part number PP-012, also manufactured by CUI, Inc. The acceptable DC voltage input range is between 6V and 17V DC. If the DC Power Input jack is provided with a voltage ≥ 6V DC, the unit will automatically begin drawing power on this connector instead of over USB.

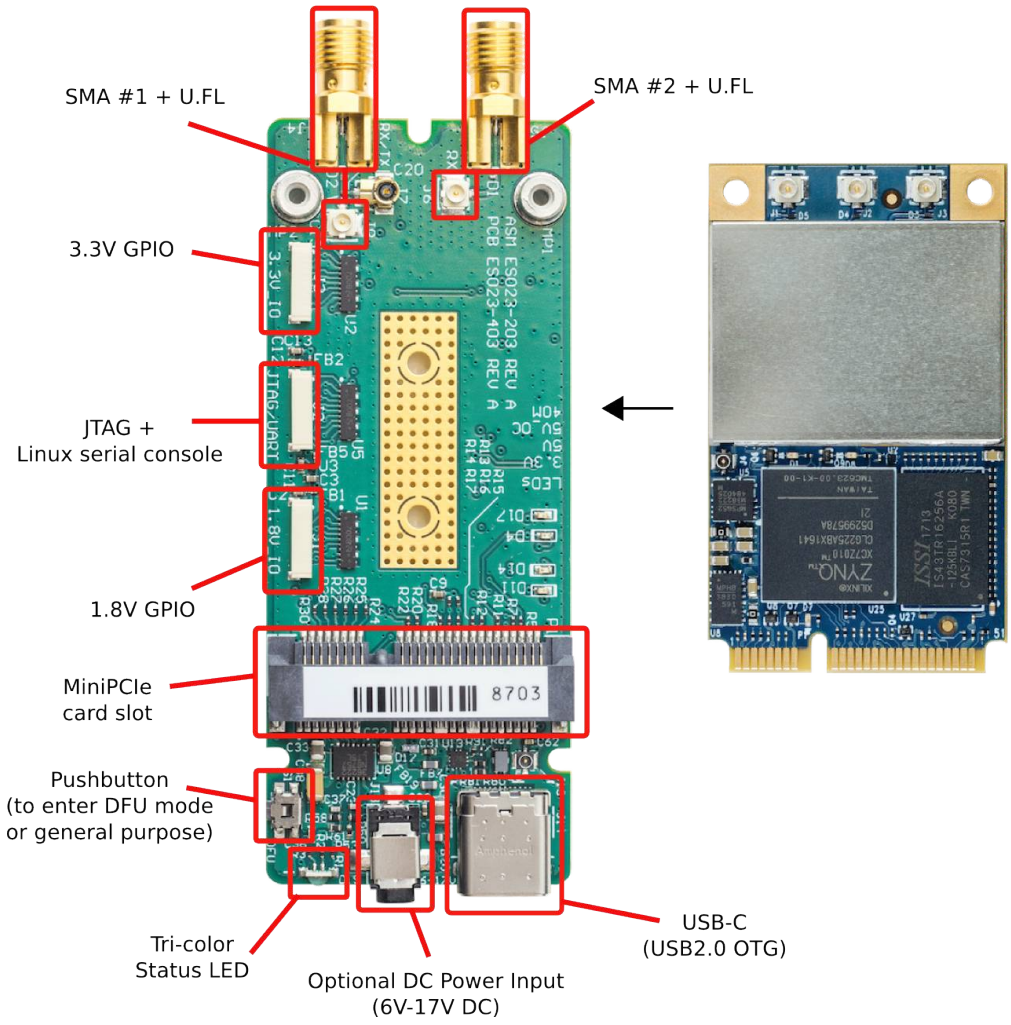


Figure 3: Sidekiq Z2 Simple Carrier Card (SCC) I/O

10.2.3 Tri-Color Status LED

The Tri-Color Status LED is a digital signal controlled from the Zynq PS pins (PS_GPIO3_N, PS_GPIO4_N, and PS_GPIO5_N) to allow a user application to provide system level status. These same signals are also routed to the 3.3V GPIO connector. This tri-color LED provides red, green, and yellow output.

10.2.4 Pushbutton

The pushbutton provides a normally open momentary switch that can be used to close a general purpose input pin routed to the Zynq PS. If the pushbutton is held during the Sidekiq Z2 power up sequence, the device will enter into Device Firmware Update (DFU) mode and enumerate with a DFU profile. This is a special mode that allows a user to perform a system update, replacing the Linux kernel, bootloader, and root filesystem. See section 10.8.2 for more details on using this update

method. If Sidekiq Z2 boots normally (i.e., not in DFU mode), this pushbutton switch can be used as an input device during normal operation, and is accessible on the Zynq SoC at PS_MIO48_501.

10.2.5 MiniPCIe Card Slot

The MiniPCIe card slot is the electrical/mechanical interface where Sidekiq Z2 is installed. See section 8.10 for the complete pinout of this interface.

10.2.6 1.8V GPIO Connector

The 1.8V GPIO Connector provides access to various GPIO signals coming from both the PS as well as the PL section of the Zynq SoC. The part number for the 10-pin connector on the simple carrier card is SM10B-XSRS-ETB from manufacturer JST. Cables are included to interface this 1.8V GPIO connector to the I/O expansion card. The cables included with the EVK and PDK that interface to this connector are part number 10XSRXSR36L100, a 100 mm length cable from JST. The pinout for this connector is shown in Table 5 below.

SCC	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6	Pin 7	Pin 8	Pin 9	Pin 10
Rev-B	V1P8_IO	GND	PL_GPIO1	PL_GPIO2	GND	PL_GPIO3	PS_GPIO6	GND	PS_GPIO7	GND
Rev-A										

Table 5: 1.8V GPIO signal connector

10.2.7 JTAG/Linux Serial Console Connector

The JTAG/Linux Serial Console Connector provides access to the Zynq JTAG signals exposed on the Sidekiq Z2 MiniPCIe edge connector, as well as the Linux serial console exposed on the Sidekiq Z2 MiniPCIe edge connector. The serial console uses 1.8V digital logic coming off the simple carrier card. The part number for the 10-pin connector on the simple carrier card is SM10B-XSRS-ETB from manufacturer JST. Cables are included to interface this JTAG/Linux serial console connector to the I/O expansion card. The cables included with the EVK and PDK that interface to this connector are part number 10XSRXSR36L100, a 100 mm length cable from JST. The pinout for this connector is shown in Table 6 below.

SCC	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6	Pin 7	Pin 8	Pin 9	Pin 10
Rev-B	V1P8_JTAG	GND	JTAG_TCK	GND	JTAG_TMS	JTAG_TDO	JTAG_TDI	JTAG_BOOT_EN	UART_RX	UART_TX
Rev-A								GND		

Table 6: JTAG/Linux serial console connector

10.2.8 3.3V GPIO Connector

The 3.3V GPIO Connector provides access to various GPIO signals coming from the PS section of the Zynq SoC. The part number for the 10-pin connector on the simple carrier card is SM10B-XSRS-ETB from manufacturer JST. Cables are included to interface this 3.3V GPIO connector to the I/O expansion card. The cables included with the EVK and PDK that interface to this connector are part number 10XSRXSR36L100, a 100 mm length cable from JST. The pinout for this connector is shown in Table 7 below.

SCC	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6	Pin 7	Pin 8	Pin 9	Pin 10
Rev-B	V3P3_IO	PS_GPIO1	PS_GPIO2	PS_GPIO3_N	PS_GPIO4_N	PS_GPIO5_N	GND	SMB_DATA	SMB_CLK	GND
Rev-A	PS_GPIO1	PS_GPIO2	GND	PS_GPIO3_N	PS_GPIO4_N	PS_GPIO5_N	GND	SMB_DATA	SMB_CLK	GND

Table 7: 3.3V GPIO signal connector

10.2.9 SMA #1 + U.FL

The SMA #1 + U.FL connector provides an externally accessible RF interface in an SMA connector that can then be connected to Sidekiq Z2 via a U.FL-to-U.FL cable. The U.FL port connected to SMA #1 can be cabled in to any of the three U.FL ports available on Sidekiq Z2, depending on the customer's use case.

10.2.10 SMA #2 + U.FL

The SMA #2 + U.FL connector provides an externally accessible RF interface in an SMA connector that can then be connected to Sidekiq Z2 via a U.FL-to-U.FL cable. The U.FL port connected to SMA #2 can be cabled in to any of the three U.FL ports available on Sidekiq Z2, depending on the customer's use case.

10.3 Powering Up

To initially power up Sidekiq Z2 in the simple carrier card, a host PC can be used to provide both power and USB connectivity to the device through a standard USB-C cable. In the examples that follow, it is assumed that a Linux PC is being used as the host (running Ubuntu 16.04, though other Linux distributions should work as well). It is also possible to use Windows and MacOS as hosts. Contact Epiq Solutions for details.

Insert the USB-C cable to connect the host PC to Sidekiq Z2 simple carrier card, and the Sidekiq Z2 card should power up automatically as indicated by the green Power Good indicator LED on Sidekiq Z2. In addition, the Heartbeat LED indicator on Sidekiq Z2 should also begin blinking after several seconds. See Figure 4 for details of this setup.

Alternately, the DC wallwart included with the Sidekiq Z2 EVK/PDK can be used to power the Sidekiq Z2 and simple carrier card. This power adapter is typically used if a connection to a USB host is not utilized, and thus an alternate power source is required.

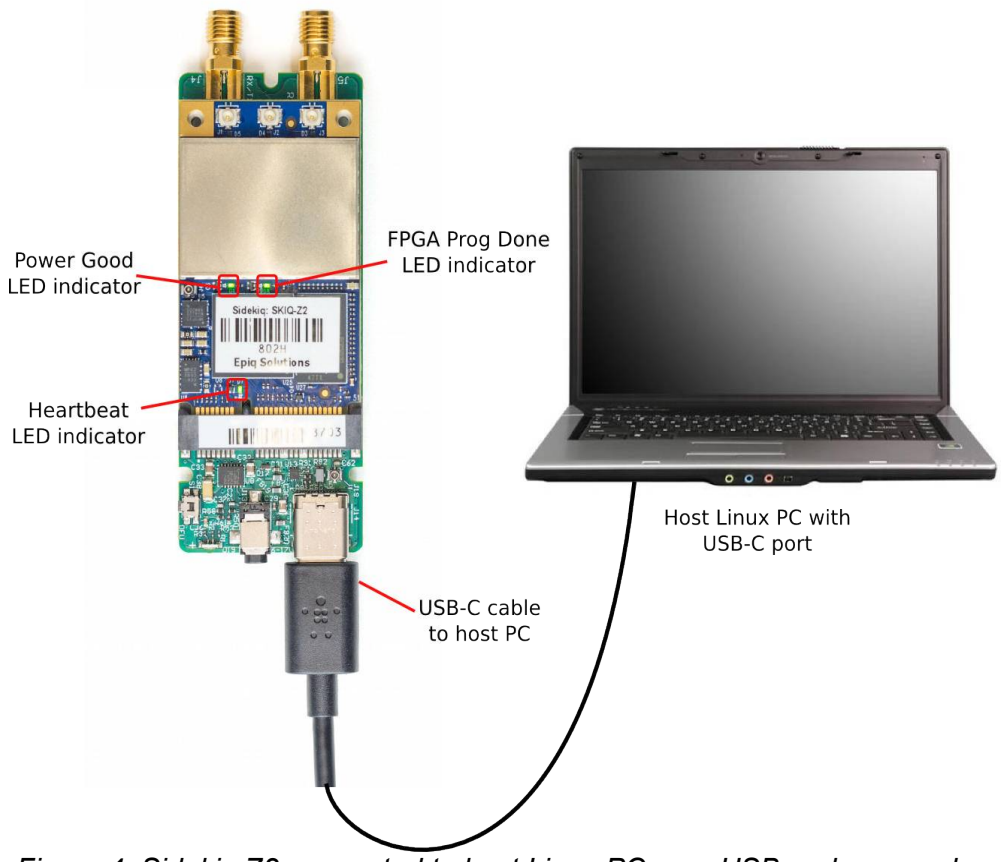


Figure 4: Sidekiq Z2 connected to host Linux PC over USB and powered up

10.4 Interfacing to Sidekiq Z2 over USB

Once Sidekiq Z2 is powered up and the Linux kernel has booted, the device appears as a dual-purpose USB device consisting of a USB mass storage device, USB ethernet gadget network device (supporting the RNDIS protocol), a serial console (supporting the CDC-ACM profile), and an IIO data interface. The USB mass storage device should be automatically mounted on the host computer, allowing the contents of the mass storage device to be accessed by the host computer. A text file called config.txt containing network configuration information for the USB ethernet gadget interface is accessible on this mass storage device. This configuration file is used to set the default IP address of the USB ethernet gadget interface on Sidekiq Z2 (annotated as “ipaddr”, with a default value of 192.168.3.1), as well as the host PC's IP address for communicating with this USB device (annotated as “ipaddr_host”, with a default value of 192.168.3.9). The IP address configuration for either the host or Sidekiq Z2 can be updated by editing the config.txt configuration file, saving the file changes, properly ejecting the USB mass storage device, and then removing power to the Sidekiq Z2 and carrier card by removing the USB cable. Upon reinsertion of the USB cable, Sidekiq Z2 will power back up and the new IP address configuration will be utilized.

An example of the contents of the config.txt file is shown below.

```
# Device Configuration File
# Edit, Save and then Eject the USB Drive

[NETWORK]
hostname = z2
ipaddr = 192.168.3.1
ipaddr_host = 192.168.3.9
netmask = 255.255.255.0

[WLAN]
ssid_wlan =
pwd_wlan =
ipaddr_wlan =

[SYSTEM]
xo_correction =
udc_handle_suspend = 0

[ACTIONS]
diagnostic_report = 0
dfu = 0
reset = 0
calibrate = 0
```

At this point, the user can login to Sidekiq Z2 over ssh using a standard terminal application on the host Linux PC. The default username is **root,** and the default password is dependent on which BSP is loaded on to Sidekiq Z2. The Analog Devices' BSP has the default root password set as **analog**, while Epiq Solutions' BSP has the default root password set as **epiq**. An example screenshot of establishing an ssh session with a Sidekiq Z2 loaded with Epiq Solutions BSP is shown in Figure 5, connecting from a host Linux PC over the USB ethernet gadget interface.

Figure 5: Example ssh session from a host Linux PC to Sidekiq Z2 over USB

The user is now logged in to the Linux computer on Sidekiq Z2, and can interact with it like any standard Linux system.

10.5 Accessing GPIO, JTAG, and the Linux Console Serial Port

The 1.8V GPIO, 3.3V GPIO, JTAG port, and Linux serial console port signals are all made available on the three edge connectors on the simple carrier card, as shown in Figure 3. These signals can be accessed through more user-friendly ports through the I/O expansion card provided with the Sidekiq Z2 EVK/PDK. This I/O expansion card provides 0.1” headers for the various GPIO pins, a standard Digilent-compatible 2x7 JTAG header interface for accessing the Zynq SoC, and a USB-C connector through which a USB-to-UART circuit provides access to the Linux serial console. Access to the Linux serial console via the USB-C connector on the I/O expansion card requires the user to connect a USB-C cable between the I/O expansion card and a host PC running a serial terminal emulator program such as minicom. The I/O expansion card will enumerate over USB as a USB serial port (typically at /dev/ttyUSB0 or similar on the host Linux system). The default Linux serial console on Sidekiq Z2 runs at a baud rate of 115.2 kbps, with 8 data bits, one stop bit, and no parity bits. The default username/password of root/epiq can be used to log in to the system for the Epiq BSP, and root/analog can be used for the Analog Devices' BSP.

When using this configuration, the three included interface cables connect the simple carrier card to the I/O expansion card, as shown in Figure 6. These interface cables must be connected prior to powering up the system.

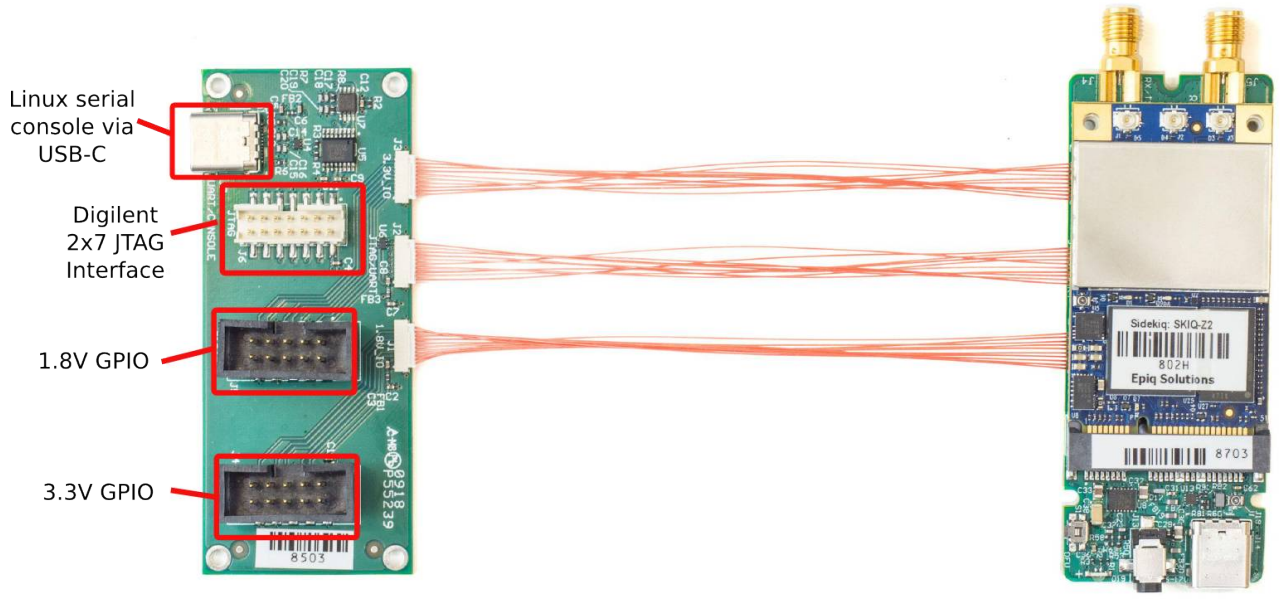


Figure 6: I/O Expander Card for simplified access to I/O on the Simple Carrier Card

10.5.1 3.3V GPIO

The 3.3V GPIO signals are accessible on the I/O expansion card via a 2x5 shrouded Amphenol connector (part number 72454-101LF). The pinout for this connector is as follows:

SCC	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6	Pin 7	Pin 8	Pin 9	Pin 10
Rev-B	V3P3_IO	PS_GPIO1	PS_GPIO2	PS_GPIO3_N	PS_GPIO4_N	PS_GPIO5_N	GND	SMB_DATA	SMB_CLK	GND
Rev-A	PS_GPIO1	PS_GPIO2	GND	PS_GPIO3_N	PS_GPIO4_N	PS_GPIO5_N	GND	SMB_DATA	SMB_CLK	GND

Table 8: 3.3V GPIO pinout on I/O expansion board

10.5.2 1.8V GPIO

The 1.8V GPIO signals are accessible on the I/O expansion card via a 2x5 shrouded Amphenol connector (part number 72454-101LF). The pinout for this connector is as follows:

IO Exp	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6	Pin 7	Pin 8	Pin 9	Pin 10
Rev-B	V1P8_IO	GND	PL_GPIO1	PL_GPIO2	GND	PL_GPIO3	PS_GPIO6	GND	PS_GPIO7	GND
Rev-A										

Table 9: 1.8V GPIO pinout on I/O expansion board

10.5.3 JTAG

In addition to Digilent-compatible 2x7 JTAG header interface connector found on Rev-A of the I/O expander card, the Rev-B version includes a jumper situated between JTAG header and the 1.8V GPIO header. This header provides user access to JTAG_BOOT_EN. See Table 3 Pin-49 and Table 6 Pin-8.

10.6 Controlling GPIO

Access to the GPIO is supported through the sysfs entries in Linux, located at **/sys/class/gpio/**. Details on interfacing with the GPIO via sysfs is described in detail in [5]. All GPIO definitions for the Sidekiq Z2 begin at an offset of 906. The MIO GPIO are under direct control of the PS, whose mapping is outlined in Table 10. PL_GPIO are controllable via the FPGA and control via the sysfs can be enabled. In order to allow control of the PL GPIO via the Linux userspace, the pin must be mapped to an EMIO offset, as outlined in [6]. From that point, the Linux sysfs offset begins at 906+54=960. For example, for accessing a GPIO mapped to EMIO0, the Linux sysfs value used should be 906 + 54 + 0 = 960.

GPIO Name	Zynq GPIO Name	Linux sysfs value
PS_GPIO1	MIO11	906 + 11 = 917
PS_GPIO2	MIO10	906 + 10 = 916
PS_GPIO3	MIO9	906 + 9 = 915
PS_GPIO4	MIO53	906 + 53 = 959
PS_GPIO5	MIO49	906 + 49 = 955
PS_GPIO6	MIO14	906 + 14 = 920
PS_GPIO7	MIO0	906 + 0 = 906

Table 10: PS GPIO Mapping

10.6.1 Linux sysfs GPIO Control Example

The following steps outline how to configure PS_GPIO6 as an output and configure the value to a logical high.

1. Export the GPIO (906 + 14 = 920)

```
# echo 920 > /sys/class/gpio/export
```

2. Verify the sysfs entry for 920 now exists

```
# ls /sys/class/gpio/
export       gpio920      gpiochip906  unexport
```

3. Configure the GPIO as an output

```
# echo out > /sys/class/gpio/gpio920/direction
```

4. Verify the GPIO is configured as an output

```
# cat /sys/class/gpio/gpio920/direction
out
```

5. Configure the GPIO as a logical 1

```
# echo 1 > /sys/class/gpio/gpio920/value
```

6. Verify the GPIO value is set to 1

```
# cat /sys/class/gpio/gpio920/value
```

```
1
```

10.7 Default Memory Partition and Filesystem Configuration

By default, Sidekiq Z2 contains a 32 MB QSPI flash memory device which is used to hold the bootloader, Linux kernel, root filesystem, and additional non-volatile storage for user applications. The root filesystem is configured as a compressed ROM file system (cramfs) which is stored in the QSPI memory part, and decompressed into RAM at power up. Thus, the contents of the root filesystem should be considered volatile, and can only be updated during a DFU system update (as outlined in section 10.8).

The default memory partition for the Analog Devices' BSP is shown in Table 11 below.

Address Range	Size	Name	Additional Details
0x000000000000-0x000000100000	1M	qspi-fsbl-uboot	Bootloader
0x000000100000-0x000000120000	128k	qspi-uboot-env	u-boot environment variables
0x000000120000-0x0000001E0000	1M	qspi-nvmfs	
0x000000200000-0x000001E00000	30M	qspi-linux	Linux Filesystem

Table 11: Default memory partition table for Analog Devices' BSP (shipped with EVK)

The default memory partition for the Epiq Solutions' BSP is shown in Table 12 below.

Address Range	Size	Name	Additional Details
0x000000000000-0x000000100000	1M	qspi-fsbl-uboot	Bootloader
0x000000100000-0x000000120000	128k	qspi-uboot-env	u-boot environment variables
0x000000120000-0x000000130000	64k	qspi-versions	Partition for read-only version info
0x000000200000-0x000000F00000	15M	qspi-linux	Linux root filesystem
0x000001100000-0x000001400000	3M	qspi-app-ro	Read-only application partition
0x000001400000-0x000002000000	12M	qspi-app-rw	Read-write application partition

Table 12: Default partition table for Epiq Solutions' BSP (shipped with PDK upgrade)

Given the limited non-volatile storage on Sidekiq Z2, it is highly recommended to store compressed executables whenever possible on the device. Utilities such as **xz** are capable of achieving substantial savings when used to compress a binary executable. These executables can then be decompressed into RAM and executed there, where there is substantially more space (512 MB).

Additional non-volatile storage space can be added in to the system through an external USB mass storage device connected to the USB OTG port on Sidekiq Z2. The USB OTG port must be configured to operate in host mode in order to be able to enumerate a USB mass storage device.

10.8 Using Buildroot to Update uBoot, Linux Kernel, and Root Filesystem

Both the IIO-based Sidekiq Z2 as well as the libsidekiq-based Sidekiq Z2 root filesystem, Linux kernel, and u-boot can be modified, rebuilt, and updated by utilizing the DFU mode.

10.8.1 Building

Sidekiq Z2 EVK refers to https://github.com/epiqsolutions/plutosdr-fw
Sidekiq Z2 PDK refers to the Sidekiq Z2 Software Developers Manual found on [2].

10.8.2 Updating

The Sidekiq Z2 must be placed into DFU mode and then dfu-util [7] can be utilized to program the build artifacts from a host Linux computer.

Entering DFU

The Sidekiq Z2 can be placed into DFU mode from Linux, u-boot, or via the DFU button.

- When logged into Z2 via SSH or serial console session, the command **device_reboot sf** can be executed.
- From Z2 serial console u-boot, the command **run dfu_sf** can be executed.
- Pressing and holding the DFU pushbutton prior to power up will cause Sidekiq Z2 to automatically enter DFU mode. Contact Epiq Solutions for more details on accessing this mode.

Confirming DFU Mode

To confirm that the Sidekiq Z2 is in DFU mode, lsusb can be executed from a host Linux computer. An entry similar to the bolded entry should be observed.

```
$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 004: ID 8087:0a2b Intel Corp.
Bus 001 Device 003: ID 046d:c408 Logitech, Inc. Marble Mouse (4-button)
Bus 001 Device 002: ID 2516:0011
Bus 001 Device 007: ID 2fa2:5a02
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Additionally, dfu-util can be used to list out all DFU capable devices connected to the host system via **dfu-util -l**. NOTE: it is possible that USB permissions for the Sidekiq Z2 DFU may not have been configured. As a result, dfu-util may need to be ran with sudo. An example of the expected output is shown below.

```
$ dfu-util -l
dfu-util 0.9

Copyright 2005-2009 Weston Schmidt, Harald Welte and OpenMoko Inc.
```

```
Copyright 2010-2016 Tormod Volden and Stefan Schmidt
This program is Free Software and has ABSOLUTELY NO WARRANTY
Please report bugs to http://sourceforge.net/p/dfu-util/tickets/

Found DFU: [2fa2:5a02] ver=0221, devnum=7, cfg=1, intf=0, path="1-1", alt=4,
name="spare.dfu", serial="UNKNOWN"
Found DFU: [2fa2:5a02] ver=0221, devnum=7, cfg=1, intf=0, path="1-1", alt=3,
name="uboot-env.dfu", serial="UNKNOWN"
Found DFU: [2fa2:5a02] ver=0221, devnum=7, cfg=1, intf=0, path="1-1", alt=2,
name="uboot-extra-env.dfu", serial="UNKNOWN"
Found DFU: [2fa2:5a02] ver=0221, devnum=7, cfg=1, intf=0, path="1-1", alt=1,
name="firmware.dfu", serial="UNKNOWN"
Found DFU: [2fa2:5a02] ver=0221, devnum=7, cfg=1, intf=0, path="1-1", alt=0,
name="boot.dfu", serial="UNKNOWN"
```

Updating

The dfu-util application can be used to update the various components of the Sidekiq Z2. NOTE: it is possible that USB permissions for the Sidekiq Z2 DFU may not have been configured. As a result, dfu-util may need to be executed with the sudo command, that is, run prefixed by sudo (sudo dfu-util).

The u-boot image can be updated via:

dfu-util -a boot.dfu -D boot.dfu

The kernel and root filesystem can be updated via:

dfu-util -a firmware.dfu -D sidekiqz2.dfu

The u-boot environment variables can be updated via:

dfu-util -a uboot-env.dfu -D uboot-env.dfu

Once the update has completed, the Sidekiq Z2 needs to be power-cycled by disconnecting the USB cable from the Z2 simple card carrier along with the external power source (if connected). Reconnect USB cable and power source (if used) to power-up the Z2.

10.9 Power Consumption

The power consumption of Sidekiq Z2 varies depending on the configuration and application of the card. The following power measurements are provided for Sidekiq Z2 (as a standalone card) as well as the total system power consumption when installed in the simple carrier card. For these power consumption measurements, libsidekiq is utilized, along with the optimized FPGA reference design available with the Sidekiq Z2 PDK.

Configuration	Z2 Power Consumption (in Watts)	Z2 & Simple Carrier Card Power Consumption in Watts (9V Provided via Barrel Connector)	Z2 & Simple Carrier Card Power Consumption in Watts (5V Provided via USB-C connector)
Z2 booted up to Linux, sitting idle, no USB connection	0.96 W	1.16 W	1.14 W
Z2 booted up to Linux sitting idle, USB ethernet network connection between host computer and card, with the host computer ssh'ed in to Sidekiq Z2	0.96 W	1.16 W	1.13 W
Z2 booted up to Linux, USB ethernet network interface connection between host computer and card, Rx tuned to 850 MHz, running rx_samples application with a sample rate of 5 Msamples/sec storing the samples locally on Z2, on-board LNA disabled, no Tx	1.53 W	1.92 W	2.02 W
Z2 booted up to Linux, USB ethernet network interface connection between host computer and card, Rx tuned to 850 MHz, running rx_samples application with a sample rate of 5 Msamples/sec storing the samples locally on Z2, on-board LNA enabled, no Tx	1.83 W	2.13 W	2.23 W
Z2 idle, ssh connection open, AD936x not initialized	1.0 W		1.1 W
"rx_benchmark --rate 61400000" (61.4 MS/s)	2.2 W		2.5 W
"rx_benchmark --rate 15000000" (15 MS/s)	2.0 W		2.1 W
"rx_benchmark --rate 5000000" (5 MS/s)	1.7 W		1.9 W
"rx_benchmark --rate 1000000" (1 MS/s)	1.6 W		1.8 W
"tx_samples --rate 5e6 --attenuation 0 --block size=8188"	2.0 W		2.2 W
"tx_samples --rate 15e6 --attenuation 0 --block-size=8188"	2.2 W		2.4 W
"tx_samples --rate 25e6 --attenuation 0 --block-size=8188"	2.3 W		2.6 W
"tx_samples --rate 30.72e6 --attenuation 0 --block-size=8188"	2.4 W		2.8 W
"tx_samples --rate 50e6 --attenuation 0 --block-size=8188"	2.5 W		2.9 W
"tx_samples --rate 61.44e6 --attenuation 0 --block-size=8188"	2.7 W		3.1 W
"xcr_benchmark --rate 5e6"	1.5 W		1.7 W
"xcr_benchmark --rate 10e6"	1.8 W		2.0 W
"xcr_benchmark --rate 15e6"	1.9 W		2.1 W
"xcr_benchmark --rate 20e6"	2.0 W		2.1 W
"xcr_benchmark --rate 25e6"	2.0 W		2.2 W
"xcr_benchmark --rate 30.72e6"	2.0 W		2.2 W
"xcr_benchmark --rate 61.44e6"	2.1 W		2.4 W

Table 13: Example power consumption estimates for Sidekiq Z2

10.10 Thermal Dissipation

Effective use of Sidekiq Z2 in a system also requires consideration of an appropriate thermal dissipation solution. Since Sidekiq Z2 can be integrated into a variety of different host systems with different thermal profiles (i.e., forced air, natural convection, etc), the end user is required to perform their own system analysis to determine what level of thermal dissipation is appropriate for their use-case. Sidekiq uses components that are rated for operation to +85 deg C, and thus the end user must ensure that the temperature reported by the on-board temperature sensor does not exceed +85 deg C. **Exceeding the maximum rated temperature of +85 deg C may damage the Sidekiq card and/ or accelerate failure of the card.**

As discussed in section 8.8, both the RF shield as well as the FPGA are the two primary sources of heat requiring thermal dissipation. It is highly recommended that a thermal transfer solution, such as a thermal gap pad material [8], be used to provide a thermal dissipation path between the RF shield/FPGA and an external conduction surface in the host system. Note: The actual temperature range achievable in a given system may vary substantially depending on a number of factors, including the number of RF receivers operational, the A/D and D/A sample rates, customizations done to the FPGA, and others. Again, it is strongly recommended that a thorough system evaluation be performed by the customer to fully characterize the thermal profile of Sidekiq Z2 in their system.

10.11 Internal/External Reference Clock Options

Sidekiq Z2 supports options to use either an internal (i.e. on-board) 40 MHz TCVCXO as a reference clock, or an external 40 MHz reference clock as defined in section 8.7. Regardless of which clock source is selected, this clock serves as the reference for both the RF front end as well as the digital processing blocks in the FPGA. The selection of whether Sidekiq uses the internal reference clock or the external reference clock is stored as a configuration parameter in EEPROM on the card. This parameter is read at power up, and the clock source determination is then made.

If Sidekiq is configured to use an external reference clock, but no external reference clock is provided via the W.FL connector defined in section 8.7, any application attempting to initialize the Sidekiq card will fail.

For cases where a customer would like to switch between internal and external reference clock options, a software application can be provided to update the EEPROM configuration settings. Note that changing reference clock sources while running is not supported; the EEPROM configuration must be changed, and then the new reference clock source will be used at the next power up. Please contact Epiq Solutions for details [2].

11 Installing Sidekiq Z2 in an Alternate Host System

11.1 Overview

Any host system that provides a full size MiniPCIe slot with the standard 3.3V power rail required for MiniPCIe slots can serve as a host system for Sidekiq Z2. If integrating Sidekiq Z2 with a COTS computer system that has an available MiniPCIe slot, note that most of the I/O accessible via the MiniPCIe edge connector (such as the GPIO pins and the Linux serial console) won't be accessible by the host, since these interfaces are not normally available in a MiniPCIe card. The USB 2.0 interface provided by Sidekiq Z2 will typically be the primary interface used to communicate between the host system and Sidekiq Z2, so it is imperative that the host system provides this interface. Sidekiq Z2 does not use PCIe due to the fact that the package variant of the Zynq SoC used in Sidekiq Z2 does not support the required high-speed transceivers necessary to support PCIe.

As mentioned previously, it is critical to never install or remove Sidekiq Z2 in a host system when power is applied to the slot.

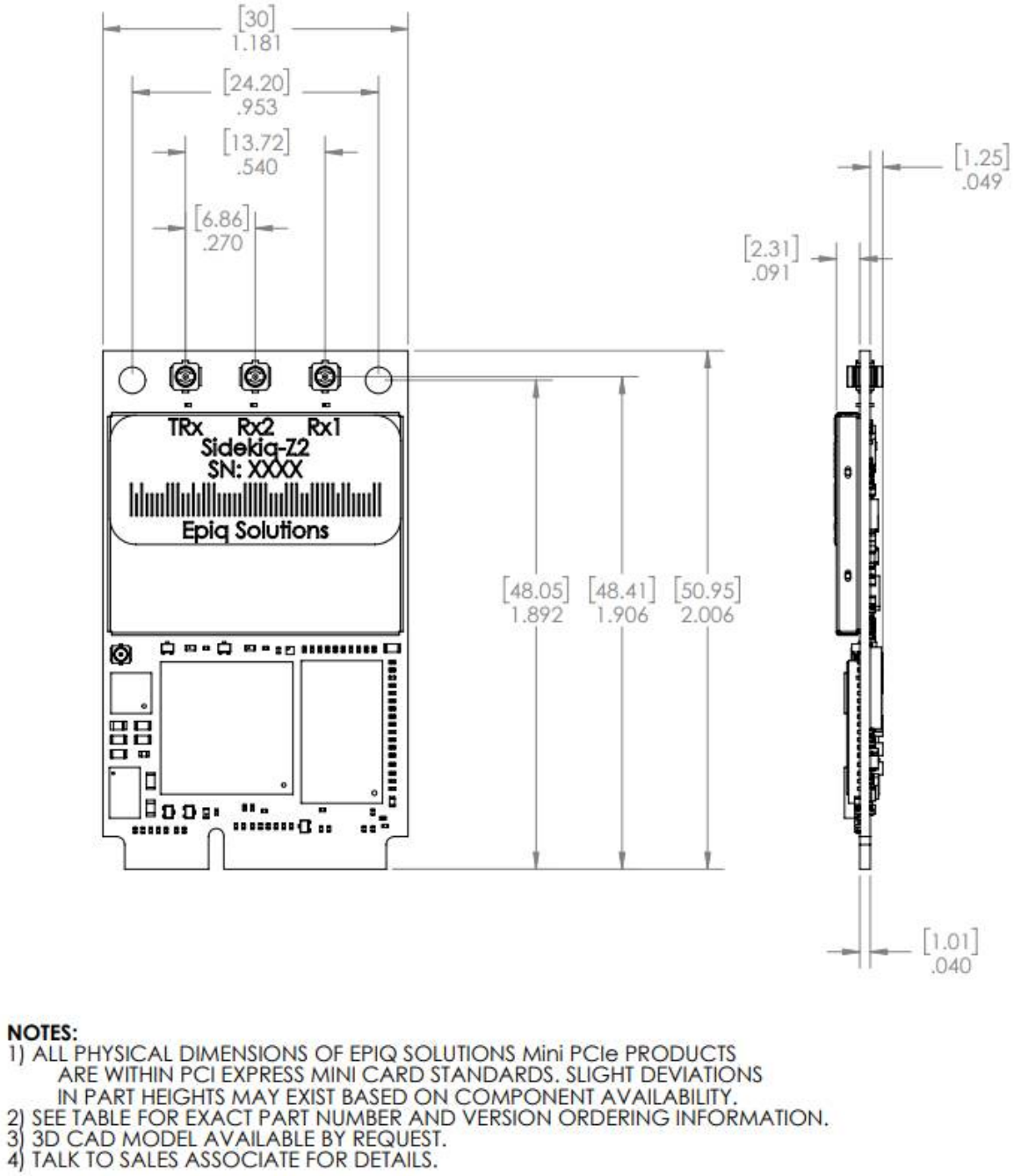
11.2 Developing a Custom Carrier Card for Sidekiq Z2

Sidekiq Z2 was architected to allow very simple carrier cards to be developed to provide the I/O and peripheral requirements of a broad range of customer use-cases. The simplest possible carrier card can provide nothing more than a MiniPCIe edge connector, mounting holes to mechanically secure the card, and a 3.3V DC input power supply. More complicated carrier cards can be developed to include additional USB devices such as USB-to-ethernet conversion ICs, microSD card storage for mass data logging, and more. Other peripherals such as GPS receivers and IMUs can also be included on a customized carrier card.

The schematics, bill of material, and gerber files for the simple carrier card outlined in section10 are available upon request from Epiq Solutions to simplify custom carrier card development. Please contact Epiq Solutions for details [2].

12 Sidekiq Z2 Mechanical Outline

A dimensioned mechanical drawing of Sidekiq Z2 is shown in Figure 6. In addition, a 3D model (in STP format) is also available. Please contact Epiq Solutions for this model.



NOTES:
1) ALL PHYSICAL DIMENSIONS OF EPIQ SOLUTIONS Mini PCIe PRODUCTS ARE WITHIN PCI EXPRESS MINI CARD STANDARDS. SLIGHT DEVIATIONS IN PART HEIGHTS MAY EXIST BASED ON COMPONENT AVAILABILITY.
2) SEE TABLE FOR EXACT PART NUMBER AND VERSION ORDERING INFORMATION.
3) 3D CAD MODEL AVAILABLE BY REQUEST.
4) TALK TO SALES ASSOCIATE FOR DETAILS.

Figure 7: Sidekiq Z2 dimensioned mechanical drawing

13 Statement of Volatility

Date: 7/10/2018

Certificate of Volatility					
Model: **Sidekiq Z2**	Part Number: **ES023-103**	Manufacturer: **Epiq Solutions**			
		Street Address: **165 Commerce Drive Suite #204**			
		City: **Schaumburg**	State: **IL**	Zip: **60173**	
Volatile Memory					
Does the item contain volatile memory (i.e., memory whose contents are lost when power is removed)? ☒ Yes ☐ No					
If the answer is 'Yes', please provide the following information for each type (use additional sheets if required):					
Type (SRAM, DRAM, etc.): **DDR3L RAM**	Size: **512 MB**	User Modifiable: ☒ Yes ☐ No	Function: **Memory for user application**	Process to Clear: **Remove power**	
Type (SRAM, DRAM, etc.): **FPGA**	Size: **512 MB**	User Modifiable: ☒ Yes ☐ No	Function: **Block RAM inside FPGA for application storage**	Process to Clear: **Remove power**	
Type (SRAM, DRAM, etc.):	Size:	User Modifiable: ☐ Yes ☐ No	Function:	Process to Clear:	
Non-Volatile Memory					
Does the item contain non-volatile memory (i.e., memory whose contents are retained when power is removed)? ☒ Yes ☐ No					
If the answer is 'Yes', please provide the following information for each type (use additional sheets if required):					
Type (BBRAM, Flash, EEPROM, etc.): **QSPI Flash**	Size: **32 MB**	User Modifiable: ☒ Yes ☐ No	Function: **Holds uboot bootloader, Linux kernel, and root filesystem**	Process to Clear: **Cleared with Linux utilities**	
Type (BBRAM, Flash, EEPROM, etc.): **EEPROM**	Size: **16 KB**	User Modifiable: ☒ Yes ☐ No	Function: **Holds system management configuration settings and parameters**	Process to Clear: **Cleared with software test application provided upon request**	
Type (BBRAM, Flash, EEPROM, etc.):	Size:	User Modifiable: ☐ Yes ☐ No	Function:	Process to Clear:	
Media					
Does the item contain media storage capability (i.e., removable or non-removable disk drives, tape drives, memory cards, etc.)? ☐ Yes ☒ No					
If the answer is 'Yes', please provide the following information for each type (use additional sheets if required):					
Type (Disk, Tape, etc.): Removable: ☐ Yes ☐ No	Size:	User Modifiable: ☐ Yes ☐ No	Function:	Process to Clear:	
Type (Disk, Tape, etc.): Removable: ☐ Yes ☐ No	Size:	User Modifiable: ☐ Yes ☐ No	Function:	Process to Clear:	
Type (Disk, Tape, etc.): Removable: ☐ Yes ☐ No	Size:	User Modifiable: ☐ Yes ☐ No	Function:	Process to Clear:	
Additional Information: **Additional non-volatile storage space can be optionally added in to the system through an external USB mass storage device (not provided) connected to the USB OTG port on Sidekiq Z2.**					

Figure 8: Sidekiq Z2 statement of volatility