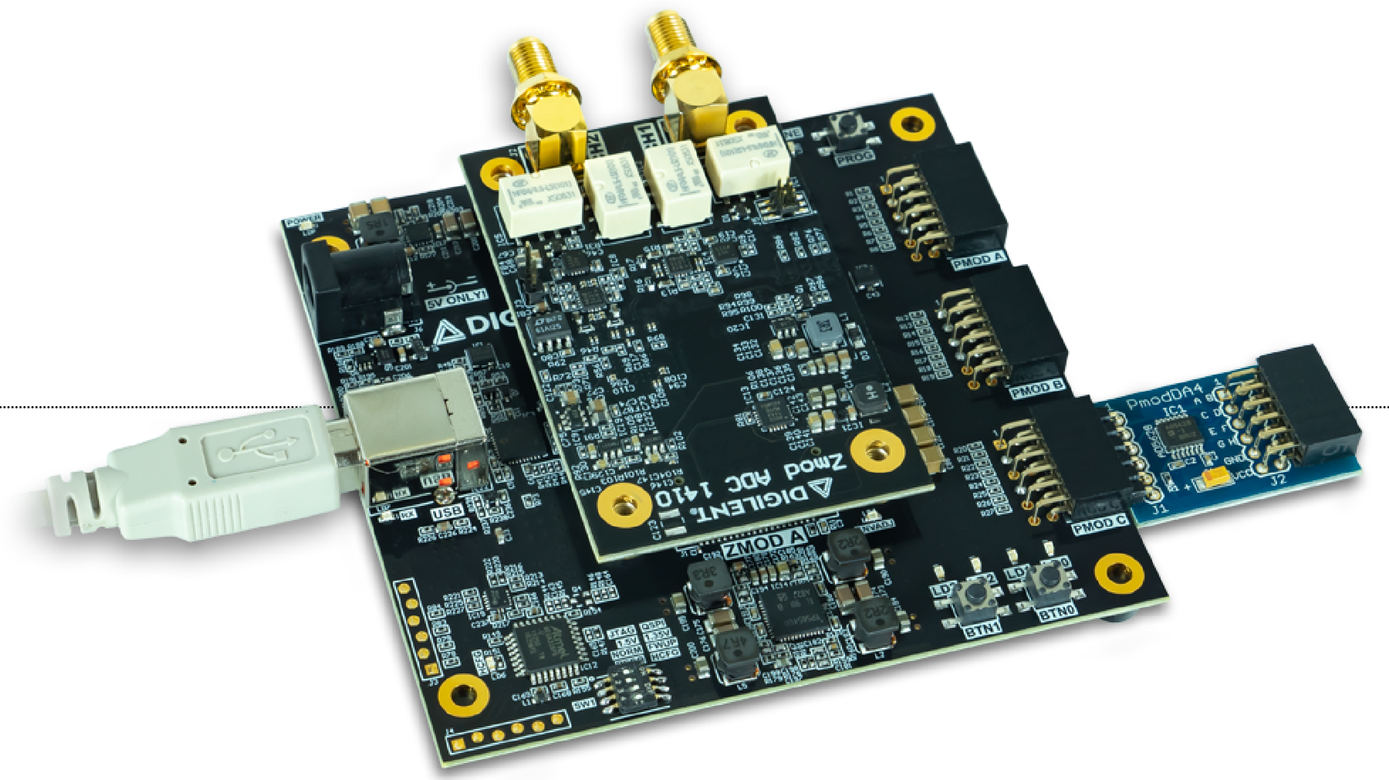
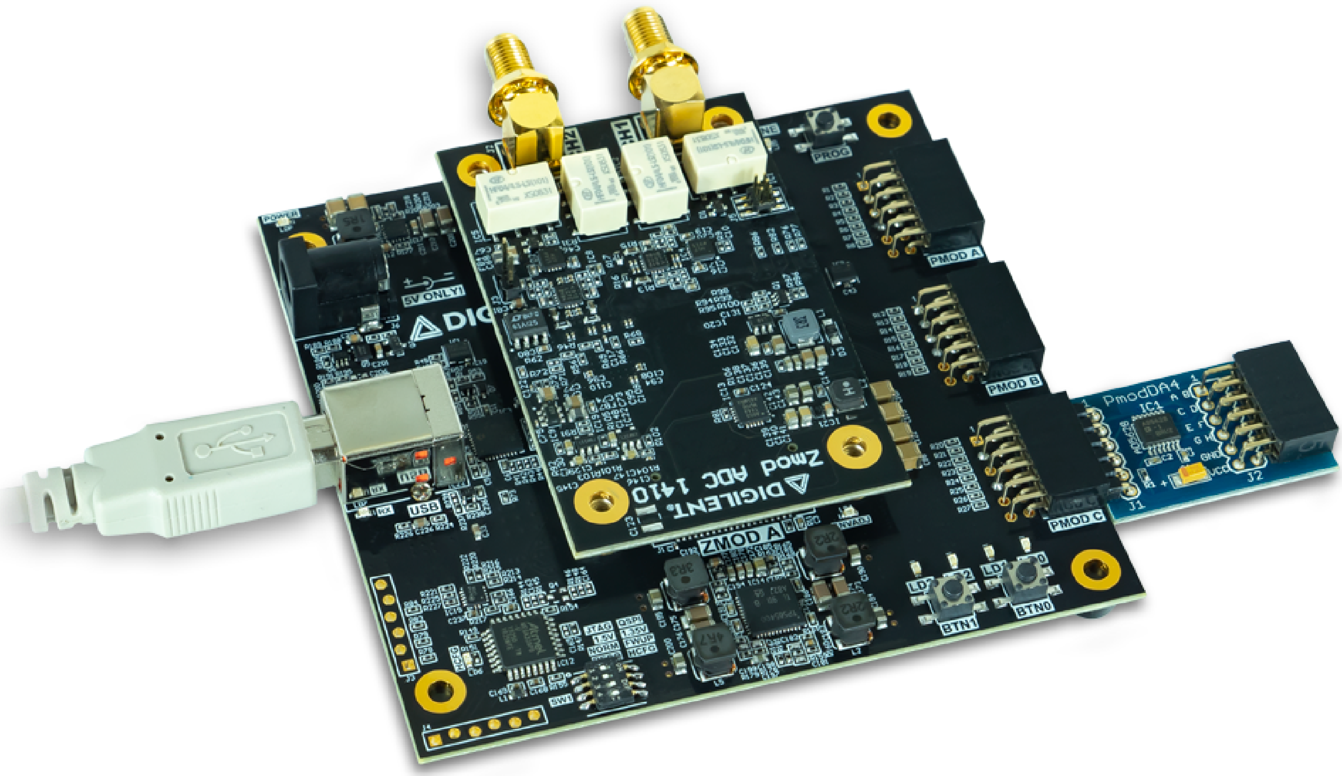
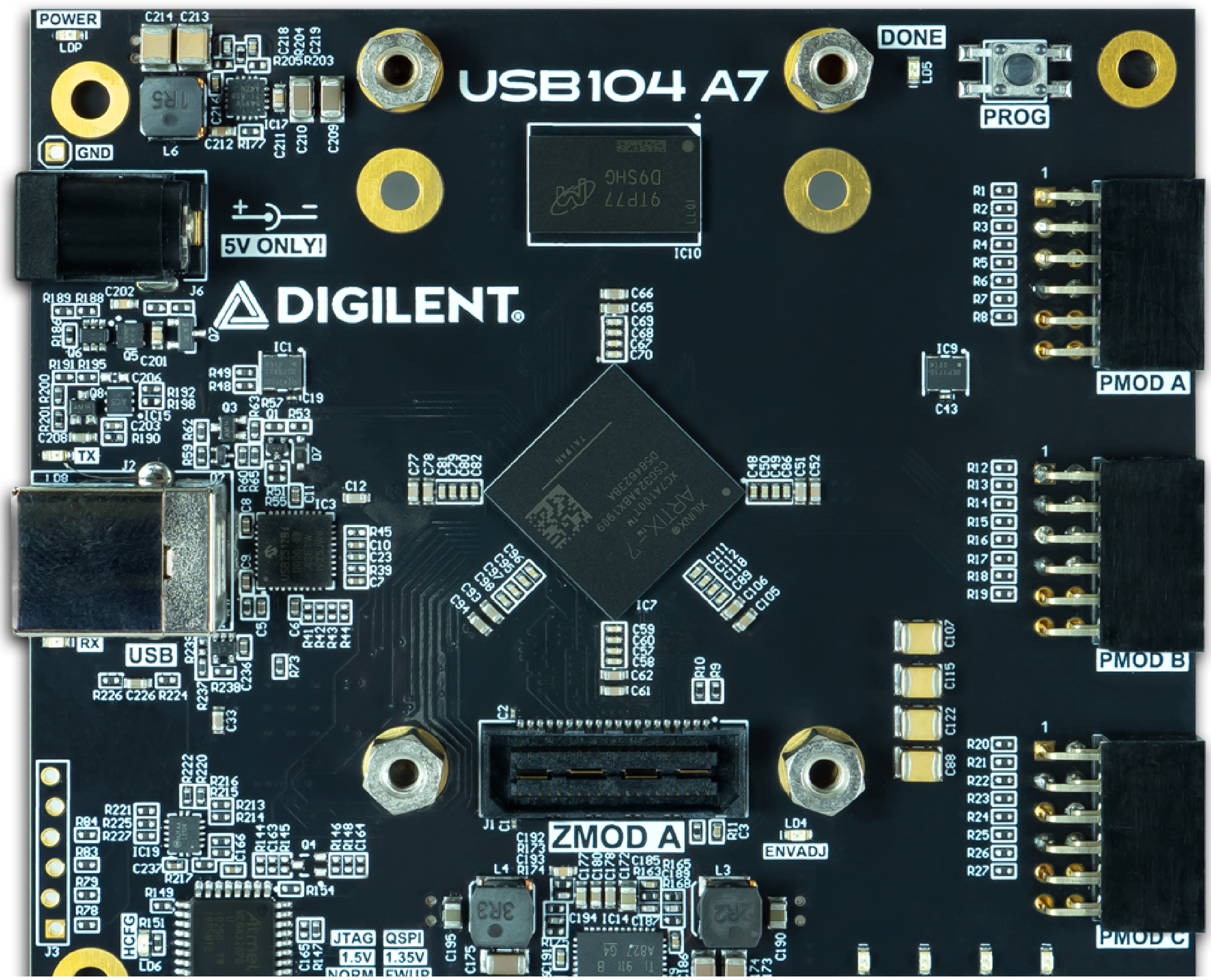
USB104 A7 Hardware Reference Manual

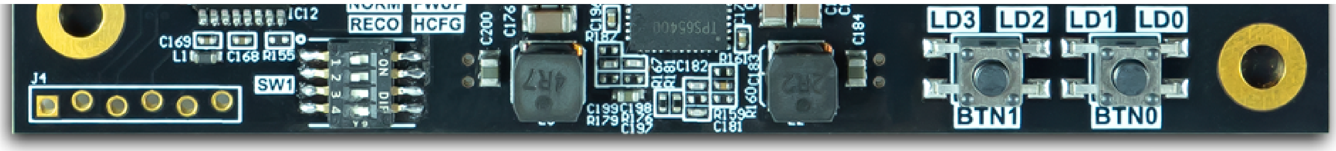
The USB104 A7 conforms to the industry-standard PC/104 form factor, and brings power and versatility to your PC/104 stackable PC. It features an Xilinx Artix-7 XC7A100T. Artix®-7 devices provide the highest performance-per-watt fabric making the USB104 A7 ideal for size, weight, and power constrained projects. I/O is provided with three 12-pin Pmod ports, a SYZYGY-compatible Zmod port. It features a USB port with multiple I/O interfaces, from UART to DSPI and DPTI, providing anything from a simple command interface up to a 40MB/s parallel interface, controllable through the Digilent Adept API ().

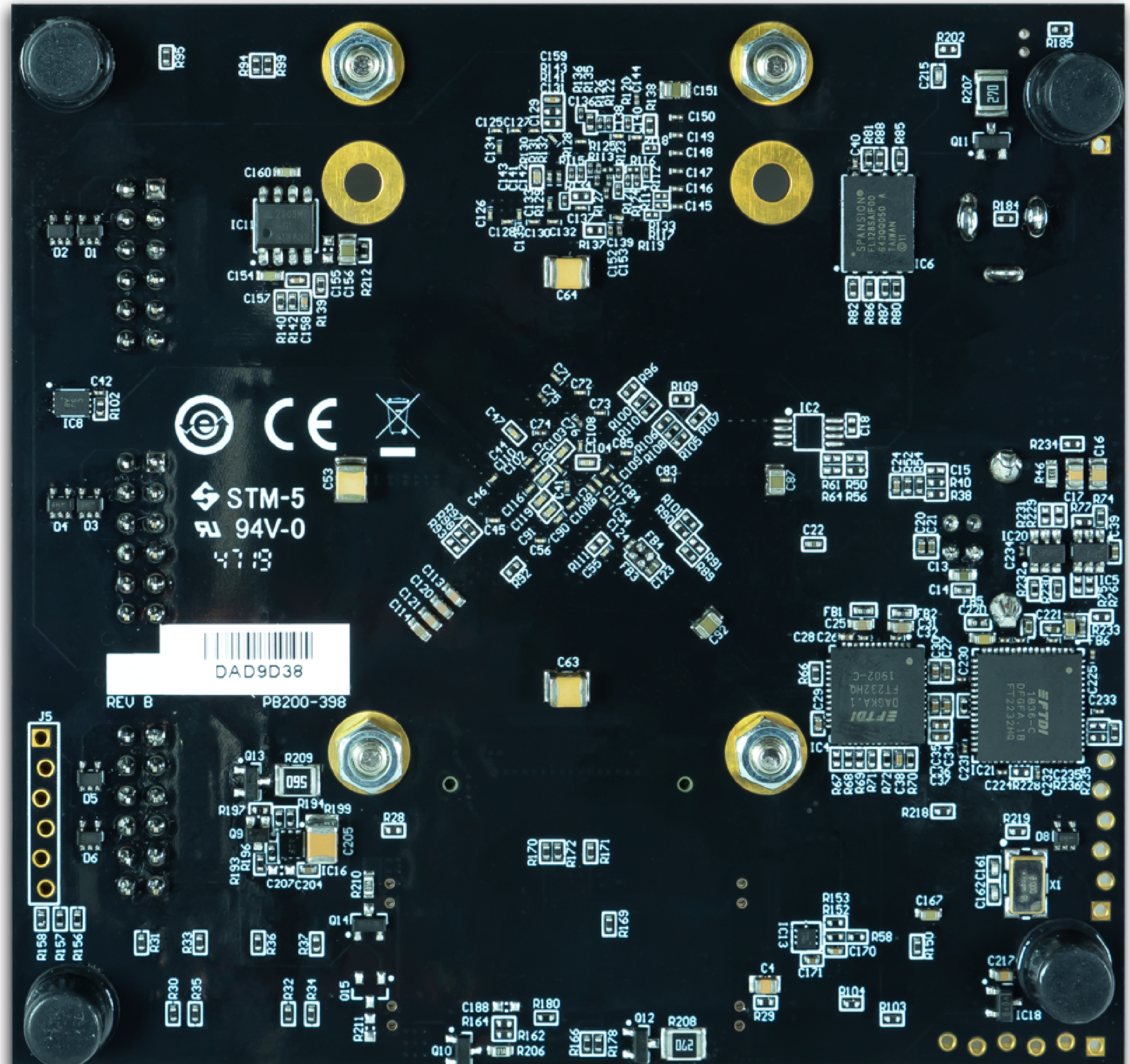


(https://reference.digilentinc.com/_media/reference/programmable-logic/usb104a7/usb104_a7-inuse-1000.png)


Zmod ADC 1410
DIGILENT
PmodDA4
PMOD A
PMOD B
PMOD C
ZMOD A
BTN1
BTN0
PROG
USB
5V ONLY!


USB104 A7
DIGILENT
POWER
GND
5V ONLY!
DONE
PROG
PMOD A
PMOD B
PMOD C
ZMOD A
ENVADJ
USB
TX
RX
JTAG
QSPI
1.5V
1.35V
ARTIX-7



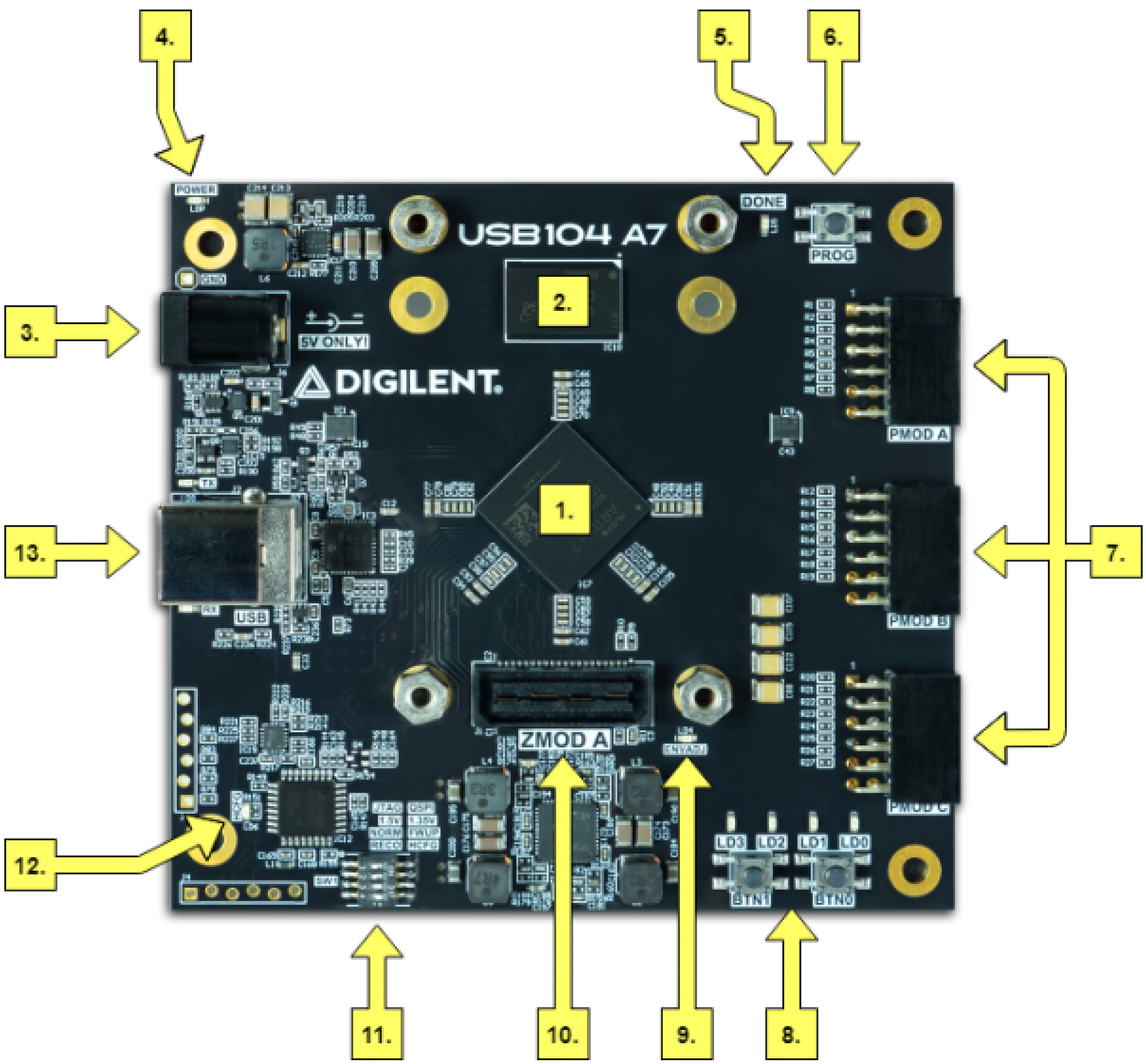


Features

- **Xilinx Artix-7 FPGA (XC7A100T-1CSG324I)**
  - 15,850 slices containing four 6-input LUTs and 8 flip-flops apiece
  - 4,860 Kbits of fast block RAM ()
  - 6 Clock Management Tiles (CMTs), each with a phase-locked loop and mixed-mode clock manager
  - 240 DSP slices
  - Internal clock speeds exceeding 450MHz
  - On-chip analog-to-digital converter (XADC) for FPGA die temperature monitoring
  - Programmable over JTAG and Quad-SPI Flash
- **Memory**
  - 512 MB () Micron DDR3 with 16-bit bus @ 800 or 667 MT/s depending on configurable voltage

- 16 MB () Spansion Quad-SPI Flash

- **Power**
  - Powered over USB or 5V external power source
- **USB**
  - DPTI/DSPI Data Transfer Interface
  - USB-JTAG Programming Circuitry
  - USB-UART Bridge
- **Zmod Port**
  - One ports following the SYZYGY Standard interface specification
  - Compatible with a variety of SYZYGY pods, allowing for a wide variety of applications
  - Dedicated differential clocks for input and output
  - 8 differential I/Os
  - 16 single-ended I/Os
  - DNA interfaces connected to Platform MCU allowing for various auto-negotiated power supply configurations
- **Pmod Ports**
  - 3 twelve-pin ports with a total of 24 FPGA-connected I/Os
- **User GPIO ()**
  - 2 push-buttons
  - 4 LEDs



USB104 A7 Callout Diagram

Callout #	Description	Callout #	Description	Callout #	Description
1	Xilinx Artix-7 FPGA	**6**	FPGA Programming Reset Button	**10**	Zmod/SYZYGY Port
2	Micron DDR3 Memory	**7**	Pmod Ports	**11**	Mode Select Switches
3	Barrel Jack for External Power Supply	**8**	User Buttons and LEDs	**12**	USB Hub Configured Indicator LED ()
4	Power Good Indicator LED ()	**9**	SYZYGY VIO Power Good Indicator LED ()	**13**	USB JTAG/UART and Data Transfer Port
5	Programming Done LED ()				

Functional Description

1. Power Supplies

The USB104 A7's power circuitry was carefully designed to meet the requirements of the Artix-7 FPGA and all peripherals while also allowing the flexibility required to support a variety of different Zmod/SYZYGY modules.

An overview of the power circuit is shown in Figure 1.1:

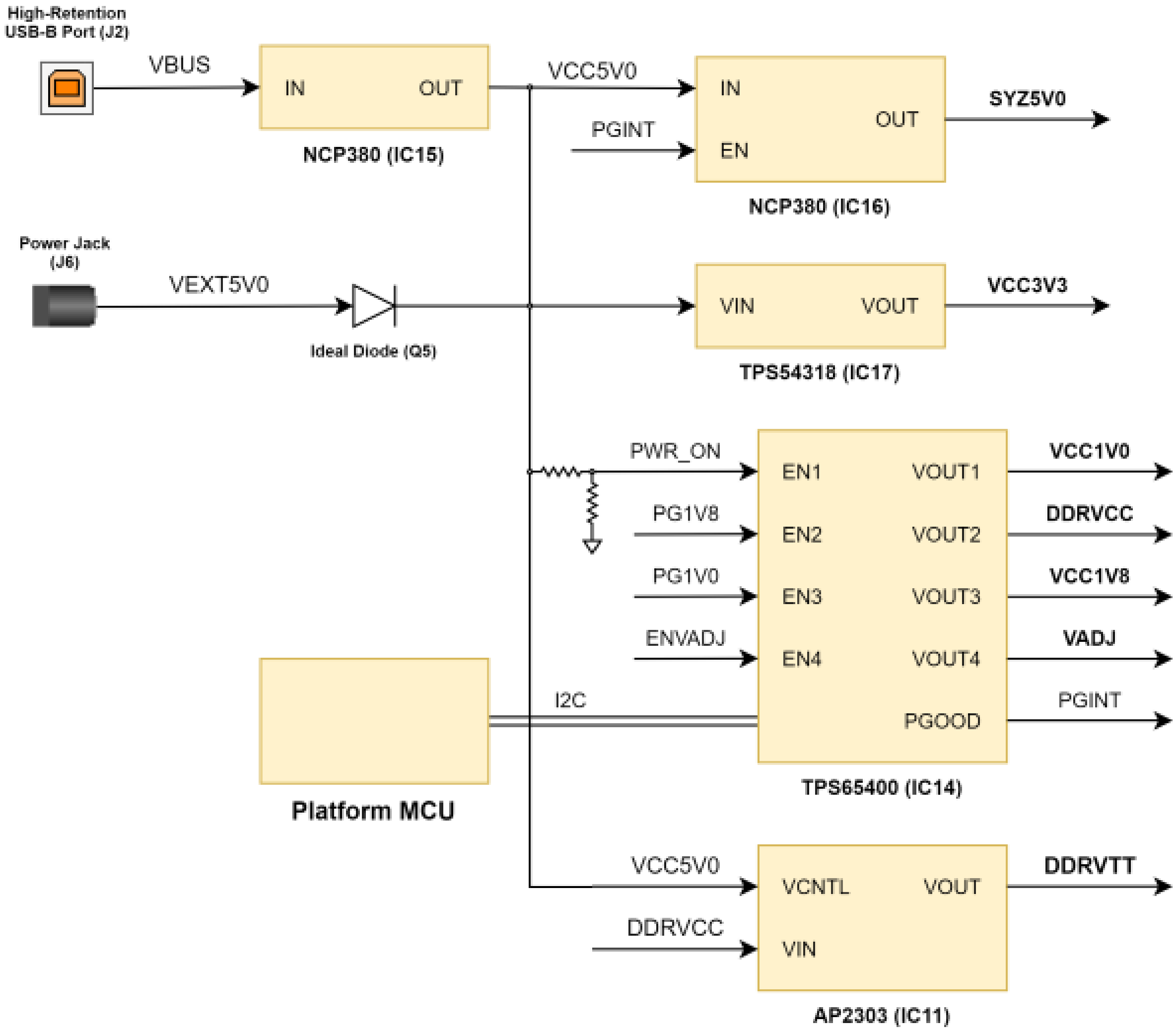


Figure 1.1: Power Circuit Overview

All on-board power supplies are automatically sequenced on when a power supply, whether USB or an external supply connected to the barrel jack (), is connected to the board.

The power indicator LED (), labeled “LDP”, is illuminated (color) when all supply rails reach their nominal voltage.

An additional indicator LED (), labeled “ENVADJ”, is illuminated when the VIO supply associated with the SYZYGY port (VADJ) is powered.

1.1. Power Input Source

The USB104 A7 can be powered either over USB or via a wall wart supply with a barrel jack, via connector J6. The wall wart supply must use a center-positive 2.1 mm internal-diameter plug and deliver between 4.65V and 5.5V DC. It should also be able to provide at least 5 A (25 Watts) in order to support power-hungry FPGA design and external peripherals. A compatible 30 Watt power supply ships with the USB104 A7.

The power input source is automatically determined by the supply circuitry. The barrel jack supply is preferred. If a supply is connected when the board is already powered through USB, then the board will seamlessly switch over to the external supply without brownout or reset. If the external supply is disconnected, the board will switch back to USB power, but the onboard supplies will temporarily shut down and then sequence back on.

Table 1.1.1: USB104 A7 Power Input Specifications

Connector Type	Connector Label	Schematic Net Name	Min/Rec/Max Voltage	Max Current Consumption
Barrel Jack	J6	VEXT5V0	4.65/5/5.5	6.0A/30W

Connector Type	Connector Label	Schematic Net Name	Min/Rec/Max Voltage	Max Current Consumption
USB Type B	J2	VBUS	4.65/5/5.5	2.0A/10W

1.2. Power Specifications

Table 1.2.1 describes the characteristics of the USB104 A7's on-board power rails. It can be used to estimate power consumption for a project, or determine how much current attached peripherals can draw before being limited.

Table 1.2.1: USB104 A7 Power Rail Specifications

Net Name	Upstream Net Name	Power IC Type	Power IC Label	Min/Typ/Max Voltage	Max Current	Major Devices and Connectors
VCC1V0	VCC5V0	Buck	IC14	1.0V +- 5%	2.5A	FPGA
VCC1V8	VCC5V0	Buck	IC14	1.8V +- 5%	1.0A	FPGA, USB JTAG/UART
DDRVCC	VCC5V0	Buck	IC14	1.35V/1.5V[1] +- 5%	2.0A	FPGA, DDR3/DDR3L
DDRVTT	DDRVCC, VCC5V0	LDO	IC11	0.675V/0.75V[2] +- 5%	0.43A	DDR3/DDR3L
VCC3V3	VCC5V0	Buck	IC17	3.3V +- 5%	3.0A	FPGA, Zmod, Pmods, USB Hub, USB JTAG/UART, USB DPTI/DSPI
VADJ	VCC5V0	Buck	IC14	1.2V +- 5% to 3.3V +- 5%	1.2A	FPGA, Zmod
SYZ5V0	VCC5V0	Load Switch	IC16	5.0V +- 5%	1.5A	Zmod

---

[1], [2] Depending on DDR voltage select switch position

1.3. Power Sequencing

The power-on sequence for the USB104 A7 is controlled by the connections between the power-good and enable pins of the supplies. The sequence is started when a power input source is connected to the board. The power supplies shut down in the opposite order when both input sources are disconnected. The startup sequence is as follows:

1. VCC5V0, SYZ5V0
2. VCC1V0
3. VCC1V8
4. DDRVCC/DDRVTT
5. VCC3V3
6. VADJ

Note: *The VADJ rail may or may not be powered on, depending on the PMCU configuration and whether a Zmod or other SYZYGY pod is installed.*

1.4. Platform MCU

As noted in previous sections, the USB104 A7 uses an Atmega328PB microcontroller (IC12), referred to as the Platform MCU (PMCU), to implement the SmartVIO requirements of the SYZYGY standard, set the DDR voltage, configure the USB Hub, and provide information about the power supply settings and SYZYGY ports to the FPGA.

The Platform MCU enumerates the SYZYGY port and determines the power needs of an installed module. The power budget of the USB104 A7 is then determined based on the needs of the Zmod, as well as other peripherals.

The PMCU firmware for the USB104 A7 supports setting the DDR operating voltage based on the position of the DDR voltage select switch (SW1/2). The state of this switch is only read during the power-on sequence. Further information on the DDR voltage select functionality can be found in Section 3, DDR3/DDR3L Memory, below.

If the hub configuration/recovery select switch (SW1/4) is placed in the “HCFG” position at power-up, the Platform MCU configures it to automatically detect which power input source is used and correctly report whether the board is self or bus powered to the USB host.

Note: *Switches SW1/3 and SW1/4 are intended for restoration of and updates to the Platform MCU firmware and the USB Hub configuration. Please reach out to Digilent Support on the Programmable Logic section of the Digilent Forum (https://forum.digilentinc.com) in the event that your board needs to be reflashed. For normal operation, SW1/3 and SW1/4 should be left in the “NORM” and “HCFG” positions, respectively.*

When the USB104 A7 is turned on, the PMCU enumerates the pods attached to the USB104 A7's SYZYGY ports and retrieves their DNA, in order to correctly configure the variable supplies.

After SYZYGY enumeration is complete, the PMCU configures itself as an I2C slave device with a chip address of 0x60. Control of the I2C bus is then handed over to the FPGA, with SCL and SDA connected to FPGA pins U16 and V17, respectively. For MicroBlaze-based designs, Xilinx's AXI IIC controller is recommended for interfacing with this bus. For more information on the PMCU's register space and communication protocol, see the PMCU specification (PDF Download (https://digilent.s3-us-west-2.amazonaws.com/resources/programmable-logic/eclypse/Eclypse-PMCU-Specification-Public.pdf)).

---

2. FPGA Configuration

After power-on, the Artix-7 FPGA must be configured (or programmed) before it can perform any functions. The FPGA can be configured in one of two ways:

1. A PC can use the Digilent USB-JTAG circuitry connected to the USB port (J2, labeled “USB”), to program the FPGA any time the power is on.
2. A file stored in the nonvolatile serial (SPI) flash device can be transferred to the FPGA using the SPI port.

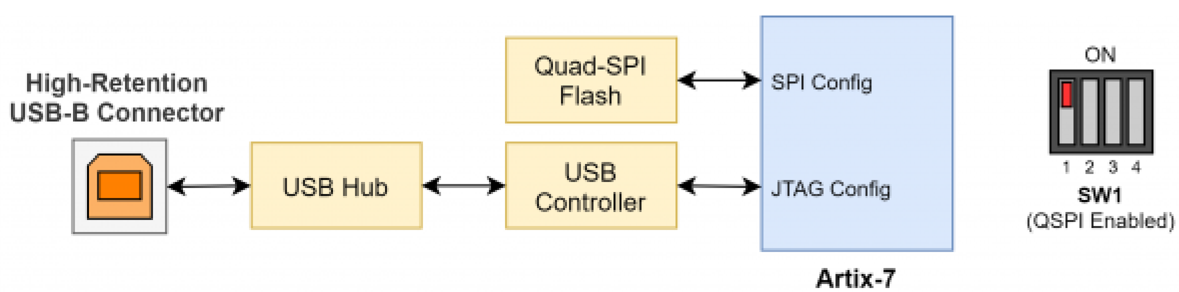


Figure 2.1: USB104 A7 Configuration Options

Figure 2.1 shows the different options available for configuring the FPGA. An on-board programming mode select switch (SW1/1) selects between the programming modes.

The FPGA configuration data is stored in files called bitstreams that have the .bit file extension. The Vivado software from Xilinx can create bitstreams from VHDL, Verilog®, or schematic-based source files.

Bitstreams are stored in SRAM-based memory cells within the FPGA. This data defines the FPGA's logic functions and circuit connections. The data remains valid until it is erased by removing board power, by pressing the reset button (BTNP, labeled "PROG") attached to the PROG input, or by writing a new configuration file to the FPGA using the JTAG port.

An Artix-7 100T bitstream is typically 30,606,304 bits and can take a long time to transfer. The time it takes to program the USB104 A7 can be decreased by compressing the bitstream before programming, and then allowing the FPGA to decompress the bitstream itself during configuration. Depending on design complexity, compression ratios of 10x can be achieved. Bitstream compression can be enabled within the Xilinx tools (Vivado) to occur during the generation of the bitstream. For instructions on how to do this, consult the Xilinx documentation for the toolset used. After being successfully programmed, the FPGA will cause the "DONE" LED () (LD5) to illuminate. Pressing the "PROG" button at any time will reset the configuration memory in the FPGA. After being reset, the FPGA will immediately attempt to reprogram itself from whatever method has been selected by the programming mode jumpers.

The following sections provide greater detail about programming the USB104 A7 using the different methods available.

2.1. JTAG Configuration

Xilinx tools typically communicate with FPGAs using the Test Access Port and Boundary-Scan Architecture, commonly referred to as JTAG. During JTAG programming, a .bit file is transferred from the PC to the FPGA using the onboard Digilent USB-JTAG circuitry (connected to port J2, labeled "USB") or an external JTAG programmer, such as the Digilent JTAG-HS2, attached to port J3. JTAG programming can be performed any time after the USB104 A7 has been powered on, regardless of what the programming mode select switch (SW1/1) is set to. If the FPGA is already configured, then the existing configuration is overwritten with the bitstream being transmitted over JTAG. Setting the mode switch to the JTAG setting (seen in Figure 2.1) can be used to prevent the FPGA from being configured from any other bitstream source until JTAG programming occurs. The maximum recommended JTAG frequency is 30 MHz ().

Programming the USB104 A7 with an uncompressed bitstream using the onboard USB-JTAG circuitry usually takes around 5 seconds. JTAG programming can be done using Vivado's hardware manager.

2.2. Quad SPI Configuration

Since the FPGA on the USB104 A7 is volatile, it relies on the Quad-SPI flash memory to store the configuration between power cycles. This configuration mode is called Master SPI. The blank FPGA takes the role of master and reads the configuration file out of the flash device upon power-up. To that effect, a configuration file needs to first be downloaded to the flash. When programming a nonvolatile flash device, a bitstream file is transferred to the flash in a two-step process. First, the FPGA is programmed with a circuit that can program flash devices, and then data is transferred to the flash device via the FPGA circuit (this complexity is hidden from the user by the Xilinx tools). This is called indirect programming. After the flash device has been programmed, it can automatically configure the FPGA at a subsequent power-on or reset event as determined by the mode switch setting (see Figure 3). Programming files stored in the flash device will remain until they are overwritten, regardless of power-cycle events.

Programming the flash can take as long as four to five minutes, which is mostly due to the lengthy erase process inherent to the memory technology. Once written however, FPGA configuration can be very fast—less than a second. Bitstream compression, SPI bus width, and configuration rate are factors controlled by the Xilinx tools that can affect configuration speed. The USB104 A7 supports x1, x2, and x4 bus widths and data rates of up to 50 MHz () for Quad-SPI programming.

Quad-SPI programming can be done using Vivado's hardware manager.

---

3. DDR3/DDR3L Memory

The USB104 A7 includes one MT41K256M16TW-107 memory component, creating a single rank, 16-bit wide interface with 512MB of capacity. It is routed to an HR (High Range) FPGA bank, powered at either 1.35V or 1.5V (discussed in Section 3.1, DDR3 Voltage Levels, below), with 40 ohm controlled single-ended trace impedance. 40 ohm internal terminations in the FPGA are used to match the trace characteristics. Similarly, on the memory side, on-die terminations (ODT) are used for impedance matching.

For proper operation of the memory, a memory controller and physical layer (PHY) interface needs to be included in the FPGA design. The easiest way to accomplish this on the USB104 A7 is to use the Xilinx 7-series memory interface solutions core generated by the MIG (Memory Interface Generator) Wizard. The MIG Wizard can generate a native FIFO-style or an

AXI4 interface to connect to user logic. This workflow allows the customization of several DDR parameters optimized for the particular application. Table 3.1 below lists the MIG Wizard settings optimized for the USB104 A7 (any settings not mentioned can be left in default state).

Table 3.1: Memory Interface Generator Settings for the USB104 A7

Setting	1.5V	1.35V
Memory type	DDR3 SDRAM	DDR3 SDRAM
Max. clock period	2500ps (800 MT/s data rate)	3000ps (666.66 MT/s data rate)
Memory part	MT41K256M16XX-107	MT41K256M16XX-107
Memory Voltage	1.5V	1.35V
Data width	16	16
Data mask	Enabled	Enabled
Recommended Input Clock Period	5000ps (200.000 MHz ())	6000ps (166.667 MHz ())
Read Burst Type and Length	Sequential	Sequential
Output Driver Impedance Control	RZQ/6	RZQ/6
Rtt (nominal) – On-die termination	RZQ/6	RZQ/6
Controller Chip Select pin	Disabled	Disabled
Internal Vref	Enabled	Enabled
Internal termination impedance	40ohms	40ohms

For clocking, it is recommended that the System clock be set to "No buffer", and connected to a 200MHz clock generated by a Clocking Wizard IP using the onboard 100MHz oscillator on pin E3 as input. The Reference clock should be set to "Use system clock".

For your convenience, both an importable UCF file and importable PRJ files for the 1.5V and 1.35V configurations have been provided on the USB104 A7 Resource Center (https://reference.digilentinc.com/reference/programmable-logic/usb104a7/start) to speed up the process of configuring the MIG.

For those using the MIG with a MicroBlaze project, the USB104 A7 MIG settings and pinout can be automatically imported from Digilent's Vivado Board Files, which can be installed into Vivado by following the steps presented in Section 3 of the Installing Vivado, Xilinx SDK, and Digilent Board Files (https://reference.digilentinc.com/vivado/installing-vivado/start) guide. The configuration imported from the board files is intended for use with the clocking scheme discussed above, without additional configuration.

For more details on the Xilinx MIG, refer to the 7 Series FPGAs Memory Interface Solutions User Guide (Xilinx UG586).

3.1. DDR3 Voltage Levels

The USB104 A7 supports two voltage levels for the DDR3/DDR3L memory. The Platform MCU reads the position of the DDR voltage select switch (SW1/2) during powerup, and sets the rail voltages appropriately. While the select switch can be set for 1.35V so that the board uses less power, this comes with a performance tradeoff, and is intended only for advanced users. The MIG.prj file provided with Digilent's board files for the USB104 A7 is intended for use with the 1.5V setting, and requires additional configuration for use with the 1.35V setting.

Table 1.4.1: DDR Voltage Select Switch Settings

Switch Position	DDRVCC Voltage	DDRVTT Voltage
1.5V (OFF)	1.5V	0.75V
1.35V (ON)	1.35V	0.675V

4. Quad-SPI Flash

FPGA configuration files can be written to the Quad-SPI Flash (Spansion part number **S25FL128SAGMF100**), and setting the programming mode select switch (SW1/0) will cause the FPGA to automatically read a configuration from this device at power on. An Artix-7 100T configuration file requires 30,606,304 bits of memory, leaving about 76% of the flash device (or ~12MB) available for user data. A common use for this extra memory is to store Microblaze programs too big to fit in the onboard Block memory (typically 128 KB). These programs are then loaded and executed using a smaller bootloader program that can fit in the block memory. It is possible to automatically generate this bootloader, roll it into a single file (called an .mcs file) that also contains the bitstream and your custom Microblaze application, and program this file into SPI Flash using Xilinx SDK and Vivado. See Xilinx Answer Record 63605 for more information.

The contents of the memory can be manipulated by issuing certain commands on the SPI bus. The implementation of this protocol is outside the scope of this document. All signals in the SPI bus are general-purpose user I/O pins after FPGA configuration. On other boards, SCK is an exception because it remains a dedicated pin even after configuration, however, on the USB104 A7 the SCK signal is routed to an additional general purpose pin (L16) that can be accessed after configuration (see Figure 4.1 below). This allows access to this pin without having to instantiate the special FPGA primitive called STARTUPE2.

Xilinx's AXI Quad SPI core can be used to read/write the flash in a Microblaze design. Refer to Xilinx's product guide for this core to learn more about using it, or to Micron's datasheet for the flash device to learn how to implement a custom controller.

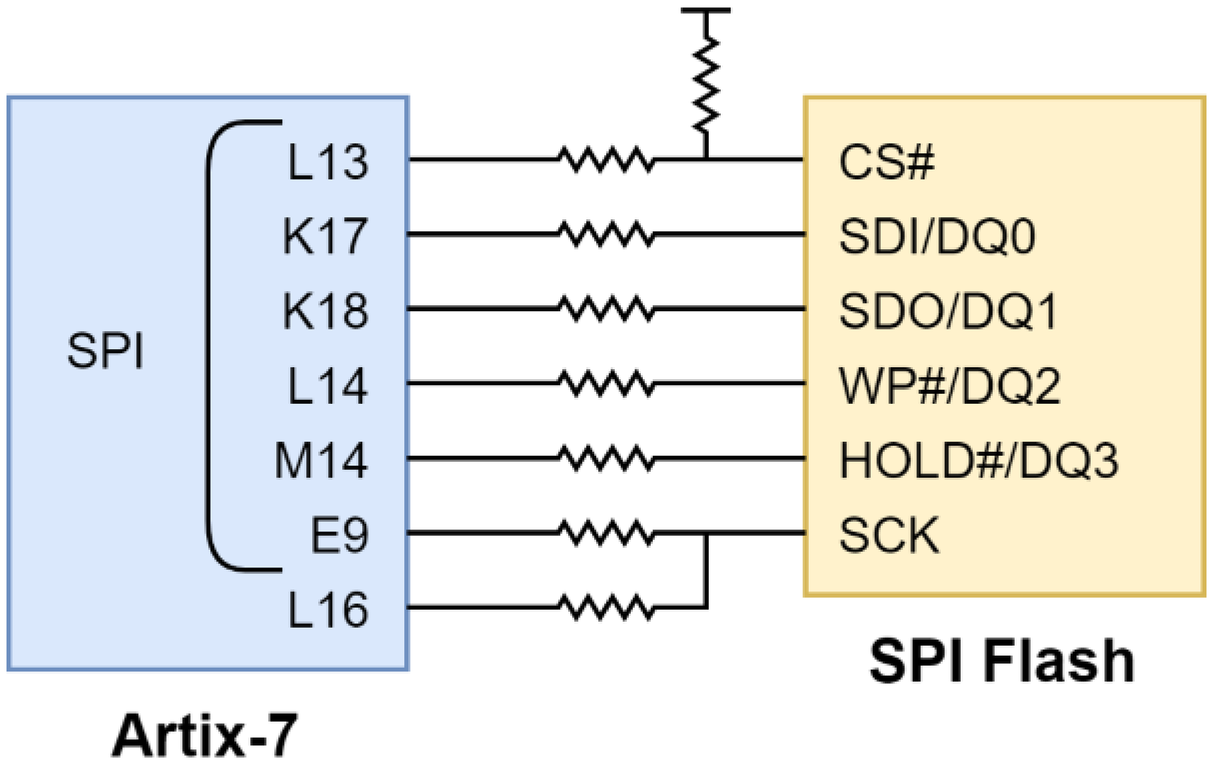


Figure 4.1 USB104 A7 SPI Flash

5. Oscillators/Clocks

The USB104 A7 includes a 100 MHz () crystal oscillator connected to pin E3, an MRCC input on FPGA bank 35.

Xilinx offers the Clocking Wizard IP core to help users generate the different clocks required for a specific design. This wizard will properly instantiate the needed MMCMs and PLLs based on the desired frequencies and phase relationships specified by the user. The wizard will then output an easy-to-use wrapper component around these clocking resources that can be inserted into the user's design. The clocking wizard can be accessed from within the Vivado and IP Integrator tools.

The 100 MHz () clock is intended as both a general purpose system clock, and to drive the system clock input of the Memory Interface Generator (MIG) IP Core to allow for proper use of the DDR3/DDR3L memory. Section 3, DDR3/DDR3L Memory, describes how to use this clock properly with the MIG. For complete information on using the MIG, see the 7 Series FPGAs Memory Interface Solutions User Guide (ug586) from Xilinx.

6. USB Port

The USB104 A7 includes a high-retention USB-B port intended for all communication with a host computer. This port is connected to a USB hub which breaks out the traffic into several different FPGA-connected interfaces, via a FTDI FT2232HQ USB-UART bridge and a FTDI FT232H hi-speed single-channel USB to UART/FIFO bridge. The USB hub and the two controller are transparent to the user. A UART interface (referred to as the USB-UART Bridge), the USB-JTAG programming interface, and a parallel data interface (DPTI) are implemented. The USB JTAG programming functionality is discussed in the FPGA Configuration section of this document. The UART and DPTI interfaces are detailed below. Each of these three interfaces is presented to the host computer as its own serial port.

Note: *The hub and bridge circuitry of the USB104 A7 is designed such that the interfaces do not interfere with one another. For example, programmers interested in using the UART or parallel transfer functionality of the USB104 A7 within their design do not need to worry about the JTAG circuitry interfering with the UART data transfers, and vice-versa.*

As long as mode switch SW1/4 is placed in the "HCFG" position, the Platform MCU configures the USB hub at power-on. Placing the switch in the "RECO" position makes the USB port and hub receive power from the host system, and is intended to be used only for firmware recovery, when the PMCU cannot be depended on for configuration.

6.1. DPTI Parallel Data Transfer Interface

The DPTI interface is an 8-bit wide parallel FIFO-style data interface supporting both asynchronous and synchronous modes. In FTDI terminology, DPTI is equivalent to "FT245-style Asynchronous or Synchronous FIFO Interface". It is available in both synchronous and asynchronous modes, configurable from the DPTI API (). In synchronous mode, data transfer is timed by the clock provided by the USB controller (connected to the FPGA on pin P17). In asynchronous mode, data transfer occurs on transitions of the read and write control signals. THe USB controller emulates a FIFO memory, providing status signals about the availability of data to be read and free space for data to be written. The FPGA controls data transfer through read, write, and output enable signals. Data transfer speeds of up to 40 MB ()/s are supported in synchronous mode. This interface is summarized in Figure 6.1.1, below:

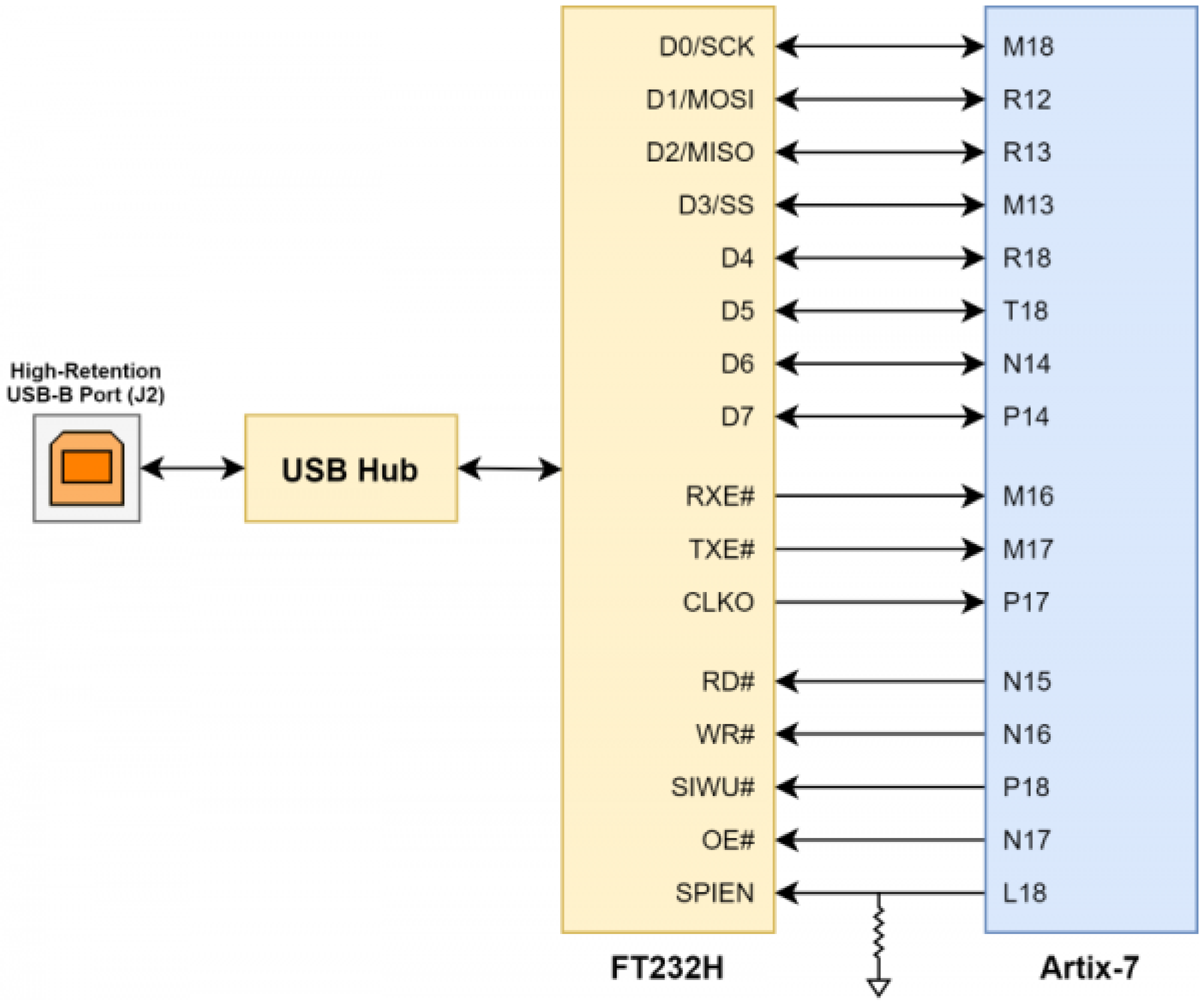


Figure 6.1.1 USB104 A7 DPTI Connections

Table 6.1.1. DPTI Signal Descriptions

Signal	Direction (FPGA)	Description
D[7:0]	I/O	Data bus.
RXE#	Input	When low, data is available to be read from the FIFO.
TXE#	Input	When low, data can be written to the FIFO.
RD#	Output	A low pulse triggers data to be read out from the FIFO.
WR#	Output	A low pulse triggers data to be written to the FIFO.

Signal	Direction (FPGA)	Description
SIWU#	Output	Send Immediate or Wake-up. In normal mode, a low pulse triggers sending a data packet with teh data currently in the FIFO, even if below the normal packet size. In suspend mode, a low pulse can wake up the host computer.
OE#	Output	When low, the data bus is driven by the USB controller (read transfer). When high, the bus is driven by the FPGA (write transfer).
CLKO	Input	60 MHz () clock used in synchronous mode. Data is launched and can be captured on the rising edge.

Microblaze support for DPTI is provided by the AXI DPTI IP core which can be found in the vivado-library (https://github.com/Digilent/vivado-library) repository on Digilent's GitHub.

The Digilent Adept Runtime can be used with the the Digilent Adept API () in order to simplify the host side communication through DPTI. The Adept Runtime can be downloaded through the Adept (https://reference.digilentinc.com/reference/software/adept/start) wiki page. The Zmod ADC () and Zmod DAC () example projects, which can be found on the USB104 A7 Resource Center (https://reference.digilentinc.com/reference/programmable-logic/usb104a7/start) include example PC-side applications intended for receiving and transmitting data through the DPTI interface.

Note: *A DSPI interface is also connected, but no demos are provided as of time of writing. Since the interfaces share pins, DPTI and DSPI cannot be used simultaneously. The pulldown resistor on the SPI enable line means that the default interface is DPTI. SPIEN should be held low or not be driven by the FPGA while using DPTI.*

6.2. USB-UART Bridge (Serial Port)

The USB104 A7's FTDI FT2232HQ USB-UART bridge allows you use PC applications to communicate with the board using standard Windows COM port commands. Free USB-COM port drivers, available from www.ftdichip.com (http://www.ftdichip.com) under the “Virtual Com Port” or VCP heading, convert USB packets to UART/serial port data. These drivers are included when installing Vivado. Serial port data is exchanged with the FPGA using a two-wire serial port (TXD/RXD). After the drivers are installed, I/O commands can be used from the PC directed to the COM port to produce serial data traffic on the U12 (UART_TXD_IN, FPGA input) and V12 (UART_RXD_OUT, FPGA output) pins.

Two on-board status LEDs provide visual feedback on traffic flowing through the port: the transmit LED () (labeled “TX”) and the receive LED () (labeled “RX”), placed adjacent to the USB port. Signal names that imply direction are from the point-of-view of the DTE (Data Terminal Equipment), in this case the PC.

The connections between the FT2232HQ and the Artix-7 are shown in Figure 6.2.1:

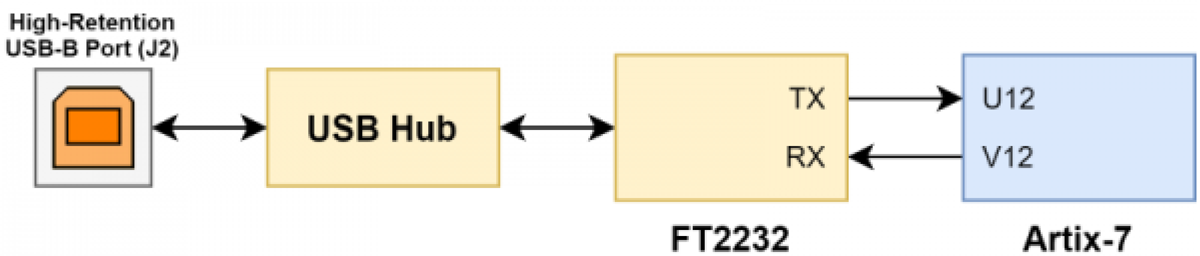


Figure 6.2.1 USB104 A7 UART Connections

7. Zmod Port

The USB104 A7 features one Zmod ports, which uses a SYZYGY Standard interface to communicate with an installed SYZYGY pod. The port is compatible with version 1.1 of the SYZYGY specification from Opal Kelly.

SYZYGY SmartVIO functionality is implemented by the Eclypse's Platform MCU, as discussed in the 1. Power Supplies section of this document. Each port's SYZYGY DNA is connected to both the Platform MCU and to the Artix-7 FPGA's U16 and V17 pins through a single I2C bus. Once the board is fully powered on, and the PMCU has configured itself in I2C slave mode, SYZYGY DNA data can be read directly from the pods, and the negotiated voltages and currents can be read from the PMCU over this bus. In addition, a SYZYGY detect signal is provided to the FPGA by the PMCU, connected to FPGA pin T11. This signal is asserted to logic '1' when a pod is connected.

Warning: *SYZYGY pods are NOT hot-swappable. Connecting or disconnecting a pod from the USB104 A7 while the board is powered on may cause damage to the pod and/or the board, and is to be avoided.*

A SYZYGY Standard interface contains 16 single-ended I/O pins, 8 differential I/O pairs (which can alternatively be used as 16 additional single-ended I/O pins), and two dedicated differential clocks - one for input and one for output. Bank 15 of the FPGA is dedicated to the Zmod port and is powered by a dedicated adjustable rail, configured by the Platform MCU as the USB104 A7 is powered on. Template constraints for the Zmod port can be found in the USB104 A7's Master XDC file, available through Digilent's 🌍 digilent-xdc (https://github.com/Digilent/digilent-xdc) repository on Github.

Digilent provides Eclypse-compatible low-level IPs, scripted Vivado flows, and software libraries to support each Digilent Zmod (https://reference.digilentinc.com/reference/zmod/start). Demos are available for the Zmod ADC () and the Zmod DAC () on the USB104 A7's Resource Center (https://reference.digilentinc.com/reference/programmable-logic/usb104a7/start).

For more information on the SYZYGY standard, see 🌍 syzygyfpga.io (https://syzygyfpga.io/).

7.1. SYZYGY Pod Compatibility

The USB104 A7's Zmod port is compatible with a variety of different SYZYGY pods. Information required to determine if the USB104 A7 is compatible with a certain pod is summarized in Table 7.1.1:

Table 7.1.1: SYZYGY Compatibility Table

Parameter	Zmod A (STD)
Port Type	Standard
	Single-Wide
Total 5V Supply Current	1.5A
Total 3.3V Supply Current	1.0A
VIO Supply Voltage Range	1.2V to 3.3V
Total VIO Supply Current	1.0A (VIO Group 1)
Port Groups	Group 1: A
I/O Count	28 total (8 DP)
Length Matching	27 mm +- 0.2 mm (includes Artix package delay)

8. Pmod Connectors

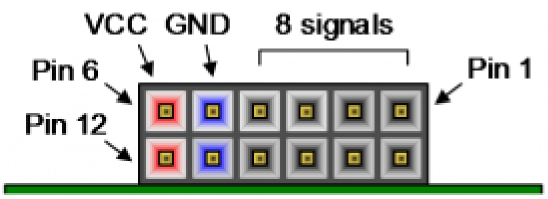
Pmod connectors are 2×6, right-angle, 100-mil spaced female connectors that mate with standard 2×6 pin headers. Each 12-pin Pmod connector provides two 3.3V VCC () signals (pins 6 and 12), two Ground signals (pins 5 and 11), and eight logic signals, as shown in Figure 8.1. The VCC () and Ground pins can deliver up to 1A of current, but care must be taken not to exceed any of the power budgets of the onboard regulators or the external power supply (these are described in the "Power supplies" section).

Warning: *Since the Pmod pins are connected to Spartan-7 FPGA pins using a 3.3V logic standard, care should be taken not to drive these pins over 3.4V.*

Digilent produces a large collection of Pmod accessory boards that can attach to the Pmod expansion connectors to add ready-made functions like A/D's, D/A's, motor drivers, sensors, and other functions. See www.digilentinc.com (http://www.digilentinc.com) for more information.

The USB104 A7 has four Pmod connectors, some of which behave differently than others. Each Pmod connector falls into one of two categories: standard or high-speed. Also, some Pmod connectors share their connections with the inner rows of the shield connector, and should not be used at the same time as a shield that requires those pins. Table 8.1 specifies which category each Pmod falls into, whether it shares any pins with the shield connector, and also lists the FPGA pins they are connected to. The following sections describe the different types of Pmods.

Note: *Some Pmods use a 1×6 pin header. Designs and IP provided by Digilent for these Pmods connect their pins to the top row of the standard 2×6 pin interface (pins 1-4).*



(https://reference.digilentinc.com/_detail/basys3-pmod_connector.png?id=reference%3Aprogrammable-logic%3Ausb104a7%3Areference-manual) *Figure 8.1. Pmod connector.*

Table 8.1: USB104 A7 Pmod Pinout.

	Pmod JA	**Pmod JB**	**Pmod JC**
Pmod Type	**Standard**	**Standard**	**Standard**
Pin 1	F4	C4	C5
Pin 2	F3	B2	C6
Pin 3	E2	B3	B6
Pin 4	D2	B4	C7
Pin 7	H2	B1	A5
Pin 8	G2	A1	A6
Pin 9	C2	A3	B7
Pin 10	C1	A4	D8

The standard Pmod connectors are connected to the FPGA via 200 Ohm series resistors. The series resistors prevent short circuits that can occur if the user accidentally drives a signal that is supposed to be used as an input. The downside to this added protection is that these resistors can limit the maximum switching speed of the data signals.

9. User GPIO

The USB104 A7 includes 2 push buttons and 4 individual LEDs, as shown in Figure 7.1.

The push buttons are connected to the FPGA via series resistors to prevent damage from inadvertent short circuits which could occur if an FPGA pin assigned to a button was defined as an output. The two push buttons are “momentary” switches that normally generate a low output when they are at rest, and a high output only when they are pressed.

The four individual high-efficiency LEDs are anode-connected to the FPGA via 330-ohm resistors, so they will turn on when a logic high voltage is applied to their respective I/O pin.

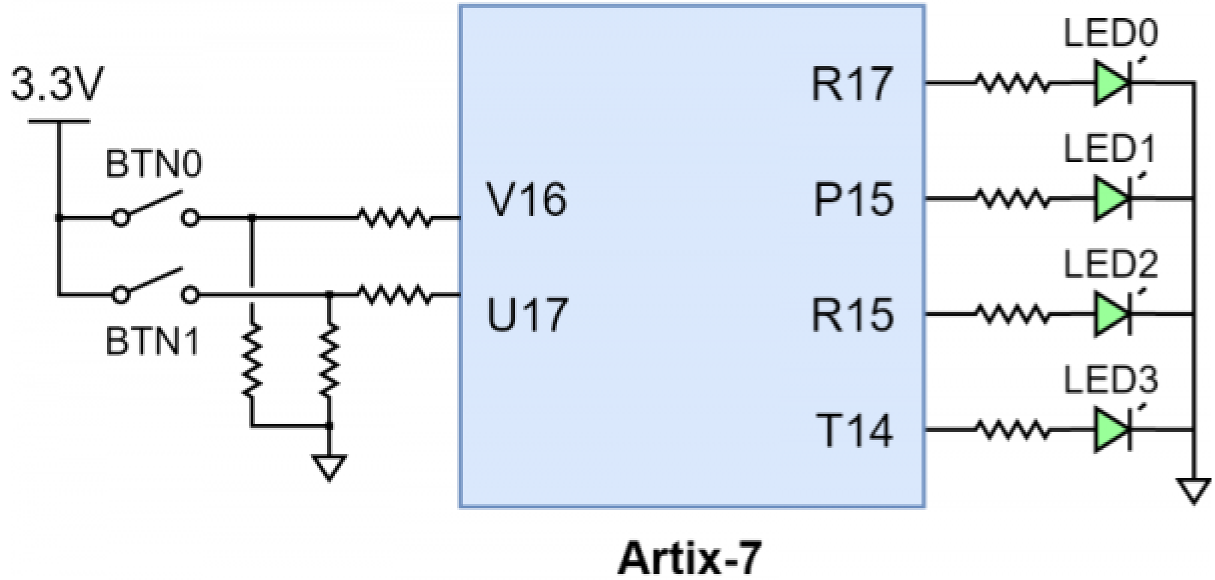


Figure 9.1. USB104 A7 GPIO ()

Company (https://store.digilentinc.com/company/)

- About Us (https://store.digilentinc.com/company/#about-digilent)
- FAQs (https://store.digilentinc.com/company/#faqs)
- Shipping & Returns (https://store.digilentinc.com/shipping-returns/)
- Jobs (https://store.digilentinc.com/company/#jobs)
- Legal & Privacy (https://store.digilentinc.com/legal-privacy/)

News (https://store.digilentinc.com/news/)

- Blog (https://blog.digilentinc.com/)
- Newsletter (https://store.digilentinc.com/news/#newsletter)
- Events (https://store.digilentinc.com/news/#events)

Affiliations (https://store.digilentinc.com/affiliations/)

- List of Distributors (https://store.digilentinc.com/affiliations/#distributors)
- Technology Partners (https://store.digilentinc.com/affiliations/#partners)

Subscribe to our newsletter

Get the latest updates on new products and upcoming sales

Your email address

Submit

Contact Us

- Support Channels (https://store.digilentinc.com/support/#channels)

Digilent
1300 NE Henley Ct. Suite 3
Pullman, WA 99163
United States of America

- (http://twitter.com/DigilentInc)
- (http://facebook.com/Digilent)
- (https://www.youtube.com/user/DigilentInc)
- (https://github.com/digilent)
- (https://instagram.com/digilentinc)
- (https://www.linkedin.com/company/1454013)
- (https://www.flickr.com/photos/127815101@N07)