AN-1006
APPLICATION NOTE

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

Using the EVAL-ADUSB2EBZ

by Brett Gildersleeve

INTRODUCTION

The EVAL-ADUSB2EBZ features USB-to-I²C and SPI conversion. It is compatible with 1.8 V and 3.3 V target devices and allows for SigmaStudio™ integration for most SigmaDSP® processors. Its on-board power regulators are capable of supplying the target board, and it features a standard Aardvark-compatible programming header. The EVAL-UDSUB2EBZ provides SPI control of up to five slave devices with a low profile surface-mount USB miniature Type B connector, and it allows for plug-and-play operation.

The EVAL-ADUSB2EBZ is ideal for downloading code and register settings to SigmaDSP processors and codecs with SigmaStudio. It can also be used for real-time tuning of SigmaDSP production units with SigmaStudio.

GENERAL DESCRIPTION

The EVAL-ADUSB2EBZ, also known as the USBi, is a standalone communications interface and programmer for SigmaDSPsystems. It translates USB control commands from SigmaStudio to the I²C and SPI communications protocols. The USBi is powered over the USB cable; therefore, no external power supply is required.

The ribbon cable and 10-pin header form a bridge to the target board to connect the communications signals to the target IC. The ribbon cable also carries 5 V power from the USB hub, which can be used to power the target board if desired.

The on-board regulators enable both 1.8 V and 3.3 V IOVDD operation, allowing for increased compatibility with target devices.

Up to five slave devices can be controlled by the USBi simultaneously. To control multiple SPI devices, additional latch signals are provided, although they are not connected to the ribbon cable.

The USBi can be used to control SigmaDSP systems in real time via SigmaStudio, and is capable of programming an EEPROM in self-boot systems. It is an ideal solution for in-circuit programming and tuning of prototype systems.

The USBi only supports USB 2.0 interfaces; the USBi will not work with PCs that only support USB Version 1.0 and USB Version 1.1.

FUNCTIONAL BLOCK DIAGRAM

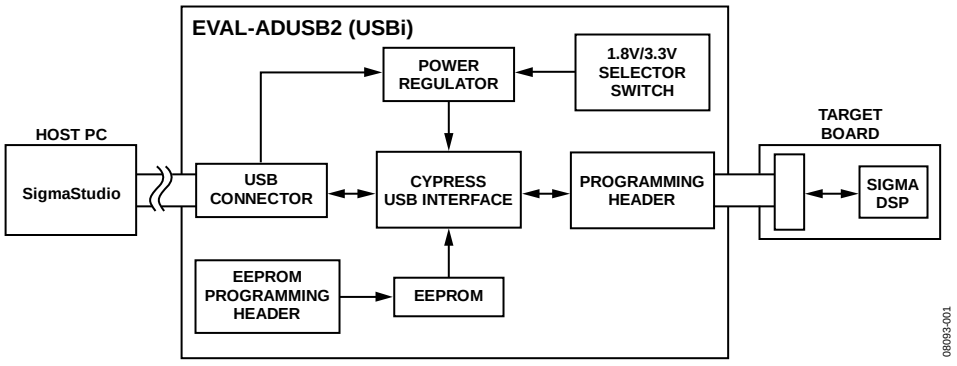

Figure 1.

TABLE OF CONTENTS

REVISION HISTORY

4/10—Rev. 0 to Rev. A

Changes to General Description Section 1
Added Warning Section 6

5/09—Revision 0: Initial Version

USING THE USB INTERFACE WITH SIGMASTUDIO

INSTALLING THE DRIVERS

SigmaStudio must be installed to use the USBi. Once SigmaStudio has been properly installed, connect the USBi to an available USB port with the included USB cable. At this point, Windows® XP recognizes the device and prompts the user to install drivers.

Figure 2. Found New Hardware Notification

Select the **Install from a list or specific location (Advanced)** option and click **Next >**.

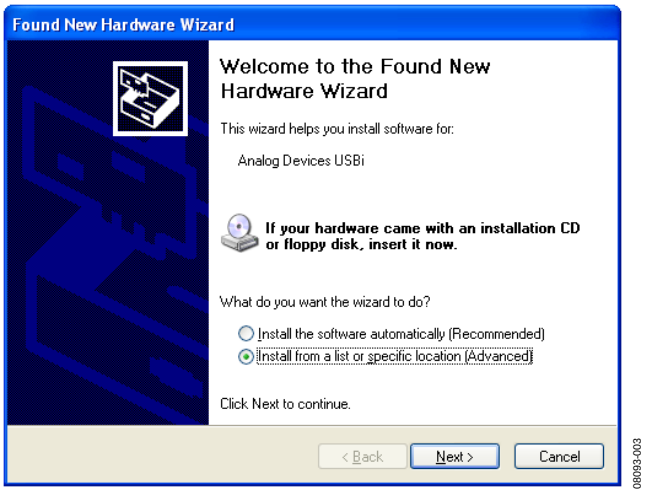

Figure 3. Found New Hardware Wizard—Installation

Click **Search for the best driver in these locations**, then select **Include this location in the search**. Click **Browse** to find the SigmaStudio 3.0\USB drivers directory.

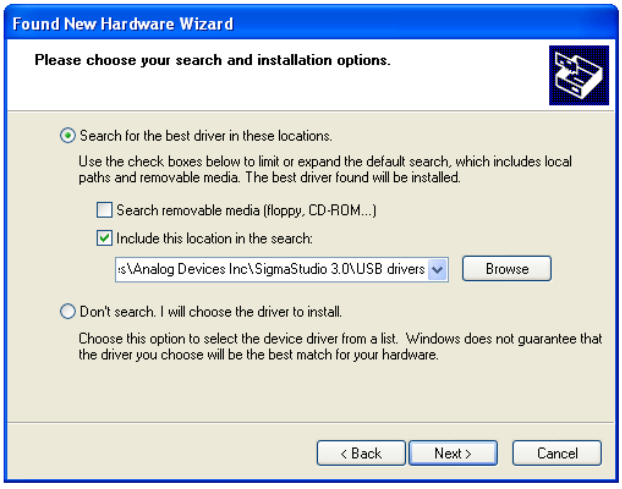

Figure 4. Windows Found New Hardware Wizard—Search and Installation Options

When the warning about Windows Logo testing appears on the screen, click **Continue Anyway**.

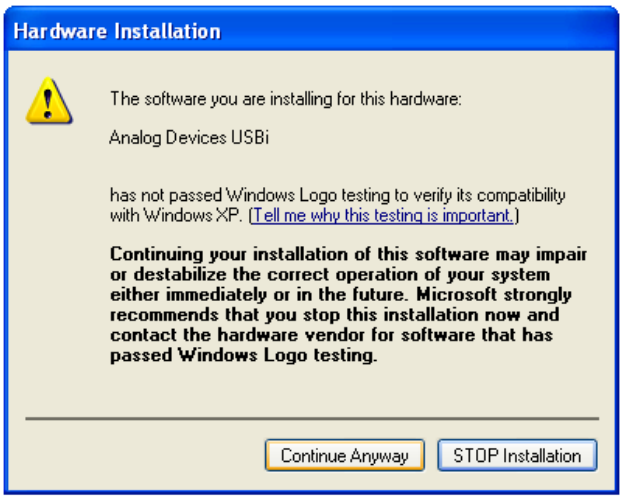

Figure 5. Windows Logo Testing Warning

ADDING THE USBi TO A SIGMASTUDIO PROJECT

To use the USBi in conjunction with SigmaStudio, first select it in the **Communication Channels** subsection of the toolbox in the **Hardware Configuration** tab, and add it to the project space.

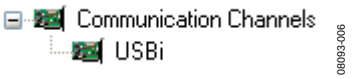

Figure 6. Adding the USBi Communication Channel

If SigmaStudio cannot detect the USBi on the USB port of the computer, then the background of the **USB** label will be red. This may happen when the USBi is not connected or when the drivers are incorrectly installed.

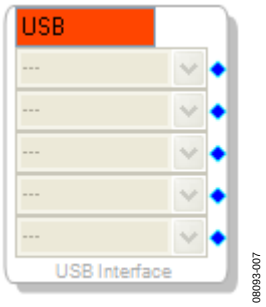

Figure 7. USBi Not Detected by SigmaStudio

If SigmaStudio detects the USBi on the USB port of the computer, the background of the **USB** label changes to orange.

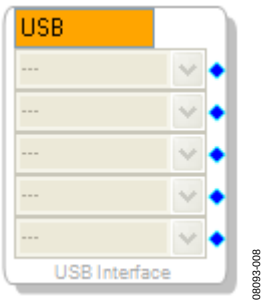

Figure 8. USBi Detected by SigmaStudio

CONFIGURING THE USBi TO COMMUNICATE WITH AN IC

To use the USBi to communicate with the target IC, connect it by click-dragging a wire between the blue pin of the USBi and the green pin of the IC. The corresponding drop-down box of the USBi automatically fills with the default mode and channel for that IC.

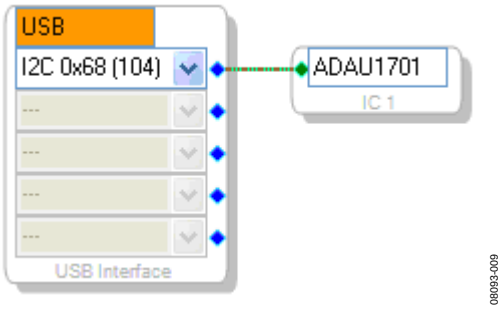

Figure 9. Connecting the USBi to an IC

To change the communication mode and channel, click the drop-down box and select the appropriate mode and channel from the list.

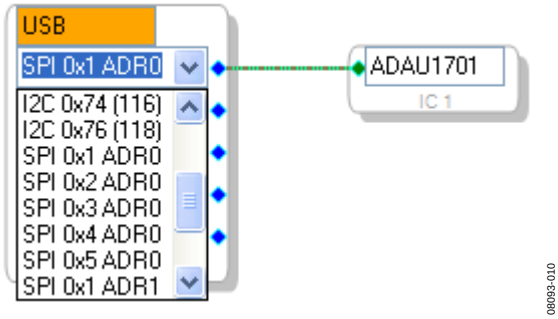

Figure 10. Selecting the Communications Mode and Channel

CONFIGURING THE USBi TO COMMUNICATE WITH MULTIPLE ICS

The USBi can communicate with up to five ICs simultaneously. To communicate with more than one IC, add another IC to the project and connect it to the next available pin of the USBi.

Multiple Address Operation with I²C

The USBi can support up to four identical devices on the same bus if the I²C address pins of the target devices are independently set to four different addresses, matching the addresses in the drop-down box in the **Hardware Configuration** tab of SigmaStudio.

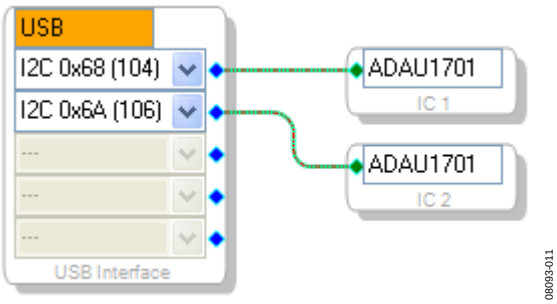

Figure 11. Multiple Address Operation with I²C

Multiple Address Operation with SPI

The USBi can support up to two identical devices on the same SPI latch if the SPI address pins of the target devices are independently set to two different addresses, matching the addresses in the drop-down box in the **Hardware Configuration** tab of SigmaStudio.

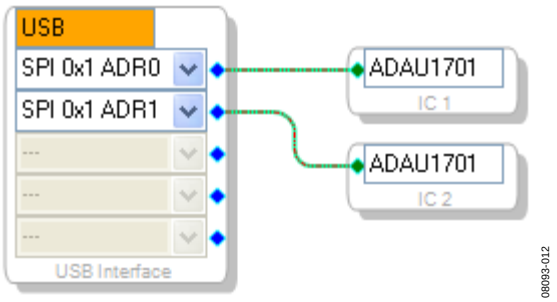

Figure 12. Multiple Address Operation with SPI

Multiple Latch Operation with SPI

The USBi can support devices on five different SPI latches. When multiple latches are used, the additional SPI latch signals from the USBi that are not connected to the ribbon cable need to be manually wired to the target.

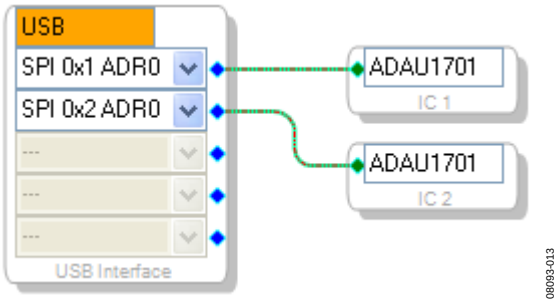

Figure 13. Multiple Latch Operation with SPI

The locations of extended SPI latch signals are shown in Figure 14.

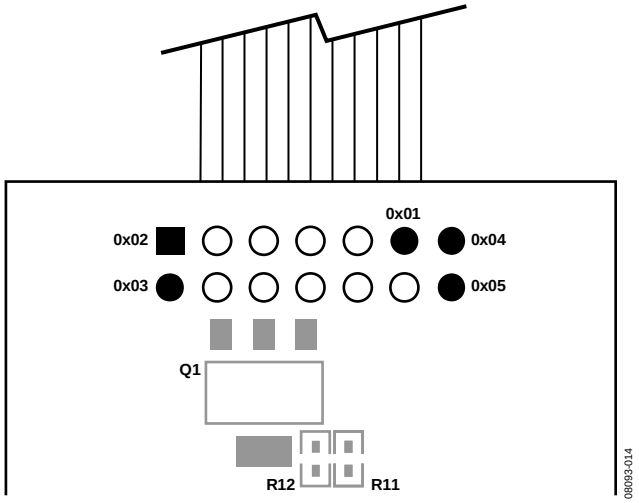

Figure 14. Extended SPI Latch Signal Pinout (Bottom View of Board)

Combined Multiple Latch and Multiple Address Operation with SPI

A combination of multiple latch and multiple address schemes can be used, but the total number of devices cannot exceed five.

CONTROLLING THE USBi

The USBi has several functions for controlling the target hardware. The control options are accessed in SigmaStudio by right-clicking on the **USB Interface** in the **Hardware Configuration** tab.

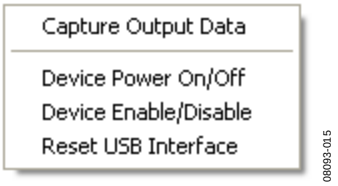

Figure 15. USBi Control Menu

Capture Output Data

This option accesses the **Capture Window**, which displays a log of all communication between the PC and the target IC (see Figure 17).

Device Power On/Off

This option switches the line that supplies power to the target board. By default, the device power is on.

Device Enable/Disable

For supported ICs, selecting this option switches the device to low power mode.

Reset USB Interface

This function performs a software reset of the USB driver, and causes the Cypress USB microcontroller to reload its firmware.

MONITORING THE USBi

Using the **Capture Window**, it is possible to view all outgoing communications transfers from the PC to the target IC. For each write, the write mode, time of write, cell name (if applicable), parameter name, address, value, data (in decimal and hexadecimal), and byte length are shown.

For block writes where more than one memory location is written, only the first location is shown. The expand/collapse button in the leftmost column allows the user to view the full data write.

USING THE USBi TO PROGRAM A SELF-BOOT EEPROM

After compiling a project, the registers and RAM contents can be written to a target EEPROM for self-boot. To use this functionality, an EEPROM IC must be connected to the USBi in the **Hardware Configuration** window. After verifying that the EEPROM write protect pin is disabled on the target board, right-click the target IC (SigmaDSP), and select **Write Latest Compilation to E2PROM**.

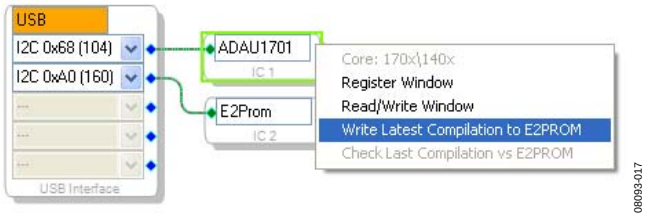

Figure 16. Writing to the Self-Boot EEPROM

WARNING

The USBi has an EEPROM on the I²C bus at Address 0x51, which it uses to indicate its Vendor ID and Product ID to the PC, as well as boot its internal program. You should avoid having any other EEPROMs in your system design at this address. This EEPROM is not write-protected; therefore, if you attempt to write to Address 0x51, you will overwrite the USBi's onboard EEPROM, and the USBi will cease to function. The USBi cannot be reprogrammed without returning the board to Analog Devices. Most EEPROMs are set to Address 0x51 by setting its pins A0 = 1 and A1 = A2 = 0.

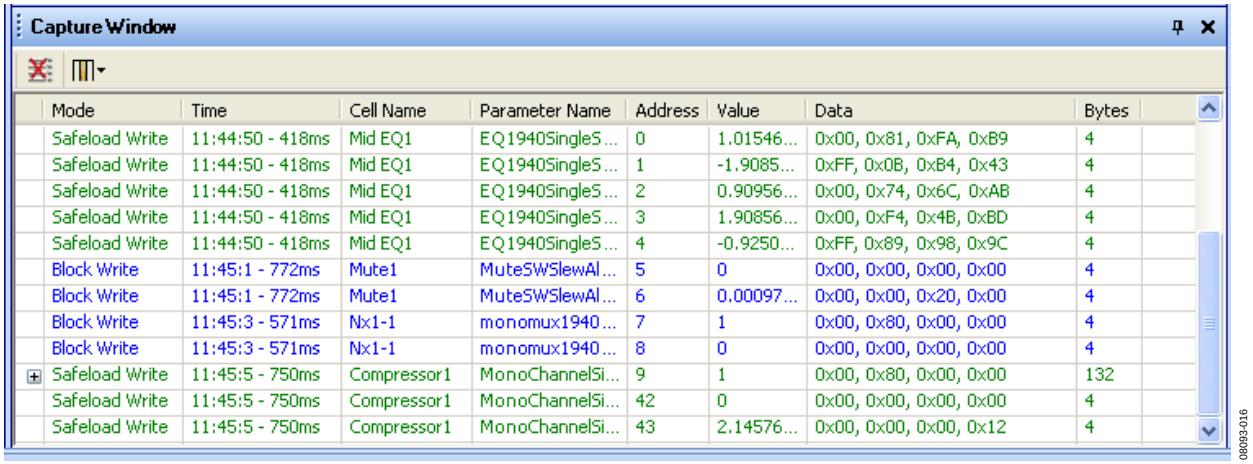

Figure 17. Output Capture Window

CIRCUIT SCHEMATICS

USB CONNECTOR

The connection between the host PC and the Cypress USB interface device is via a standard USB cable that carries D+ and D− signals for data communications, a 5 V power supply, and ground. The D+ and D− lines are a one-wire communication interface carried by half-duplex differential signals on a twisted pair. The clock is embedded in the data using the nonreturn-to-zero inverted (NRZI) line code. These signal lines connect directly to pins on the Cypress USB interface.

A surface-mounted USB miniature Type B jack was selected for its low profile and increasing ubiquity in consumer electronics.

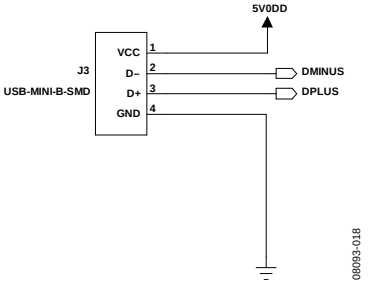

Figure 18. USB Connector Schematic

POWER REGULATOR

The Cypress USB Interface I/O ports are capable of operating in both 1.8 V and 3.3 V modes, depending on the target device in the system. Two regulators, one for 5 V to 3.3 V regulation and the other for 5 V to 1.8 V regulation, run simultaneously when the board is powered. A switch (S1) is provided to easily switch the IOVDD supply between the two regulators. LED D4 provides visual feedback that the board is being supplied with 5 V power from the PC USB port.

The position of switch S1 should not be changed when the board is connected to the USB bus.

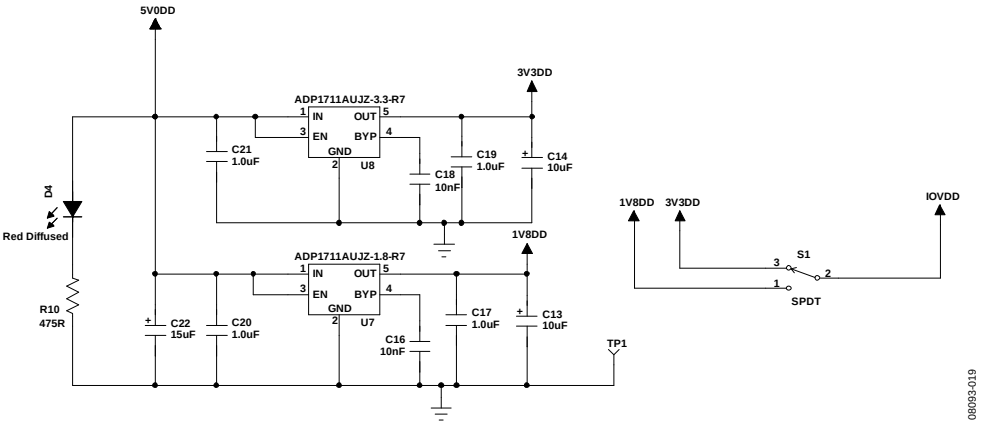

Figure 19. Power Regulator Schematic

CYPRESS USB INTERFACE

The Cypress USB interface is the core of the system, including all of the necessary functionality to convert USB commands into corresponding I²C or SPI read/write transfers, and acts as a FIFO to route data between the host PC and the target device.

CRYSTAL OSCILLATOR SCHEMATIC

The Cypress USB interface is its own clock master, and the board includes a crystal oscillator circuit with a 24 MHz piezoelectric crystal resonator to provide stability to the oscillator circuit. The crystal resonator is driven in parallel by the XTALOUT and XTALIN pins of the Cypress USB interface.

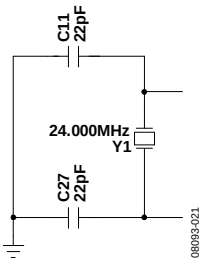

Figure 20. Crystal Oscillator Schematic

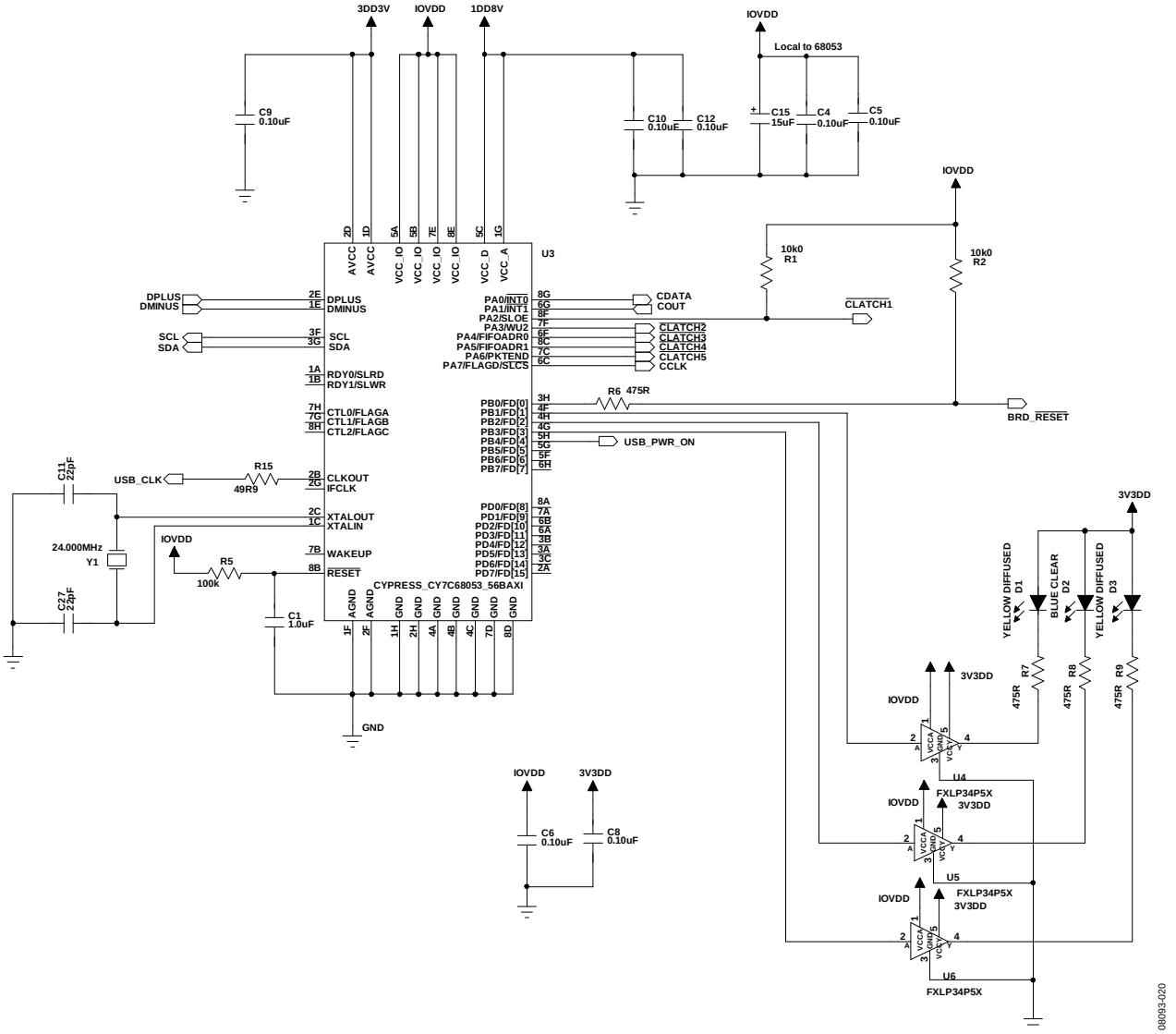

Figure 21. Cypress USB Interface Schematic

LEDS

The LEDs provide feedback to the user about the status of the Cypress USB microcontroller.

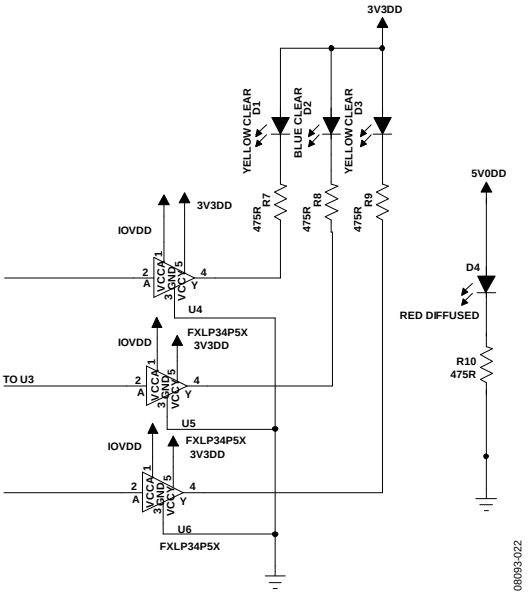

Figure 22. LEDs Schematic

Table 1. LED Functions

Reference Designator	Color	Functionality
D1	Yellow	I²C mode is active
D2	Blue	GPIO LED, for firmware debug purposes
D3	Yellow	SPI mode is active
D4	Red	5 V power being is supplied over the USB bus

EEPROM

The EEPROM is an important system element that identifies the board to the host PC and stores the firmware for the Cypress USB Interface. The EEPROM is programmed during manufacturing via the J2 connector.

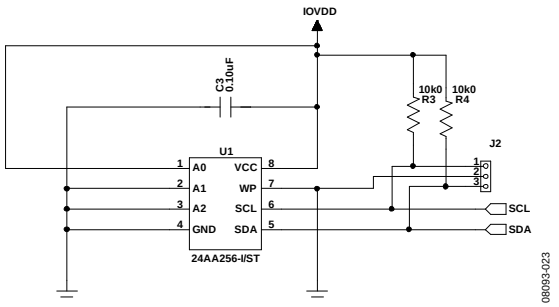

Figure 23. EEPROM Schematic

TARGET BOARD POWER SWITCH

The USBi is capable of supplying power to the target board after the Cypress USB microcontroller has finished its boot up process. The USB_PWR_ON signal connects to the base of Q2 and turns on both transistors when driven high.

This circuit also enables a software-controlled target reset from SigmaStudio.

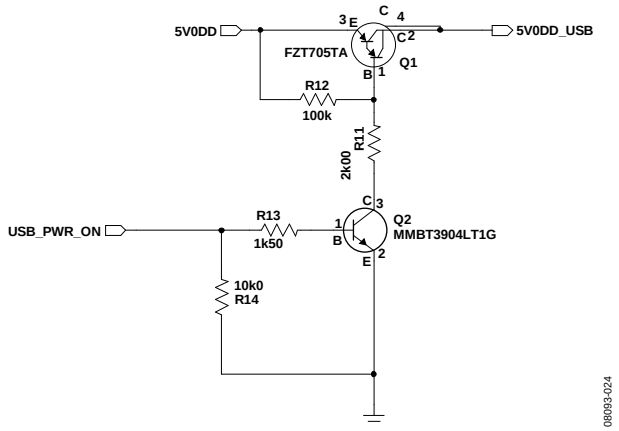

Figure 24. Target Power Switch Schematic

TARGET BOARD PROGRAMMING HEADER

To properly boot the Cypress USB microcontroller from the EEPROM, it is necessary to remove all other devices from the I²C bus. The ADG721BRMZ analog switch remains open, isolating the I²C bus from the target, until the boot process has completed.

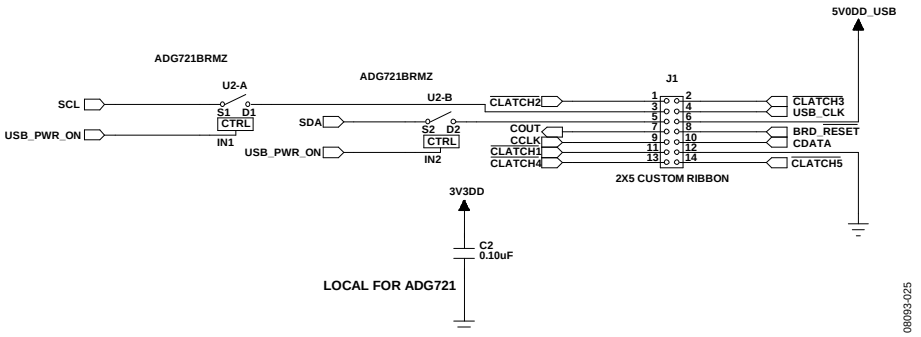

Figure 25. Target Board Programming Header Schematic

EVALUATION BOARD SCHEMATICS AND ARTWORK

SCHEMATICS

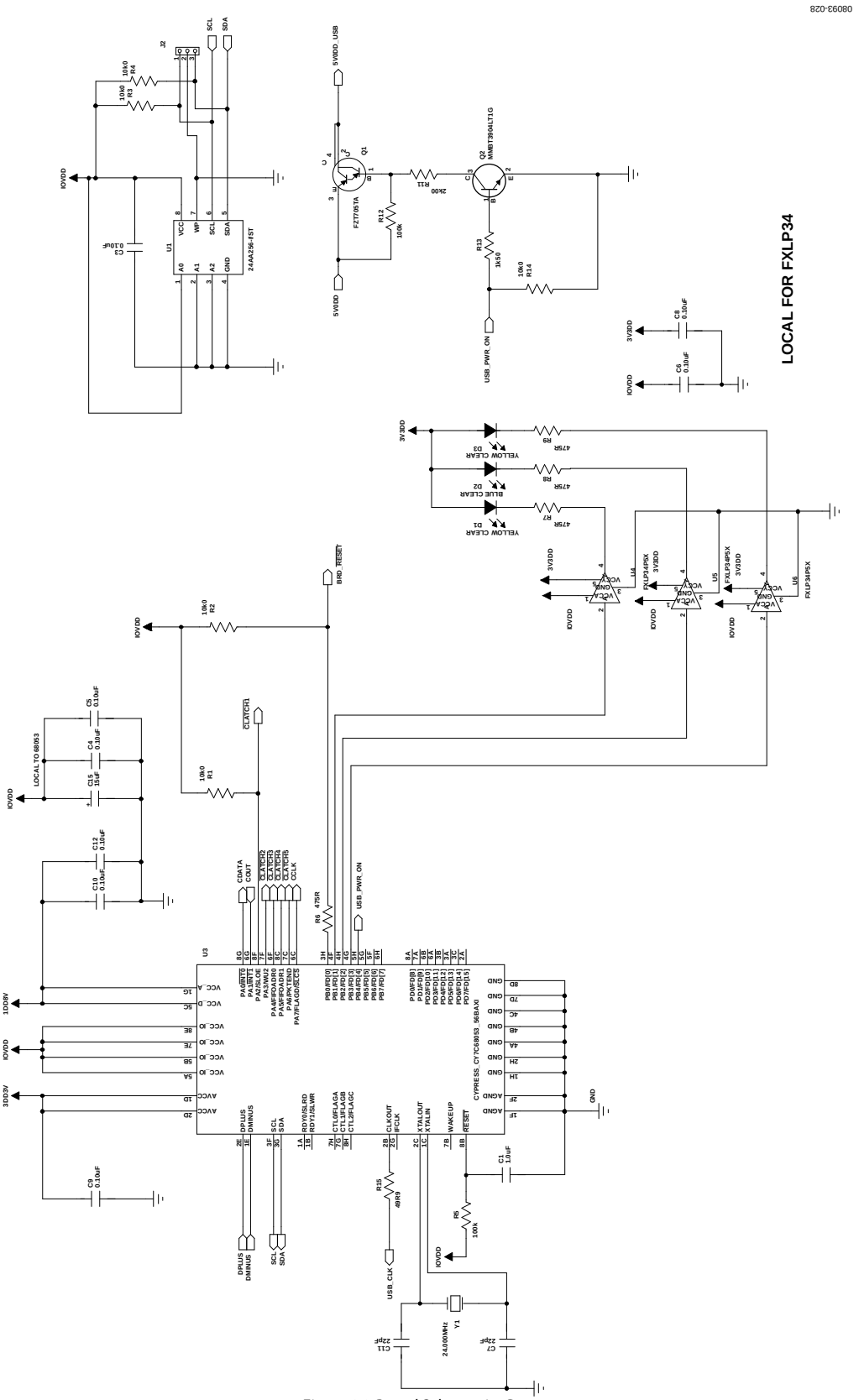

Figure 26. Board Schematics Page 1

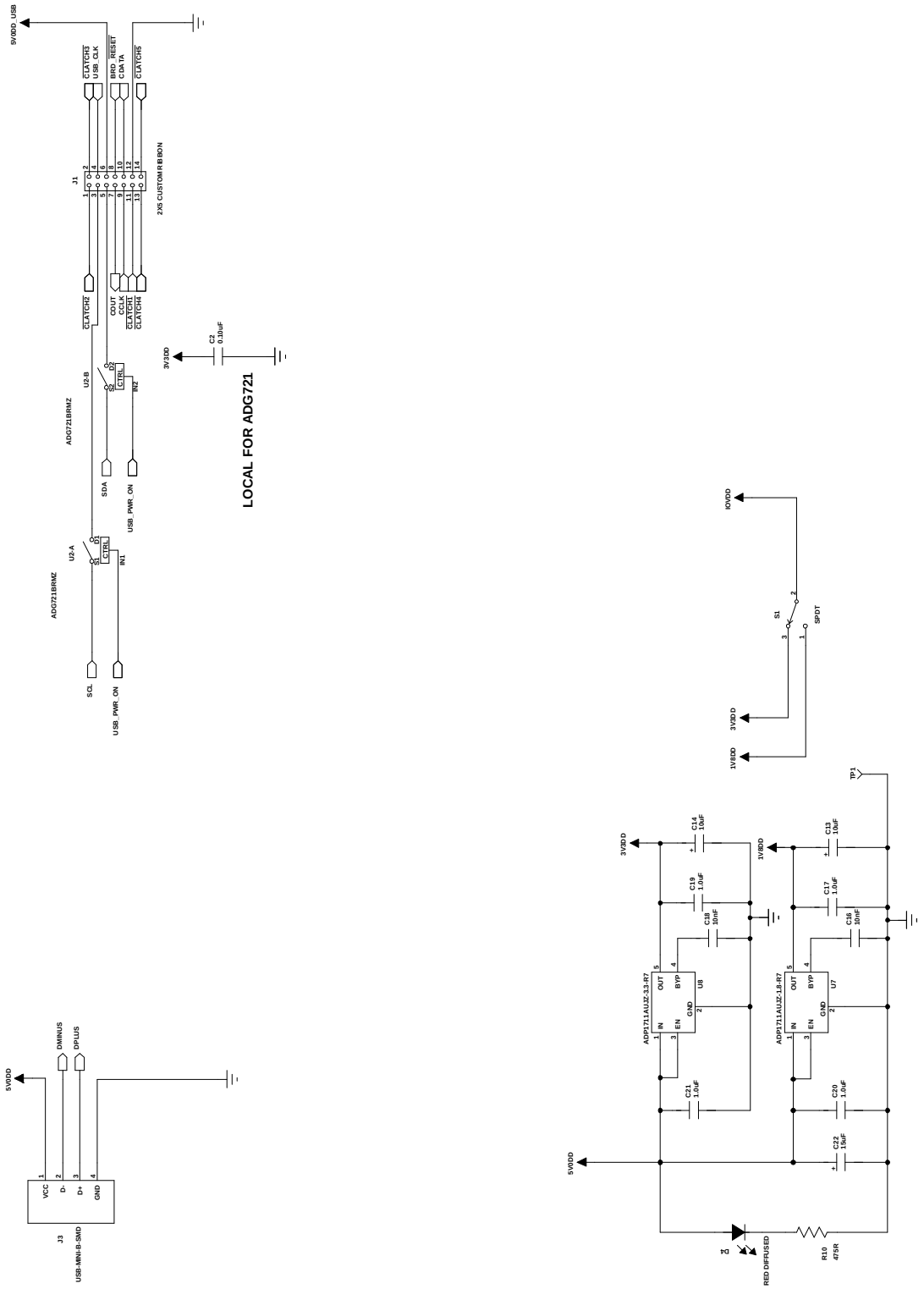

Figure 27. Board Schematics Page 2

BOARD LAYOUT

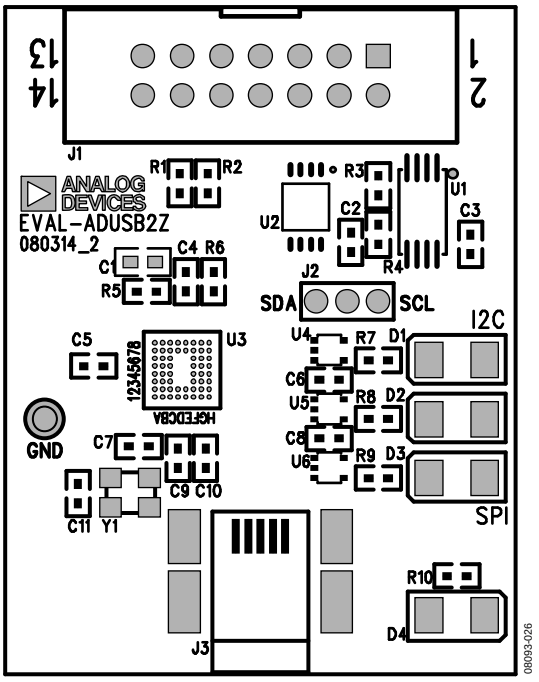

Figure 28. Board Layout—Top View

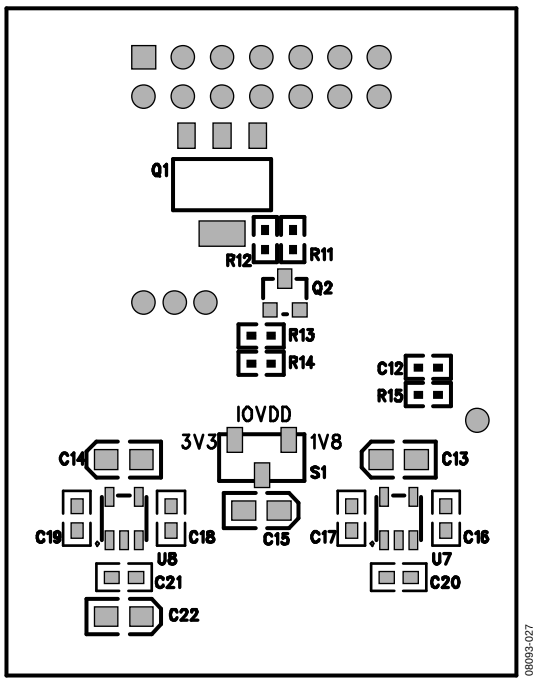

Figure 29. Board Layout—Bottom View

BILL OF MATERIALS

Table 2.

Qty	Reference Designator	Description	Manufacturer Part Number	Vendor	Vendor Order No.
5	C1, C17, C19 to C21	1.0 µF, 10%, multilayer ceramic, 16 V, X7R (0603)	EMK107B7105KA-T	Digi-Key	587-1241-1-ND
9	C2 to C6, C8 to C10, C12	0.10 µF, 10%, multilayer ceramic, 16 V, X7R (0402)	ECJ-0EX1C104K	Digi-Key	PCC13490CT-ND
2	C7, C11	22 pF, 5%, multilayer ceramic, 50 V, NP0 (0402)	GRM1555C1H220JZ01D	Digi-Key	490-1283-1-ND
2	C13, C14	10 µF, 20%, SMD tantalum capacitor, 0805, 6.3 V	TCP0J106M8R	Digi-Key	511-1447-1-ND
2	C15, C22	15 µF, 20%, SMD tantalum capacitor 0805 6.3 V	TCP0J156M8R	Digi-Key	511-1448-1-ND
2	C16, C18	10 nF, 5%, multilayer ceramic, 25 V, NP0 (0603)	C1608C0G1E103J	Digi-Key	445-2664-1-ND
2	D1, D3	LED, yellow clear, 6.0 mcd, 585 nm, 1206	SML-LX1206YC-TR	Digi-Key	67-1358-1-ND
1	D2	LED, blue clear, 25 mcd, 470 nm, 1206	SML-LX1206USBC-TR	Digi-Key	67-1701-1-ND
1	D4	LED, red diffused, 6.0 mcd, 635 nm, 1206	SML-LX1206IW-TR	Digi-Key	67-1003-1-ND
1	J1	Header, 10-way, custom ribbon cable, install centered on 14-way footprint	RCC-2184-ND	Digi-Key	RCC-2184-ND
1	J2	3-way socket, 2 mm, single row, 1 × 3	25630301RP2	Digi-Key	2563S-03-ND
1	J3	USB, mini Type B receptacle SMD	54819-0572	Digi-Key	WM17116CT-ND
1	Q1	PNP Darlington transistor, SOT223	FZT705TA	Digi-Key	FZT705CT-ND
1	Q2	NPN general-purpose transistor	MMBT3904LT1G	Digi-Key	MMBT3904LT1GOSCT-ND
4	R1 to R4	10.0 kΩ chip resistor, 1%, 63mW, thick film, 0402	MCR01MZPF1002	Digi-Key	RHM10.0KLCT-ND
2	R5, R12	100 kΩ chip resistor, 1%, 63 mW, thick film, 0402	MCR01MZPF1003	Digi-Key	RHM100KLCT-ND
5	R6 to R10	475 Ω chip resistor, 1%, 63 mW, thick film, 0402	CRCW0402475RFKED	Digi-Key	541-475LCT-ND
1	R11	2.00 kΩ chip resistor, 1%, 63 mW, thick film, 0402	ERJ-2RKF2001X	Digi-Key	P2.00KLCT-ND
1	R13	1.50 kΩ chip resistor, 1%, 63 mW, thick film, 0402	ERJ-2RKF1501X	Digi-Key	P1.50KLCT-ND
1	R14	10.0 kΩ chip resistor, 1%, 63 mW, thick film, 0402	MCR01MZPF1002	Digi-Key	RHM10.0KLCT-ND
1	R15	49.9 Ω chip resistor, 1%, 63 mW, thick film, 0402	MCR01MZPF49R9	Digi-Key	RHM49.9LCT-ND
1	S1	SPDT slide switch SMD J hook	CAS-120TA	Digi-Key	CAS120JCT-ND
1	TP1	Mini test point white 0.040 inch hole diameter, 0.10 inch × 0.020 inch	5002	Digi-Key	5002K-ND
1	U1	256 kb I^2C, CMOS serial EEPROM	24AA256-I/ST	Digi-Key	24AA256-I/ST-ND
1	U2	CMOS, low voltage, 4 Ω dual SPST switch	ADG721BRMZ	Analog Devices	ADG721BRMZ
1	U3	USB microcontroller, I^2C (3) 8-bit ports	CY7C68053-56BAXI	Arrow Electronics	CY7C68053-56BAXI
3	U4 to U6	Translator, 1-bit, unidirect SC70-5	FXLP34P5X	Digi-Key	FXLP34P5XCT-ND
1	U7	Adjustable, low dropout voltage regulator, 1.0%	ADP1711AUJZ-1.8-R7	Analog Devices	ADP1711AUJZ-1.8-R7
1	U8	Adjustable, low dropout voltage regulator, 1.0%	ADP1711AUJZ-3.3-R7	Analog Devices	ADP1711AUJZ-3.3-R7
1	Y1	Crystal, 24.000 MHz, SMT 18 pF, 3.2 mm × 2.5 mm	ABM8-24.000MHZ-B2-T	Digi-Key	535-9138-1-ND

NOTES

NOTES

NOTES

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

AN08093-0-4/10(A)