SAR_SPIM_USB Example Project
1.00

Features

- Dual SAR ADCs
- USBUART and SPI Master communication interfaces
- Exception Handling

General Description

This Example Project implements two 4-channel SAR ADCs along with SPI and USB interfaces for communicating the converted data. The ADC channel to be read is decided by the user through the USB host device. The ADC then processes the input and sends it to a SPI Slave and back to the USB host. This design also uses the LCD character display for debugging. To test this design, a second project (SAR_SPIM_USB_Test) implementing a SPI Slave is available as a PSoC® Creator™ Example Project.

Development Kit Configuration

The following configuration instructions provide a guideline to test this Example Project. For simplicity, the instructions describe the stepwise process to be followed when testing this design with the PSoC Development Kit (CY8CKIT-001) and PSoC 5 processor module, but can be generalized for the PSoC 5 Development Kit (CY8CKIT-050) as well.

1. Set LCD power jumper J12 to ON position and position jumpers for Vdd, Vdda and Vddd to be at 5V for both the main and test board.
2. Attach a 24MHz crystal (Y2) to the PSoC 5 processor module on the main development kit, along with appropriate capacitors C26 and C27 (~22pF).
3. In order to generate different voltages to test the Example Project, set up a resistor ladder on the breadboard available on the PSoC DVK (See Figure 1 and Figure 3). Use 7 resistances of 10k ohm in series, followed by a 0 ohm resistor or jumper wire to ground. Ensure that the LCD is connected to the LCD header P18 on both boards.
4. The SPI pin connections are as follows: P5[0] is SCLK, P5[1] – MOSI, P5[2] – SS, P5[3] – MISO. Connect these pins to corresponding (same) pins on the Slave board.

Alias	Name	Pin	Lock
	Pin_Ground	P0[0] OpAmp:out	☑
	Mux1_3	P0[1] OpAmp:out	☑
	\ADC_SAR_1:Bypass\	P0[2] OpAmp+	☑
	Mux1_2	P0[3] OpAmp-, DSM:ExtVref	☑
	\ADC_SAR_0:Bypass\	P0[4] OpAmp+	☑
	Mux1_1	P0[5] OpAmp-	☑
	Mux1_0	P0[6] IDAC:HI	☑
	Current_Source	P0[7] IDAC:HI	☑
	Mux0_3	P1[2]	☑
	Mux0_2	P1[4]	☑
	Mux0_1	P1[6]	☑
	Mux0_0	P1[7]	☑
	\LCD_Char:LCDPort\[6:0]	P2[6:0]	☑
	SCLK	P5[0]	☑
	MOSI	P5[1]	☑
	SS	P5[2]	☑
	MISO	P5[3]	☑
	\USBUART_1:Dp\	P15[6] SWD:IO, USB:D+	☑
	\USBUART_1:Dm\	P15[7] SWD:CK, USB:D-	☑

Figure 1. Pin connections for the SAR_SPIM_USB project

5. Ensure that the grounds of the two boards are tied together using a short wire.
6. Build the SAR_SPIM_USB project and then program the hex file onto the Master board, and repeat this for the SAR_SPIM_USB_Test project with its corresponding board. After programming is complete, disconnect the MiniProg3.
7. Connect a USB cable between the PC and USB port J9 on the main development board.
8. Reset both the master and the slave devices.

To access the PSoC via USB follow these steps:

1. Connect the PSoC DVK to the PC using a USB cable (if not already done so in step 7 above).
2. Select the SAR_SPIM_USB.inf file from the project directory, as the driver for this example once Windows asks for it.

3. Go to Start > Devices and Printers and identify COM port number associated with the project.
4. Open a terminal emulation software such as 'PuTTY' and open the COM (Serial) port number identified in previous step. Ensure that the speed is set to 9600 bps. See Figure 2.
5. The input window will open with a blank screen. Type in a channel value.

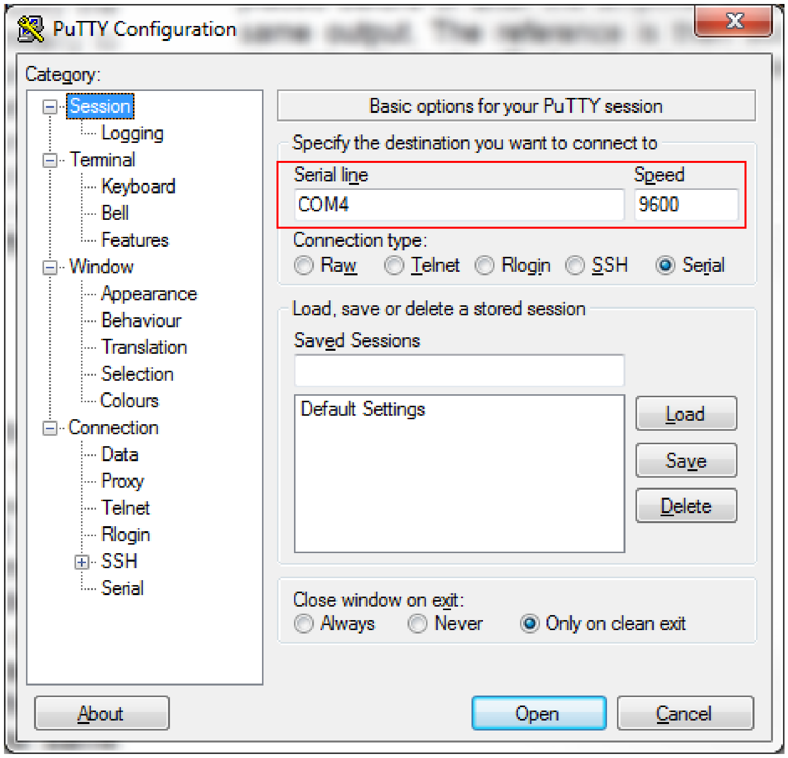

Figure 2. Putty window

Project Configuration

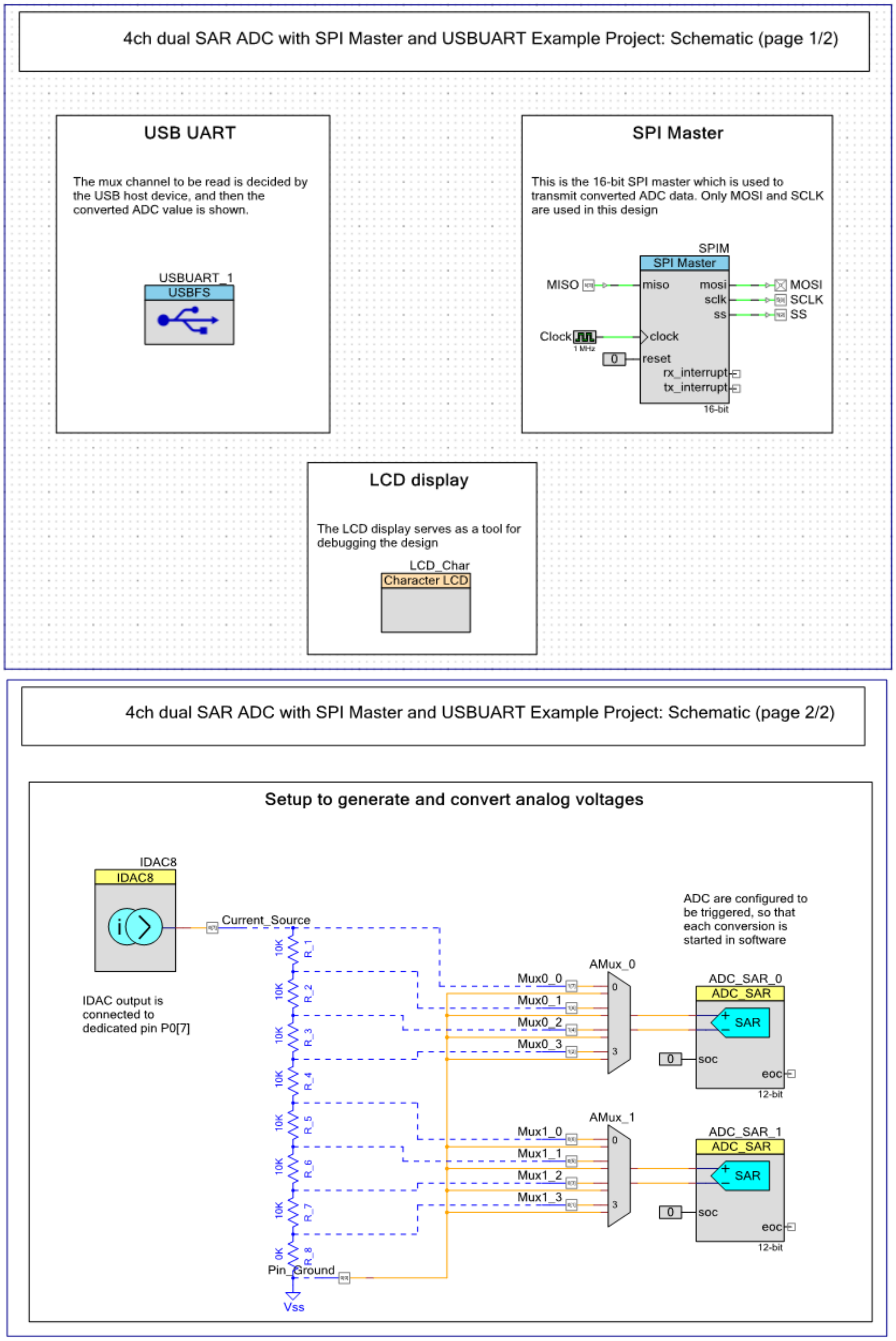

Figure 3. Main project top design schematic

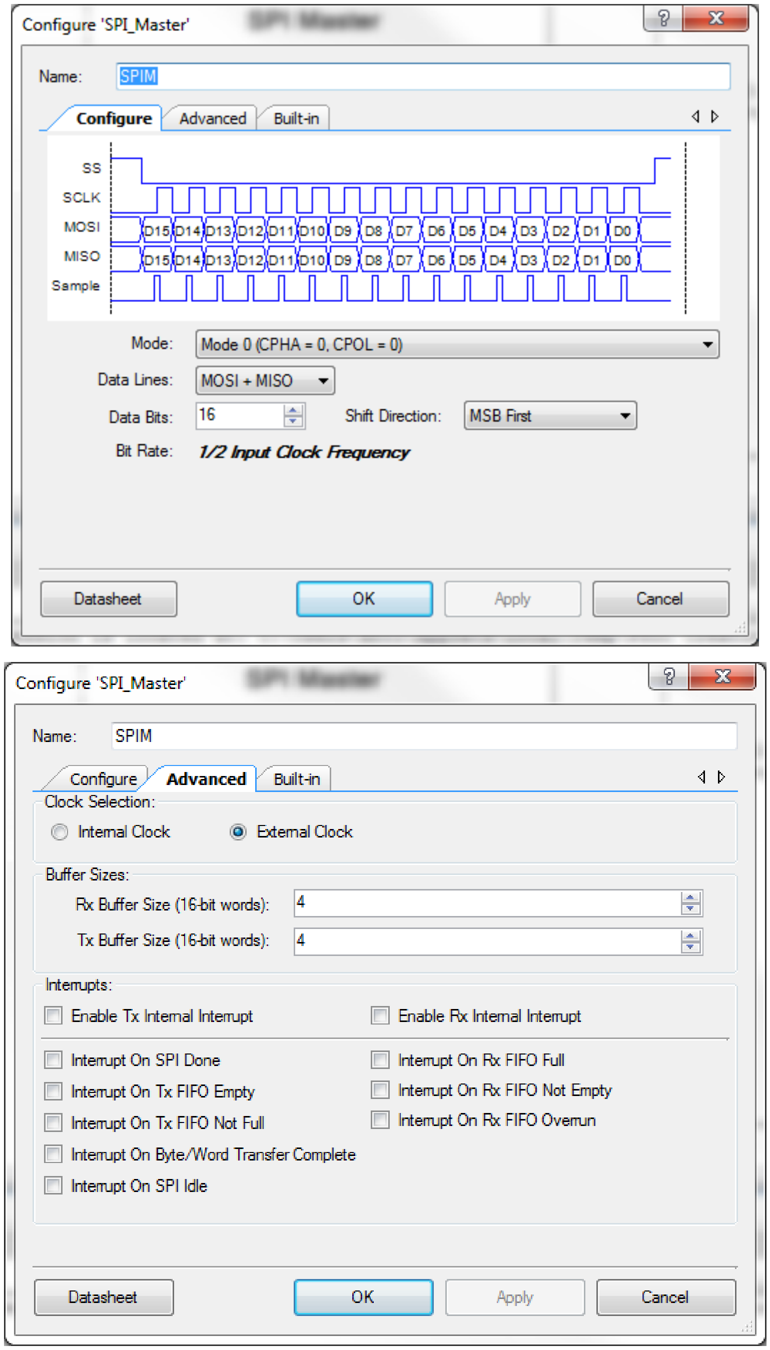

Figure 4. SPI Master configuration

The top design schematic is shown in Figure 3.

The SPI Master Full Duplex Mode macro is used for the SPI Master; the default settings are retained, except that the data bits parameter is set to 16, and an external 2 MHz clock is used. See Figure 4.

The IDAC is set to source current in the 0-31.875 µA range and initial value of 10µA. This value can be adjusted according to the input range of the ADC and the value of the resistors in the resistor ladder.

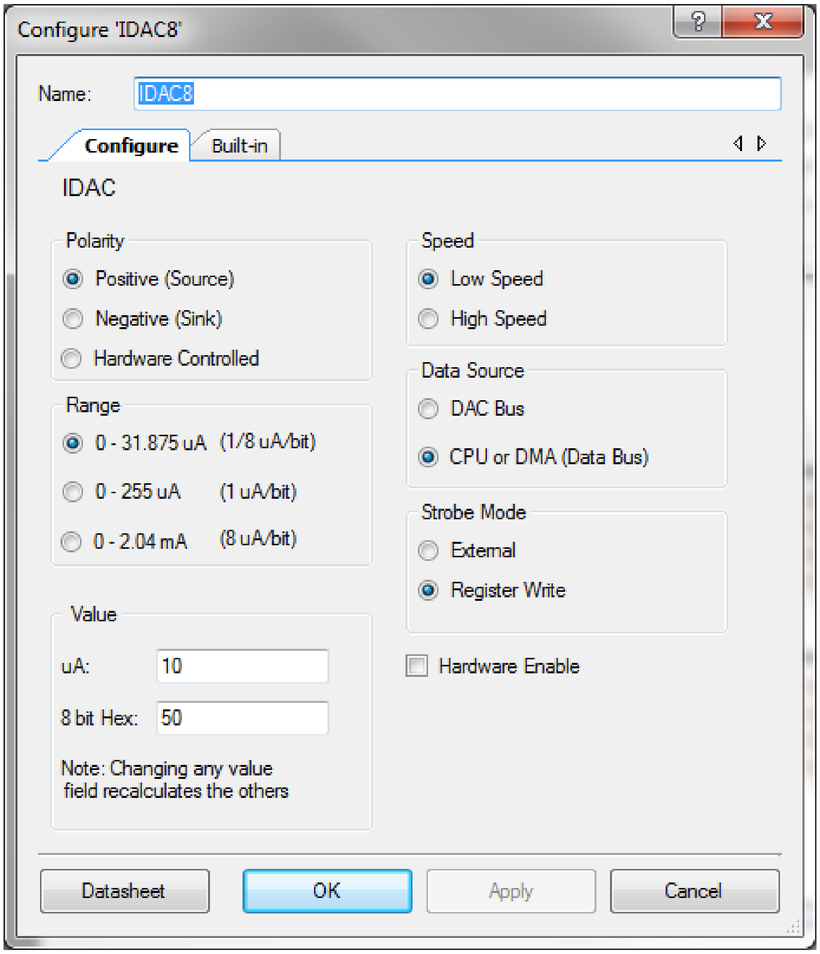

Figure 5. IDAC configuration

The analog and digital pins are retained with their default settings. The analog hardware mux is chosen to multiplex the 8 differential inputs to the SAR ADC. This facilitates arbitrary input channel selection via firmware. Note that the sampling mode for the SAR is set to 'triggered', and hence each ADC conversion needs to be started either in software using the StartConvert() API, or via a rising edge on the 'soc' input of the SAR ADC component.

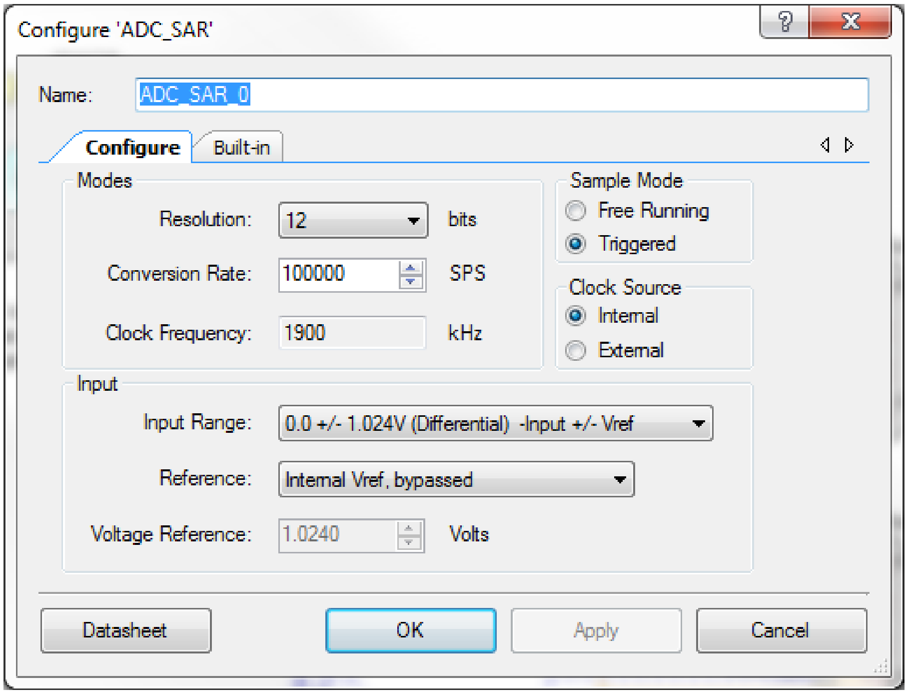

Figure 6. SAR ADC configuration

The USB component uses an **external** 24 MHz crystal oscillator (See configuration instruction 2), and requires some modifications to the ‘Clocks’ section in the design-wide resources file in PSoC Creator. These include enabling the 100 kHz ILO, setting the USB clock to IMO*2, and enabling the 24Mhz external crystal - Figure 7.

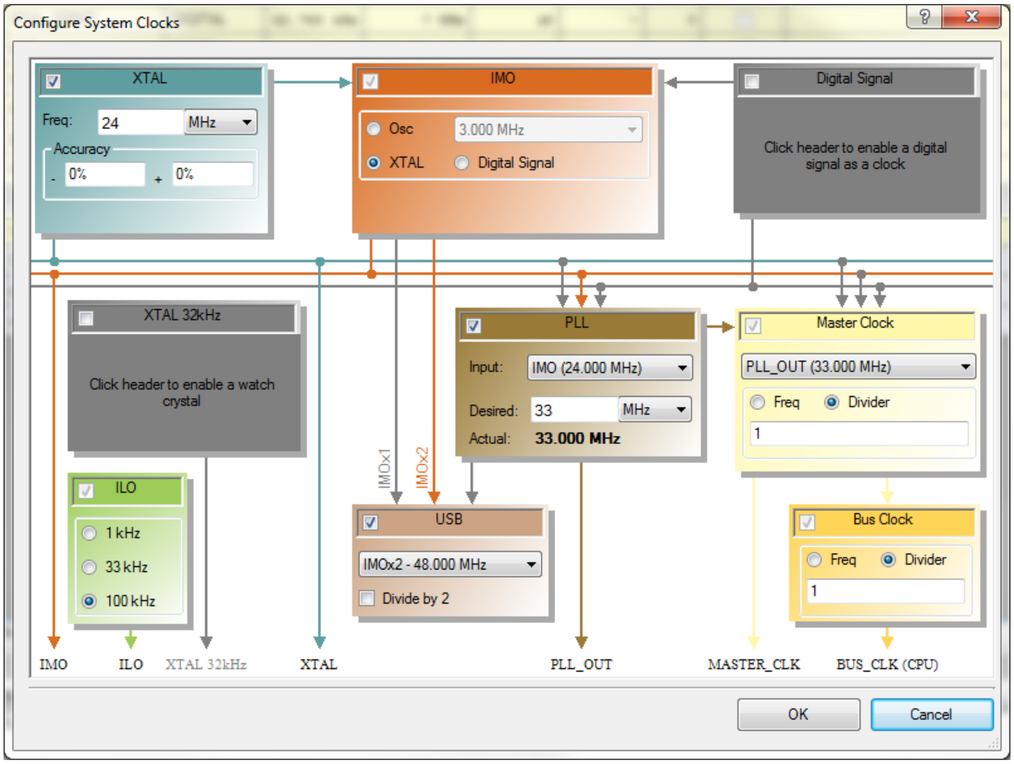

Figure 7. System clock configuration for the master project

Figure 8 and Figure 9 show the test project configuration. This project is basically a simple SPI slave, and serves to test the SPI interface of the master device.

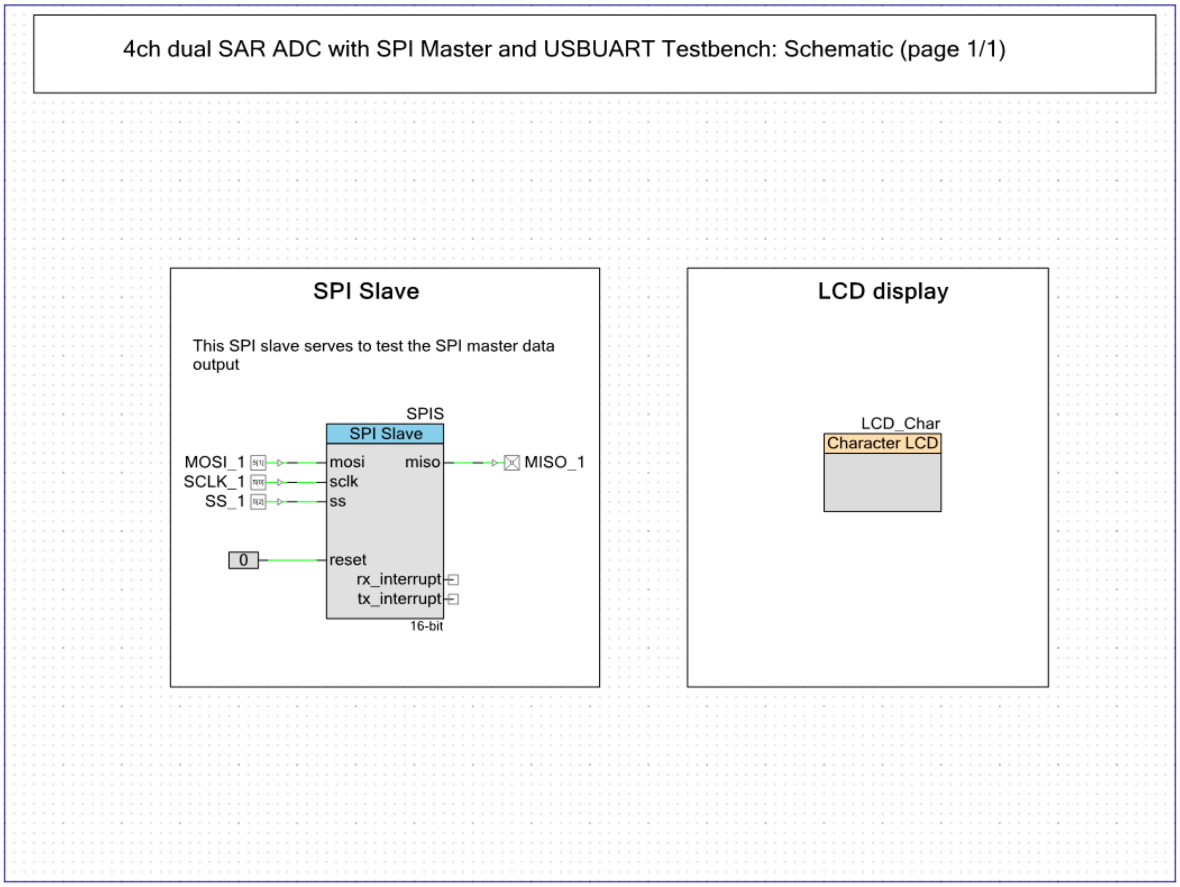

Figure 8. Top design schematic for test project

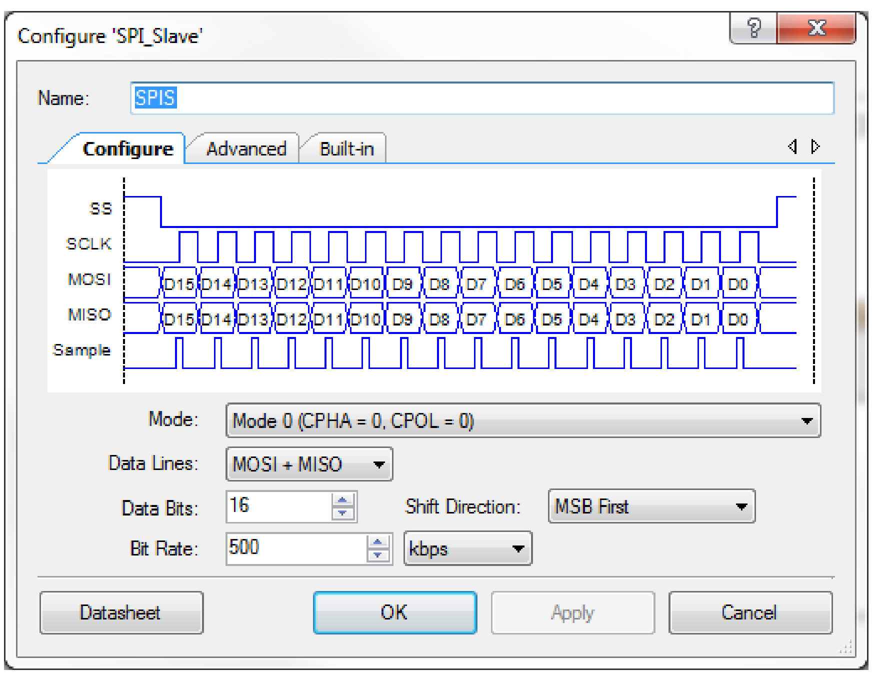

Figure 9. SPI Slave component configuration

Project Description

The analog voltages generated by the IDAC and resistive ladder are input to the two 4ch analog hardware muxes. The USB host device chooses which channel to read. This voltage is converted by the ADC, and transmitted over SPI using the SPI master and also back to the USB host device. This digital value is also displayed on the Character LCD.

The receiver board has a pre-configured SPI slave which waits for data from the SPI master. When data arrives, this is displayed using the Character LCD. The functionality can be verified by checking the data displayed on the computer screen, and the main and test board LCDs (at the same time).

Expected Results

The user should be able to select the ADC channel to be read via the terminal emulation software. The converted value should then be displayed on the terminal emulation window as well as the Character LCD on the Master and Slave boards. This value is the hexadecimal representation of the analog voltages from the resistor ladder.

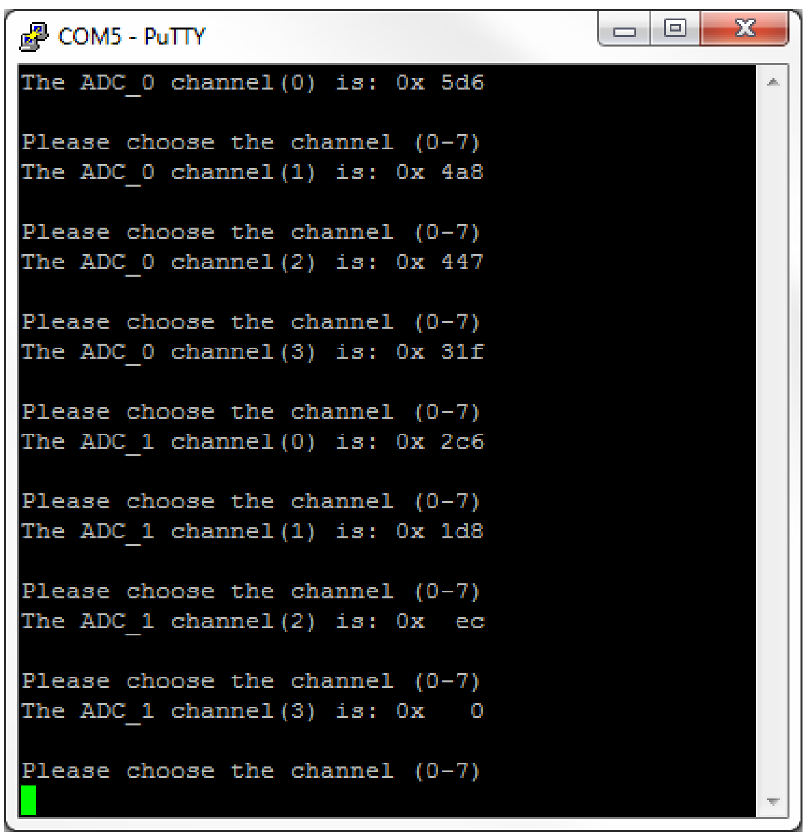

Figure 10. Expected output on terminal window

Related Material

Example Projects

- ADC_16Channel
- DelSig_I2CS
- DelSig_SPIM
- SAR_SPIM_USB
- ADC_DMA_VDAC

Application Notes

- AN57294 - USB 101: An Introduction to Universal Serial Bus 2.0
- AN57473 - PSoC® 3 / PSoC 5 USB HID Fundamentals with Mouse and Joystick
- AN56377 - PSoC® 3 and PSoC 5 USB Transfer Types

Component Datasheets

- Full Speed USB (USBFS) 2.12
- ADC Successive Approximation Register (ADC_SAR) v1.71
- Serial Peripheral Interface (SPI) Master 2.21
- Serial Peripheral Interface (SPI) Slave 2.20

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com