ADC_DelSig example project
2.20

Features

- Project uses Default Differential mode
- Continuous conversion mode with 16-bit resolution
- Reference used is internal reference

General Description

This example project demonstrates the operation of Delta Sigma ADC in differential mode.

Development kit configuration

1. This project is written with a 2X16 LCD display as the one available in the Cypress kit CY8CKIT-001.
2. Build the project and program the hex file on to CY8C3866AXI-040 using MiniProg3.
3. Connect pins as described below and power cycle the device.
4. Observe the ADC output on the LCD.

Project configuration

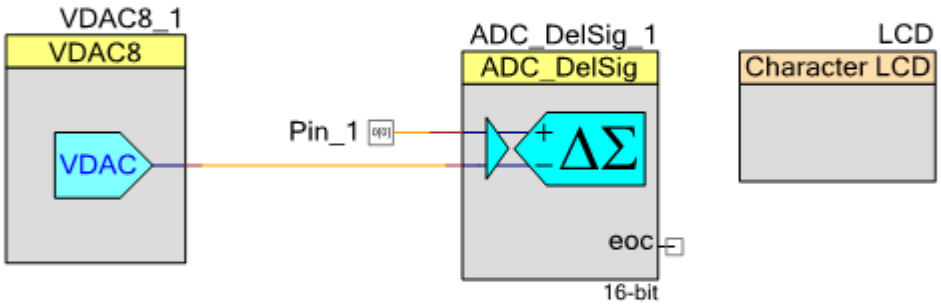
The example project consists of ADC_DelSig, VDAC and Char LCD components. The top design schematic is shown in Figure 1. The Character LCD component is used for displaying the ADC output. The VDAC component is used to supply analog input to the negative input of the ADC.

Test Setup:

1) Positive terminal of ADC is connected to the analog pin which is mapped to P0[0] of CY8CKIT-001. Connect the analog voltage from variable resistor to P0[0].

2) VDAC is connected to negative terminal of ADC. Value 251 is written to VDAC data register. This is done by setting Value parameter to a value 251 in the VDAC configure window. This results in output voltage of 1000 mV from VDAC output terminal which is connected to ADC negative input terminal.

3) LCD is used to print the result(converted digital value for the corresponding analog value). LCD is mapped to P2[6:0] of CY8CKIT-001. LCD displays the digital value for the corresponding input value to ADC.



Procedure :

1. Build the project and program the hex file on to the target device.

2. Power cycle the device and observe the results on the LCD.

3. The digital value is displayed in the LCD module which corresponds to resultant input analog value given to input terminals of ADC.

4. Vary the input analog voltage by using variable resistor and observe the digital value on the LCD. If the effective input value is 0 volts then digital output displayed on the LCD is -1000 mV. If the effective input voltage is 1.024V, then output displayed on the LCD is 0 mV.

Figure 1. Top design schematic.

The Character LCD and VDAC use their default configurations. The ADC is configured in the default differential mode with 16-bit Continuous conversion mode. The ADC_DelSig component configuration window is shown below in figure 2.

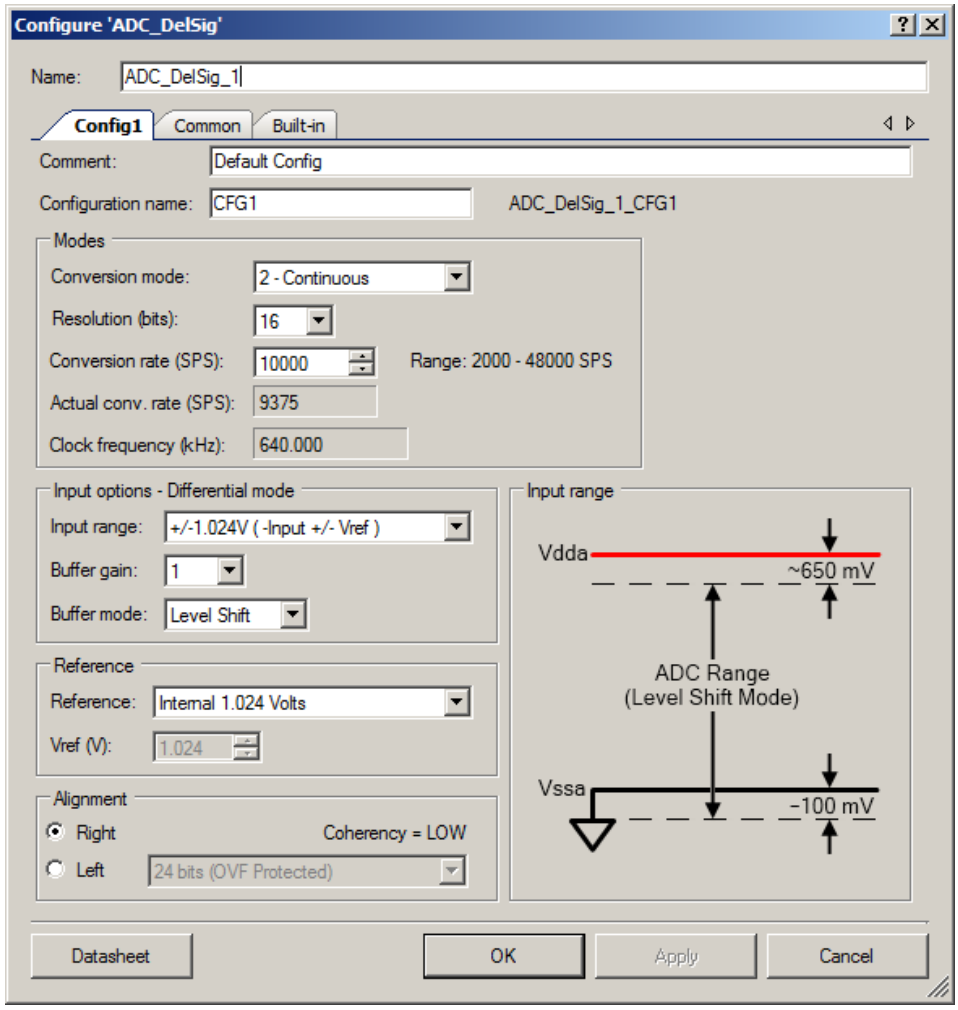


Figure 2. ADC_DelSig Component Configuration.

Project description

In the main function all components are started. For the proper usage of the Character LCD and VDAC components, please refer to the corresponding component datasheets.

ADC_DelSig is configured in the default differential mode. ADC_DelSig uses continuous conversion mode to convert the input analog voltage. ADC_DelSig_IsEndConversion() API is

used to check the end of conversion. The converted digital value is read using ADC_DelSig_GetResult16() API and result is displayed on the LCD.

Expected Results

The LCD should display the converted output value which is equivalent to the effective analog input voltage given to the input terminals of ADC component.

© Cypress Semiconductor Corporation, 2009-2013. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.