CE95273 - Peripheral-to-Peripheral Data Transfer Using DMA with PSoC® 3 and PSoC 5LP

Objective

This code example demonstrates peripheral-to-peripheral data transfer using the PSoC direct memory access (DMA) controller. Analog-to-digital converter (ADC) sampled data is transferred to a digital-to-analog converter (DAC) output register.

Overview

This code example shows how the PSoC 3 and PSoC 5LP DMA controller is used to transfer data to and from various peripherals, including the ADC and DAC, without using the CPU. The ADC continually samples an analog input pin. At the end of each sample, it generates a DMA request signal. The DMA transfers data from an ADC output data register to a DAC input data register.

The ADC resolution is configured to be 8 bits; however, it can produce a 16-bit signed value to handle analog input overflow and underflow. The DAC is 8-bit. The DMA transfers the 8 least significant bits (LSb) of data from the ADC to the DAC.

The firmware continuously displays the ADC signed 16-bit register and the 8-bit DAC data register on a kit LCD for monitoring. You can also monitor the ADC input and DAC output pins with a multimeter or oscilloscope.

Requirements

Tool: PSoC Creator™ 3.1 or later

Programming Language: C (DP8051 Keil 9.51, ARM GCC 4.8.4, and ARM MDK compilers)

Associated Parts: All PSoC 3 and PSoC 5LP parts

Related Hardware: CY8CKIT-030, CY8CKIT-050

Design

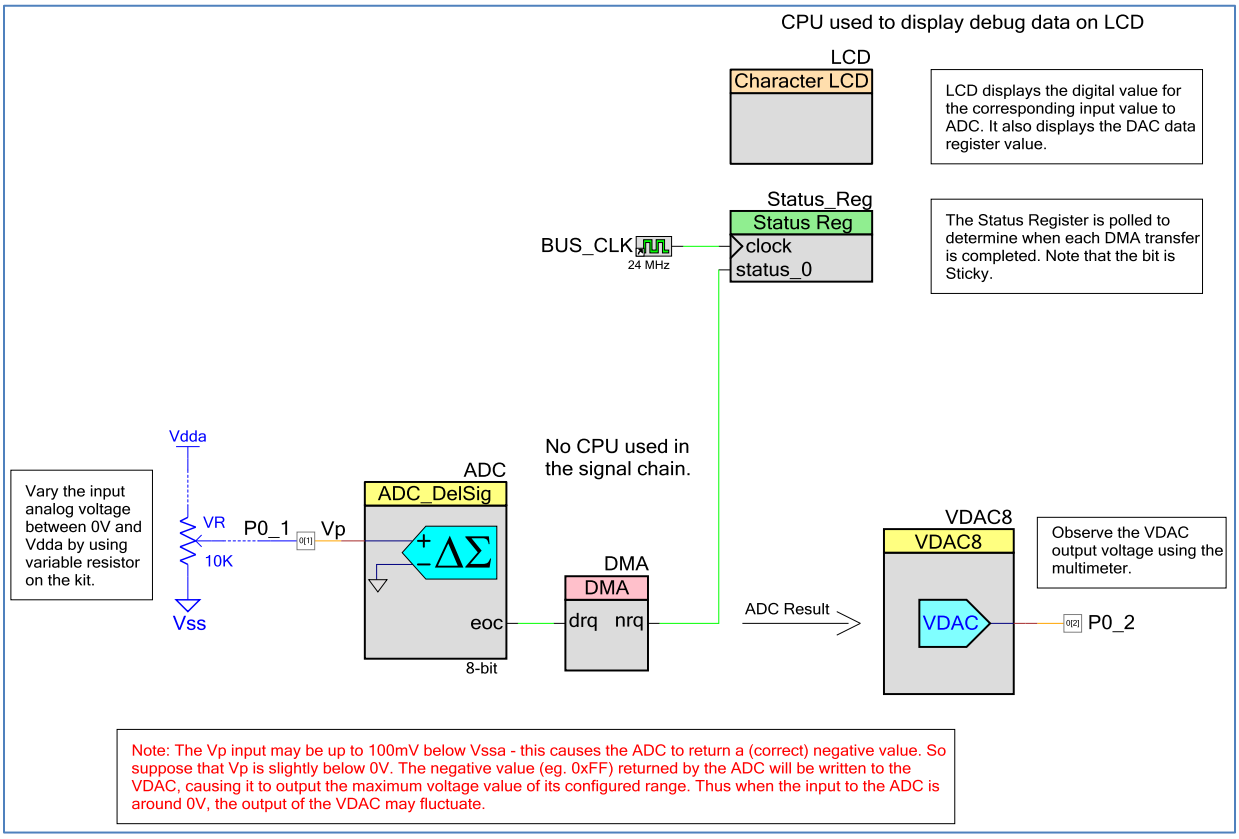
Figure 1 shows the PSoC Creator schematic design of the code example.

The example project features the following:

- Delta Sigma ADC Component for sampling an input voltage. The input voltage to the ADC may be adjusted between 0 V and Vdda by using a variable resistor on the CY8CKIT-030, CY8CKIT-50, or an external analog voltage source.
- DMA Component triggered by the ADC end-of-conversion (EOC) terminal. For a detailed description of PSoC 3 and PSoC 5LP DMA, see the DMA-specific application notes listed in Related Documents.
  - The source address of the DMA's transfer descriptor points to the ADC sample register.
  - The destination address of the DMA's transfer descriptor points to the DAC data register.
  - The DMA transfers one byte on every burst.
  - The DMA generates a pulse on the nrq terminal at the end of each transfer.
  - A sticky status register captures the completion of a DMA transfer.
- Eight-bit DAC Component configured to drive a voltage signal between 0 V and 1.020 V.
- A display Component (Character LCD) to show the ADC sample value and the DAC data register.
  - The first LCD line shows the raw 16-bit ADC output.
  - The second LCD line shows the 8-bit DAC input value.
  - The LCD is updated when a DMA transfer is complete.

Note that a status register, with the sticky bit option set, is used to detect the end of a DMA transfer. The firmware polls the status register to detect the DMA transfer completion and then updates the LCD with the ADC and DAC data.

Figure 1. PSoC 3 and PSoC 5LP Peripheral-to-Peripheral Data Transfer Using DMA



Design Considerations

The ADC, even when it is configured with 8 bits, can produce a 16-bit value to handle voltages below Vssa (underflow). It reports voltages below Vssa as 16-bit negative values. Writing these values to the DAC causes it to output about the maximum DAC voltage. Thus, when the input to the ADC is about 0 V, the DAC output may fluctuate.

Similar behavior occurs when the measured voltage is higher than 1.020 V (overflow). The ADC returns a 16-bit value greater than 255. When the LSb is written to the DAC, the output fluctuates at approximately the minimum DAC voltage. Thus, when the input to the ADC is about 1.020 V, the DAC output may also fluctuate.

Hardware Setup

For the basic kit board setup, see the corresponding kit user guide.

This example project is designed to run on the CY8CKIT-030 (by default) and CY8CKIT-050. To use the latter, change the project's device from CY8C3866AXI-40 to CY8C586AXI-LP035 with the Device Selector (see demonstration video).

The only instruction is to connect a jumper wire between the P0[1] pin and the VR pin from the P6 connector. You use a multimeter to observe the DAC output pin voltage (P0[2]).

Software Setup

No special software setup is required. You can use all supported compilers, with any optimization.

Components

Table 1 lists the PSoC Creator Components used in this example, as well as the resources used by each one.

Table 1. PSoC Creator Components

Component	Hardware Resources
Delta Sigma ADC	PSoC 3 or PSoC 5LP Delta Sigma ADC
DMA	1 DMA channel, 1 transaction descriptor
Status Register	UDB (1 status register)
Clock	Bus clock
Character LCD	7 pins
VDAC8	1 DAC block
Pin	2 analog pins

Parameter Settings

Table 2 lists the parameter settings for the PSoC Creator Components used in the code example. Only the parameters that vary from the default are listed.

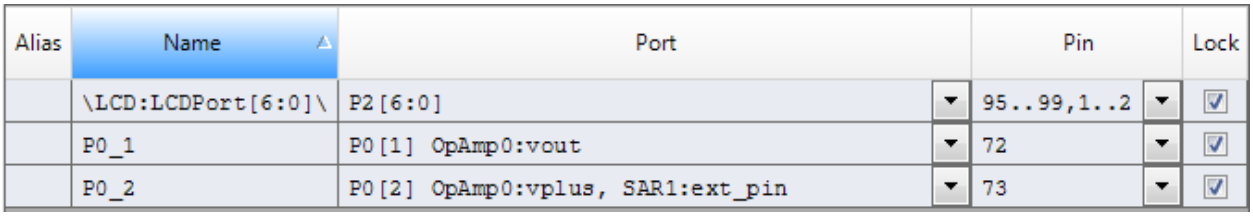
Table 2. PSoC Creator Component Parameter Settings

Component	Non-Default Parameter Settings
Delta Sigma ADC	Conversion mode: Continuous Input range: Vssa to 1.024 V Buffer mode: Bypass buffer Input mode: Single ended Resolution: 8 bits
DMA	Hardware request: Level
Status Register	Sticky mode
Clock	Source: BUS_CLK
Character LCD	None
VDAC8	None
Pin	ADC input pin: Analog VDAC output pin: Analog

Design-Wide Resources

Figure 2 shows the pin assignments for the examples. No other Design-Wide Resources need to be changed from their default settings.

Figure 2. Pin Assignments for PSoC 3 and PSoC 5LP Example



Operation

After building and installing the code example in the appropriate kit, test the code example as follows:

- Use the potentiometer to change the voltage at the ADC input pin. Observe the character LCD.
  - Observe that the top row displays “ADC Output: XXXX.” The 16-bit hexadecimal value represents the 8-bit ADC sample value. Confirm that the values below zero show “0xFFFF” and the values above 1.020 V show a number greater than 0x00FF.
  - Observe that the bottom row displays “DAC Input: XX.” The 8-bit hexadecimal value represents the 8 LSb of the ADC sample.
- Use a multimeter to measure the output voltage at the P0[2] pin. Confirm that the value is 1.020 V when the DAC input is 0xFF and 0 V when the DAC input is 0x00. Confirm that the numbers between 0x00 and 0xFF are linearly proportional.

Upgrade Information

N/A

Related Documents

Table 3 lists the relevant application notes, code examples, knowledge base articles, device datasheets, and Component datasheets.

Table 3. Related Documents

Application Notes		
AN61102	PSoC 3 and PSoC 5LP – ADC Data Buffering Using DMA	Shows how to configure the DMA to buffer the ADC data
AN84783	Accurate Measurement Using PSoC 3 and PSoC 5LP Delta-Sigma ADCs	Shows how to increase the accuracy of measurements using the 20-bit Delta-Sigma ADC
AN68403	PSoC 3 and PSoC 5LP Analog Signal Chain Calibration	Shows how gain and offset errors can be eliminated in the entire signal chain
AN52705	PSoC 3 and PSoC 5LP – Getting Started with DMA	Provides an introduction to DMA in PSoC 3 and PSoC 5LP
AN84810	PSoC 3 and PSoC 5LP Advanced DMA Topics	Discusses several advanced PSoC 3 and PSoC 5LP DMA topics and design challenges
AN64275	PSoC 3 and PSoC 5LP : Getting More Resolution from 8-bit DACs	Discusses several methods to increase the resolution of the DACs available in the PSoC 3 and PSoC 5LP families
AN69133	PSoC 3 / PSoC 5LP Easy Waveform Generation with the WaveDAC8 Component	Shows how to generate continuous waveforms that require no CPU overhead
Code Examples		
CE95271	Delta Sigma ADC in Differential Mode with PSoC 3/5LP	
CE95274	SAR ADC and PrISM with PSoC 5LP	
CE95276	Sequencing SAR ADC with PSoC 5LP	
CE95277	Delta Sigma ADC in single-ended mode with PSoC 3/5LP	
CE95299	Delta Sigma ADC using 16 multiplexed single-ended inputs with PSoC 3/5LP	
CE95309	Dithered Voltage DAC with PSoC 3/5LP	
CE95316	Filter from ADC to VDAC using DFB with PSoC 3/5LP	
CE95317	Filter from ADC to VDAC using DFB in polling mode with PSoC 3/5LP	

CE95326	8-Bit Current Digital to Analog Converter (IDAC8) with PSoC 3/5LP
CE95397	Voltage DAC with PSoC 3/5LP
CE95402	Generating a Sine wave with the WaveDAC8 Component with PSoC 3/5LP

Knowledge Base Articles	
ADC Channel Multiplexing	Demonstrates how to multiplex analog inputs to DelSig ADC in PSoC 3 and PSoC 5LP

PSoC Creator Component Datasheets	
Delta Sigma ADC	Provides a low-power, low-noise end for precision measurement applications
8-Bit VDAC	8-bit output voltage DAC
Direct Memory Address	Allows data transfers to and from memory, Components, and registers
Status Register	Allows firmware to read digital signals
Character LCD (CharLCD)	Contains a set of library routines that enable the use of one-, two-, or four-line CD modules that follow the Hitachi 44780 standard 4-bit interface
Clock	Creates local clocks and allows connection to system and design-wide clocks
Pins	Controls interface with physical I/O port pins
External Library	Provides a way to include Components external to the PSoC device—resistors, capacitors, transistors, inductors, switches, and so on—on a PSoC Creator schematic

Device Documentation	
PSoC 3 datasheets	PSoC 3 Technical Reference Manuals
PSoC 4 datasheets	PSoC 4 Technical Reference Manuals
PSoC 5LP datasheets	PSoC 5LP Technical Reference Manuals

Development Kit (DVK) Documentation
PSoC 3 and PSoC 5LP kits
PSoC 4 kits

Document History

Document Title: CE95273 – Peripheral-to-Peripheral Data Transfer Using DMA with PSoC® 3 and PSoC 5LP

Document Number: 001-95273

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	4743530	RLOS	04/29/2015	New spec
*A	5739925	AESATP12	05/25/2017	Updated “PSoC 4 Kit” link. Updated logo and copyright.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP| PSoC 6

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

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