Software Control of the ADS8364

Tom Hendrick

Data Acquisition Applications

ABSTRACT

The ADS8364 16-bit parallel output analog-to-digital converter has a number of features that allow for an easy interface to a variety of digital signal processors (DSP) from Texas Instruments. This application note focuses on configuring, sampling and converting analog data presented to the ADS8364 ADC using software control with the TMS320C6711™ DSP starter kit. In this application, McBSPa is used for general-purpose I/O, while the parallel data bus accepts data from the ADC via $\overline{RE}$, $\overline{WE}$ and $\overline{CSa}$. The code developed for this application note shows how to use the ADC's software control features by configuring the ADS8364 in cycle mode using the end of conversion (EOC) signal as an interrupt source to the DSP host. Project collateral discussed in this application note can be downloaded from the following URL: http://www.ti.com/lit/zip/SLAA155.

Contents

List of Figures

List of Tables

TMS320C6711 is a trademark of Texas Instruments.

1   Introduction

The ADS8364 is a six channel, simultaneous sampling, 16-bit parallel ADC. The device features a chip select (CS), input clock (CLK), parallel data input (D [0:15]), and flexible control signals that can interface directly to the C6000$_M$ family of DSPs.

The ADS8364 operates from 5-V analog (AVdd) and digital (DVdd) supplies. The device also incorporates an internal buffer that can be powered from the same 3.3-V supply as the DSP. The buffer voltage (BVdd) allows direct interfacing to 3 or 5-V systems, eliminating the need to level shift the data and control lines.

The sample code associated with this application note was developed with *Code Composer Studio*™ V2.10 on the C6711DSK platform using the ADS8364EVM. This simple code example demonstrates how to configure the DSP, initialize the data converter, and process an interrupt (via the $\overline{\text{EOC}}$ pin) from the data converter. The code is available for download from the Texas Instruments website www.ti.com. Search for the application report (SLAA155) to obtain a link to the associated code file in zip format.

The following block diagram is representative of how the ADS8364 would be connected in an actual system using software control with the TMS320C6711 DSP.
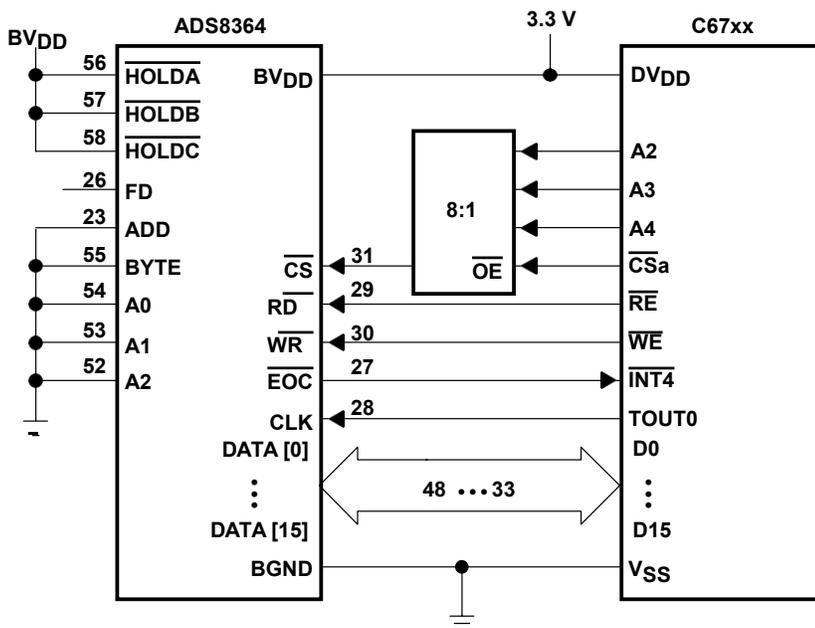

Figure 1. System Block Diagram

2   The ADS8364 EVM Interface

The ADS8364 can operate from a maximum clock frequency of 5 MHz. The sample/conversion process is completed within 20 conversion clock cycles. All six channels of the ADS8364 can be sampled/converted simultaneously, providing a maximum 250 KSPS throughput rate. In this application note, the ADS8364 was operated from a 4-MHz clock, providing a maximum throughput of 200 KSPS per channel.

C6000 and Code Composer Studio are trademarks of Texas Instruments.

The EVM uses four address lines to access the data converter. The user has the option of selecting A2 through A5 (default settings—W7 closed), or A14 through A17 (W7 open). The lower address lines control the A0, A1, and A2 pins of the ADC, while the upper most address line is sent through a single gate inverter to act as chip select ($\overline{CS}$).

In order to show the flexible interface of the ADS8364, one of the multichannel buffered serial ports (McBSP) of the TMS320C6711 is used as a general-purpose I/O port. The McBSP is used to control the $\overline{HOLDx}$ pins, the ADD pin and the RESET pin. The first data (FD) signal is sent to the McBSP as an input to indicate when data from channel A1 is present.

The DSK's read enable ($\overline{RE}$), write enable ($\overline{WE}$), and daughtercard chip select (DC_$\overline{CS}$a) are all used in the parallel interface to read data from; as well as write commands to, the ADC. The lower 16 data lines from the DSK are connected directly to the ADC aligned LSB to LSB.

Table 1 shows the connections incorporated on the evaluation module between the ADS8364 and the TMS320C6711 DSK.

Table 1. ADS8364EVM to the C6711DSK

EVM CONNECTOR/ PIN NO.	C6711 DSK CONNECTOR/PIN NO.	ADC PIN NO.	SIGNAL DESCRIPTION
J11.7–10	J1.7–10	31,56, 57, 58	Upper address selection via U10 and U11
J11.23–26	J1.23–26	31,56, 57, 58	Lower address selection via U10 and U11
J11.53–60	J1.53–60	33–40	Upper data byte [D15..D8]
J11.63–70	J1.63–70	41–48	Lower data byte [D7..D0]
J11.73	J1.73	29	ADC input – $\overline{RD}$; DSK output – DC_ARE#
J11.74	J1.74	30	ADC input – $\overline{WE}$; DSK output – DC_AWE#
J11.78	J1.78	N/A	Controls $\overline{OE}$ of address MUX
J12.33	J3.33	52	ADC input – ADD; DSK output – DC_CLKXa
J12.35	J3.35	57	ADC input – $\overline{HOLDB}$; DSK output – DC_FSXa
J12.36	J3.36	51	ADC input – $\overline{RST}$; DSK output – DC_DXa
J12.39	J3.39	56	ADC input – $\overline{HOLDA}$; DSK output – DC_CLKRa
J12.41	J3.41	58	ADC input – $\overline{HOLDC}$; DSK output – DC_FSRa
J12.42	J3.42	26	ADC output – FD; DSK input – DC_DRa
J12.45	J3.45	28	ADC input – CLK; DSK output – TOUTa (optional)
J12.53	J3.53	27	ADC output – $\overline{EOC}$; DSK input – EXT_INT4

3 Setting Up the TMS320C6711 DSK

The following section provides details on the assignment of the parallel data and communication pins to the ADS8364 device.

3.1 Set Up McBSPa GPIO/Pin Functions

In order to show the flexible interface of the ADS8364, the McBSPa port on the C6711 DSK is used as a general-purpose I/O port. This is done as a matter of convenience, since there are no serial devices on this EVM. A true application using the ADS8364 wires these control signals to a fixed logic level, as shown in the system block diagram.

Setting up McBSPa as a GPIO port is a fairly straightforward task—simply clear the associated sample rate generator and port control registers (SRGR and SPCR), then set the pin control register (PCR) for the desired functions. The EVM uses CLKXa, CLKRa, FSXa, FSRa, and DXa as general-purpose outputs, while using DRa as a general-purpose input (see Table 1). To achieve the setup shown in Figure 1, the $\overline{\text{HOLDx}}$ lines need to be set, the ADD line needs to be cleared, and FD needs to be set as an input. This is accomplished by writing 0x00003F0F to the SP1_PCR register during the DSK initialization.

3.2 Set Up the Conversion Clock

The DSK's internal timer, TOUT0, can be used as the conversion clock source for the ADS8364EVM. For the purpose of this application note, the clock is set for 50/50 duty cycle and a division multiple of 16. This provides approximately a 4.5 MHz conversion clock to the converter. An external conversion clock can also be applied through BNC connector (J9) and jumper W15. Note: the maximum conversion clock for the ADS8364 is 5 MHz.

3.3 Chip Select ($\overline{\text{CS}}$), Read ($\overline{\text{RD}}$) and Write ($\overline{\text{WR}}$)

Chip select to the ADS8364 is an active-low input signal. When $\overline{\text{CS}}$ is high, the parallel data pins are in a high-impedance state. When CS is low, the parallel data lines reflect the current state of the output buffers. The ADS8364EVM uses the C6711 DSK's DC_$\overline{\text{CS}}$a signal as a master chip select for the board. DC_$\overline{\text{CS}}$a controls the output enable ($\overline{\text{OE}}$) of a 4-bit 1-of-2 FET multiplexer, which feeds the address bus of the DSP to the ADC. Jumper W7 controls address line selection—A2 through A4 (default) or A14 through A17.

Since the ability to select two different address spaces is implemented in the EVM's design, the MSB of the address space is inverted and used as the ADS8364 chip select when writing to or reading from the ADC's base address. The three additional address lines control the A0–A2 hardware pins as shown in Table 2. When the EVM base address is selected, the A0–A2 pins of the ADS8364 are pulled to ground through 10 K$\mathbf{\Omega}$ resistors.

Table 2. ADS8364EVM Address Definitions

LOWER ADDRESS (W7 CLOSED)	COMMAND/FUNCTION	UPPER ADDRESS (W7 OPEN)
0xA0000020	ADC Base Address—enable CS, select CH_A0	0xA0020000
0xA0000024	Select CH_A1	0xA0024000
0xA0000028	Select CH_B0	0xA0028000
0xA000002C	Select CH_B1	0xA002C000
0xA0000030	Select CH_C0	0xA0030000
0xA0000034	Select CH_C1	0xA0034000
0xA0000038	Select CYCLE mode	0xA0038000
0xA000003C	Select FIFO mode	0xA003C000

4   ADC Initialization and Operation

As shown in the pin definition table of the ADS8364 data sheet, the 8 LSB's of the data bus are defined as digital I/O pins. These pins allow the ADS8364 to be controlled through software commands. When using software control, it is only necessary to write to the base address of the ADS8364 to enable the chip select. Channels pairs A through C can be converted individually or the CYCLE and FIFO modes can be set through software.

The Conversion and Reset commands are logically ANDed with the state of the $\overline{\text{HOLDA}}$, HOLDB, HOLDC, and RESET hardware pins. The ADD function, channel selection, and CYCLE and FIFO mode commands are logically ORed with the ADD, A2, A1 and A0 hardware pins. In order to use software control, tie the ADD, A2, A1, and A0 pins to logic zero, and tie the HOLDA, HOLDB, HOLDC, and RESET pins to logic one. The timing for the software commands is shown in Figure 2.

Conversion and reset commands are latched on the falling edge of the $\overline{\text{WR}}$ signal, while configuration commands are latched on the rising edge. Configuration commands need only be written once. Conversion commands need to be written with every conversion cycle. CS and WR should be held low simultaneously for 20 ns. Data should be valid for a minimum of 10 ns before (after) the write strobe.
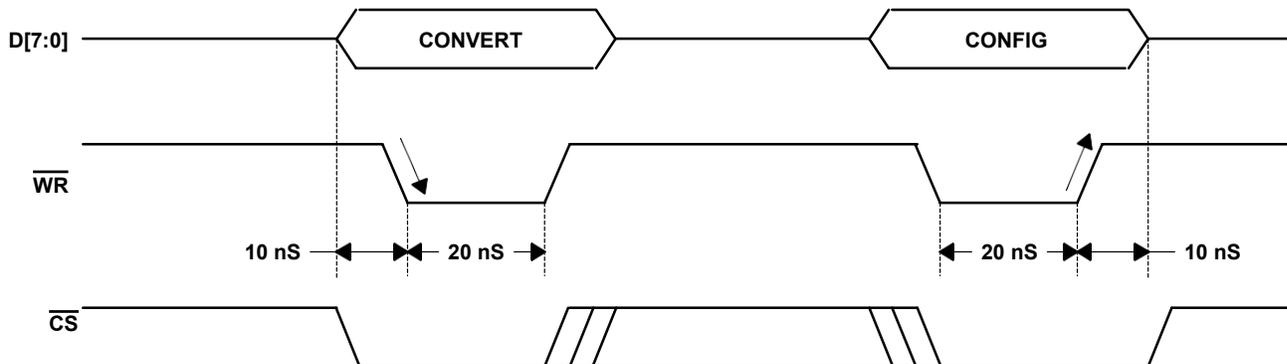

Figure 2. ADS8364 Write Timing

Table 3. ADS8364 Data Input/Command

COMMAND	D7	D6	D5	D4	D3	D2	D1	D0
CONVERT	0	x	x	x	$\overline{\text{RESET}}$	$\overline{\text{HOLDA}}$	$\overline{\text{HOLDB}}$	$\overline{\text{HOLDC}}$
CONFIGURATION	1	x	x	BIN/2s[†]	ADD	A2	A1	A0

[†] D4 = 1, data output format is straight binary; D4 = 0, data output format is 2s complement

4.1 Resetting the A8364

Toggling the active low reset pin (/RST) of the ADS8364 ensures that the read pointer is pointing to the first data register. As part of the initialization sequence for the ADC, the $\overline{RST}$ pin of the ADS8364 is brought high through the GPIO function of McBSPa. A *soft* reset command is issued, followed by a short delay. This ensures the data output of the ADC is aligned with the first set of data corresponding to channel A0, then A1, B0, B1, C0, and finally C1. The ADS8364 can be reset at any time by writing 0x0007 to the ADC's base address as defined in Table 2.

4.2 $\overline{HOLDx}$

The $\overline{HOLDx}$ hardware pins are active low sampling triggers. When the three $\overline{HOLD}$ lines are brought low together, all six analog inputs are simultaneously sampled and the conversion process begins with the next rising clock edge. The conversion is completed after 20 clock cycles, at which time the end of conversion (/EOC) pin goes low for 1/2 clock cycle. In software mode, writing zero to bits 2..0 of the conversion command at the ADC's base address initiates a conversion on channel pairs A, B, or C. In the code example associated with this application note, all three bits are cleared in a single write to the ADC with the conversion command 0x0008.

The $\overline{HOLDA}$, $\overline{HOLDB}$, and $\overline{HOLDC}$ hardware pins on the ADS8364 are controlled by the CLKRa, FSXa, and FSRa pins (as GPIO) of McBSPa. During the DSK's initialization sequence, the $\overline{HOLDx}$ lines are brought to a high state through the GPIO functions of the McBSP port.

4.3 End of Conversion ($\overline{EOC}$)

The $\overline{EOC}$ signal is active low once for each channel *pair* converted. The ADS8364, as depicted in the application note, provides three $\overline{EOC}$ pulses to the C6711DSK. Each pulse represents the completion of a conversion from the active channel. When all three hold pins of the ADC are brought low at the same time, all three channels (A, B, and C) are considered active and are converted simultaneously. The $\overline{EOC}$ signal is routed to EXT_INT4 on the C6711 DSK, located at pin 53 on the peripheral connector J3.

4.4 Reading Data

As noted in the data sheet, in cycle mode, when the $\overline{HOLDx}$ lines are simultaneously active, the conversion results are automatically stored in the data registers starting with channel A0. In FIFO mode, the first active $\overline{HOLDx}$ channel is stored in data registers A0 and A1, then B0 and B1, and finally C0 and C1. Otherwise, $\overline{HOLDa}$ conversion results are stored in registers A0 and A1, $\overline{HOLDb}$ conversion results are stored in registers B0 and B1 and $\overline{HOLDc}$ conversion results are stored in registers C0 and C1.

To get the maximum throughput from the ADS8364, the $\overline{HOLDx}$ command can be issued prior to reading the contents of the data registers. This allows the ADC to start a new conversion sequence while the host processor reads the results of the current conversion. If an interrupt service routine is used to read the data from the ADS8364, the read process must be completed before the new conversion cycle is finished (ADC issues EOC pulse). Failure to complete the read process results in data registers being overwritten. If maximum conversion rate is not a concern, data from the ADS8364 can be processed as it is required. The internal data registers hold conversion information until it is overwritten by a new conversion sequence, or cleared by a reset command.

Each channel can be read independently by simply writing the proper configuration / channel select command, then reading data from the ADC's base address. To read data from each channel sequentially, writing 0x0086 to the ADC places the ADS8364 in CYCLE mode. When CYCLE mode is selected, each read access automatically cycles through the six data registers, starting with channel A0 and ending with channel C1. Figure 3 shows a complete conversion sequence using CYCLE mode.
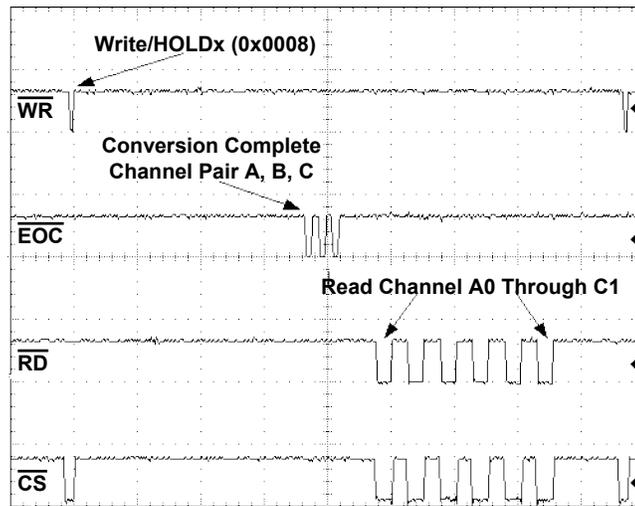

Figure 3. ADS8364 Waveforms

5   References

1. *ADS8364, 6 Channel, Simultaneous Sampling Parallel ADC, d*ata sheet, Texas Instruments SBAS219
2. *Code Composer Studio – Getting Started Guide,* user's manual, Texas Instruments SPRU509
3. *Code Composer Studio Online Users Guide,* user's manual, Texas Instruments SPRU415
4. *TMS320C6000 Technical Brief*, Texas Instruments SPRU197

IMPORTANT NOTICE AND DISCLAIMER