INTERFACING THE ADS1251/52 TO THE MSP430F449

Joe Purvis *Data Acquisition Digital Analog Converters*

ABSTRACT

This application report demonstrates an effective method to interface the ADS1251 and ADS1252 sigma-delta data converters to the MSP430F449 microcontroller. The software for this application report was developed on the HPA449 development system.

1 ADS125X Data Converter Family

The ADS1251and the ADS1252 are part of the ADS125x family of data converters. Table 1 lists the key features of each device in the family. For clarity, the features within the table are simplified. Consult the appropriate data sheet for specific device information.

Table 1. ADS125x Family Features

PART NUMBER	NUMBER OF INPUTS	RESOLUTION (BITS)	SPEED (KSPS)	REGISTER PROGRAMMABLE	POWER	
					NOMINAL SUPPLY VOLTAGE(S) V	CONSUMPTION (mW)
ADS1250	1 Differential	20	25	No	Analog +5 Digital +5	75
ADS1251	1 Differential	24	20	No	Analog +5	7.5
ADS1252	1 Differential	24	41	No	Analog +5	40
ADS1253	4 Differential	24	20	No	Analog +5	7.5
ADS1254	4 Differential	24	20	No	Analog +5	4.3
ADS1255	1 Differential	24	30	Yes	Analog +5 Digital +3.3	36
ADS1256	4 Differential	24	30	Yes	Analog +5	36
	8 Single ended	24			Digital +3.3	

1.1 ADS1251/52 ADC

The ADS1251 and ADS1252 are 24-bit, analog-to-digital converters; both incorporate the following functions on-chip:

- Fourth-order sigma-delta modulator
- Digital filter

The block diagram in Figure 1 illustrates these functions.

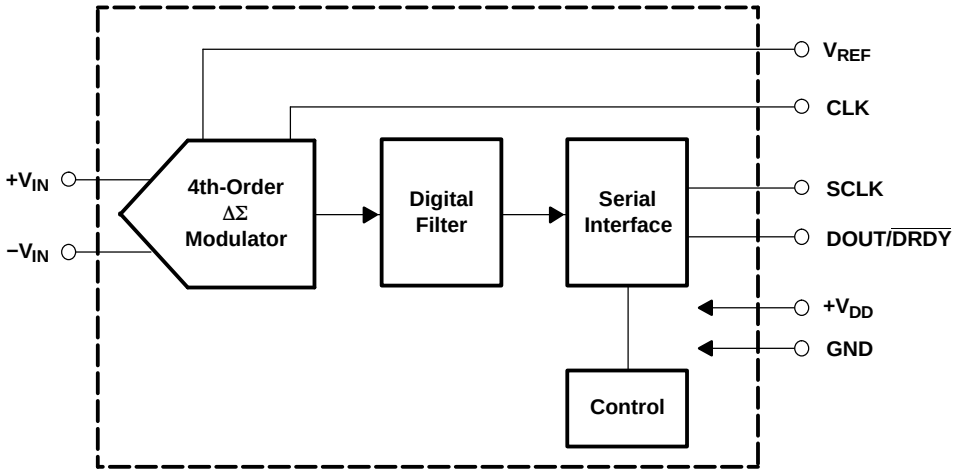


Figure 1. ADS1251/52 Functional Block Diagram

Each of these functions and the HPA449 development system are discussed in the following sections.

1.2 SIGMA-DELTA MODULATOR

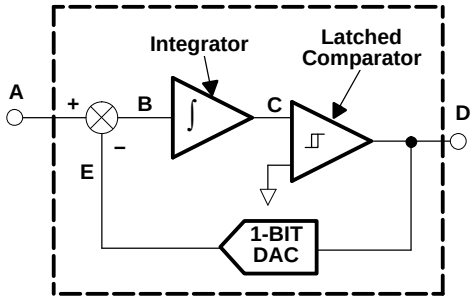


Figure 2. First-Order Sigma-Delta Modulator

The foundation of the ADS1251/52 is the sigma-delta modulator. Although detailed theory of this class of data converter is beyond the scope of this application report, the fundamental operation of this converter is discussed. Figure 2 shows a first-order sigma-delta modulator, which operates as follows.

The modulator's function is to reduce the error between the input signal and the previous output to zero. Assuming a dc input at A, this dc value is applied to a summing point and added to the value from the DAC, E. If the output from the summing junction is positive, the integrator output continues to ramp up until it trips the comparator. The comparator then signals to the DAC that the input has passed its upper threshold. In turn, the DAC then begins to reduce the effective signal at B, and the integrator begins to ramp down. The integrator continues to ramp down until it trips the comparator again. The comparator then signals to the DAC that the input has passed its lower threshold. In turn, the DAC then begins to increase the effective signal at B, and the integrator begins to ramp up again.

Because the comparator is latched, this action performs the quantization in time of the analog signal. The quantization in amplitude is accomplished through the density of the pulses originating from the comparator. If the input signal is close to full scale, more 1 pulses occur than 0 pulses in the bit stream from the comparator. Conversely, if the input-signal is close to negative full scale, more 0 pulses occur than 1 pulses in the bit stream from the comparator. Signals near midscale have equal numbers of 1 and 0 pulses. Figure 3 shows this graphically.

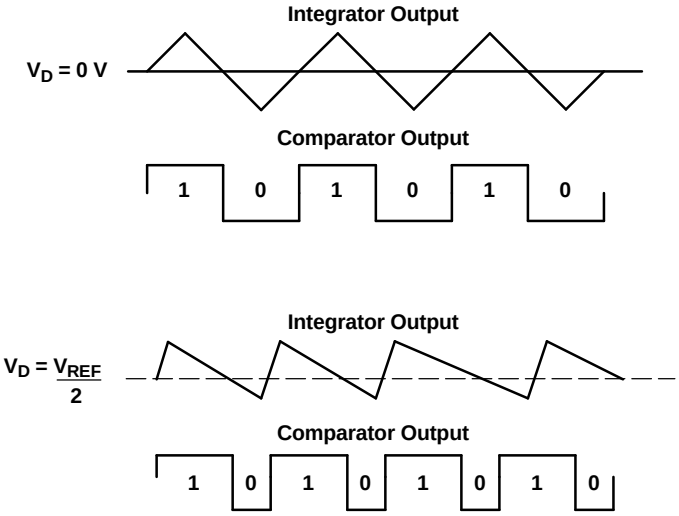


Figure 3. Comparator Pulse Density

1.2.1 FOURTH-ORDER DELTA-SIGMA MODULATOR

The modulator on an ADS125x converter is a fourth-order modulator constructed by cascading two single-bit, second-order modulators in a cascade or mesh topology. The two-modulator bit streams are then merged in a specific way to cancel out some of the unwanted lower order noise before they enter the digital filter.

1.3 DIGITAL FILTER

The process of averaging a time-domain signal is equivalent to performing a low-pass filtering operation. The more averages taken, the better characteristics the filter possesses, although the frequency response from an averager will always have a sinc characteristic.

The digital filter incorporated in the ADS1251/52 is referred to as a $sinc^5$ filter. The $sinc^2$ filter obtains the present value by averaging the previous two values. The $sinc^3$ filter uses the previous three values to obtain the present value; similarly, the $sinc^5$ filter uses the previous five values to obtain the present value. This is why, according to the ADS1251 and ADS1252 data sheets, the first valid data from the device does not occur until the sixth DOUT/$\overline{DRDY}$ cycle. Following this first valid cycle, every cycle thereafter contains valid data, although the data is the average of the previous five values.

The digital filter attenuates the out-of-band noise and decimates the data by a factor of 64.

The modulator clock for the EVM was set at 32.768 kHz, although this can be easily altered to any valid modulator frequency. In this specific instance, the pass-band response is as shown in Figure 4.

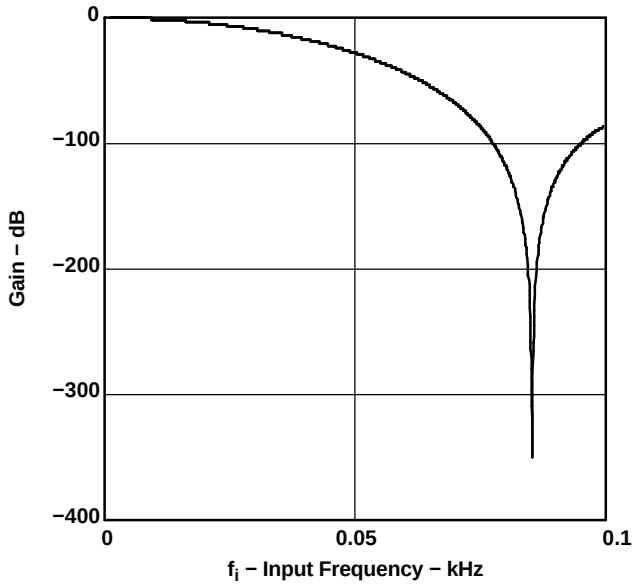


Figure 4. ADS1252 Pass-Band Response

The modulator clock establishes the data rate and the point of the notch in the frequency response. The $sinc^5$ filter accomplishes a low-pass response with a high rejection of frequencies around 86 Hz. The -3-dB point, in this case, is established around 20 Hz; this is acceptable for precise low-frequency signals.

In another example, if 50 Hz is required to be rejected, the user can set the modulator clock to 19.2 KHz. This yields a -3-dB point around 10 Hz.

1.4 HPA449 DEVELOPMENT SYSTEM

The HPA449 is the development system used to create the code that supports this application. Key features of this system follow.

- MSP430F449 processor
- Custom LCD
- RS-232 Interface
- JTAG programming header

The development system is available directly from SoftBaugh, Inc.; consult the SoftBaugh Web site at http://www.softbaugh.com

2 MSP430F449 Microcontroller

Figure 5 is the functional block diagram for the MSP430F449. The peripheral modules of this microcontroller that are used in this application are shown in Figure 5.

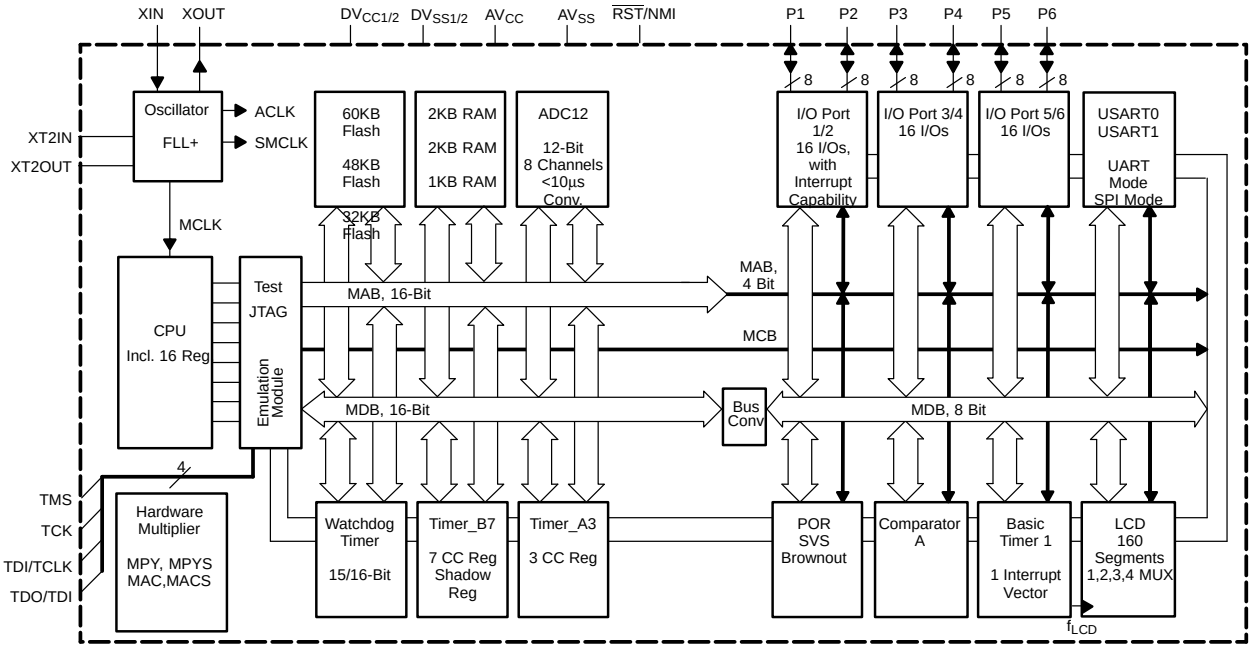


Figure 5. MSP430F449 Functional Block Diagram

Table 2. MSP430F449 Modules

MODULE	FUNCTION SUPPORTED	BIT FIELD	COMMENT
I/O Port 1	Source MCLK	Bit 1	Diagnostic service, these bits may be removed in software.
	Source ACLK	Bit 5	
I/O Port 2	Interrupt from the ADC	Bit 2	Serial A on HPA449
I/O Port 3	USART	Bit 1	SIMO (Data out)
		Bit 2	SOMI (Data in)
		Bit 3	UCLK(SCLK)
Timer_A3	The timer supports two functions:		
	Reset time – To ensure the ADC and MSP430 are synchronized, the ADC is reset. This is a period of at least 1536 MCLKs.		
	Wait time - Following an interrupt, data is not available until at least 36 MCLKs have elapsed.		

3 ADS1251/52 DIGITAL INTERFACE BASICS

Figure 6 shows the fundamental setup diagram.

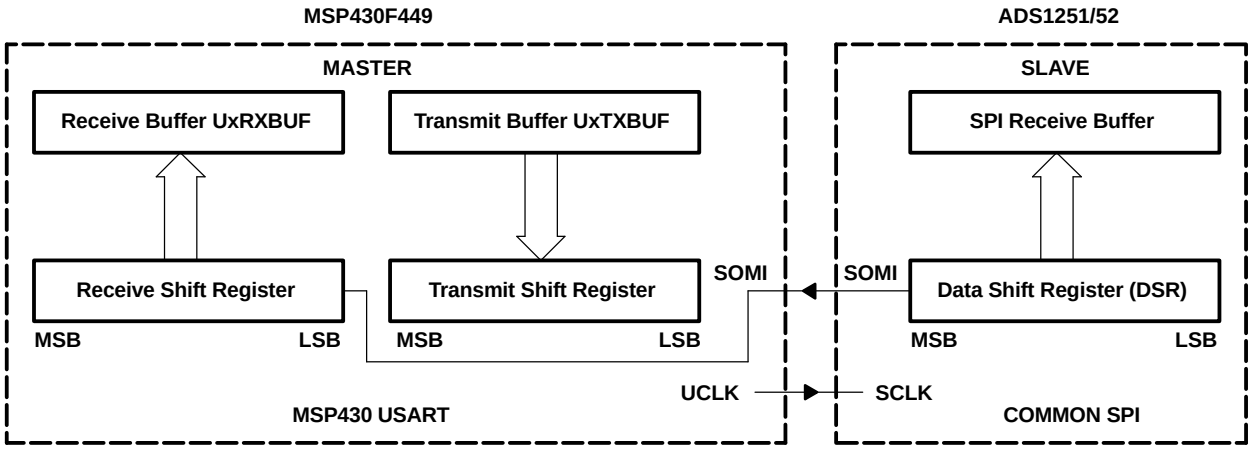


Figure 6. Interfacing the ADS1251/52 to the MSP430 Diagram

The ADS1251 and ADS1252 provide converted data as a serial bit stream to reduce the pin count and device size. The devices cannot be written to, only read from; this also reduces the complexity of the control required by the user.

The digital interface can control the following five functions:

- Power down the converter in a controlled manner
- Power up the converter in a controlled manner
- Synchronize the host system to the converter
- Indicate when new data is available
- Provide new data in a predictable manner

The ADS1251 and ADS1252 accomplish these functions through two pins:

- DOUT/$\overline{\text{DRDY}}$ - this is a multiplexed output pin that indicates when data is available to be read and also provides data to the host.
- SCLK – the logic level and duration of SCLK controls the power state of the converter and also enables synchronization of the converter to a host system. In addition, SCLK is used to shift data from the device to the host system.

3.1 ADS1251/52 OPERATING PROCEDURE

To successfully use this class of ADC, users must recognize that the host has to complete a transfer during the time allotted to it by virtue of the modulator clock. In this specific case :

Conversion cycle is 384 * MCLK. This is 384 * 30.5 µs = 11.7 ms, for this particular clock used.

Each conversion cycle has two phases: a $\overline{\text{DRDY}}$ mode and a DOUT mode (see Figure 7). The $\overline{\text{DRDY}}$ mode consumes 36 MCLK cycles, leaving the remaining 348 MCLK cycles for data, DOUT mode. Therefore, it is clear that with a 32.768-kHz modulator clock, the three bytes of data must be read out within the allotted 348 MCLK time, equivalent to 10.6 ms.

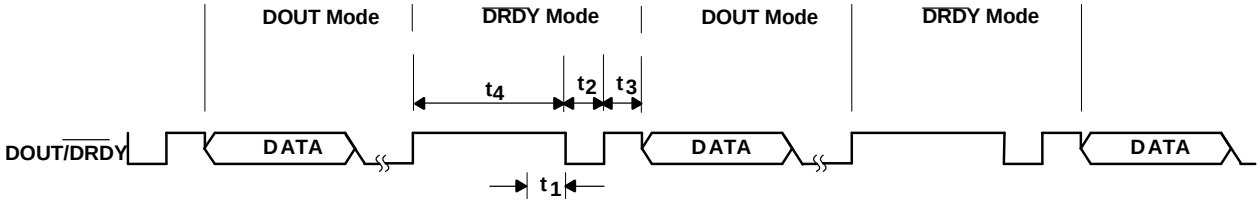


Figure 7. DOUT/$\overline{\text{DRDY}}$ Partioning

To achieve successful conversions, several events must occur sequentially:

- The system must be initialized.
- The ADC must be synchronized.
- Because only one pin is available for data and the signal indicating that data is (or is not) valid, the microprocessor must track whether the DOUT/$\overline{\text{DRDY}}$ pin is acting as either DOUT or $\overline{\text{DRDY}}$.

3.1.1 INITIALIZATION

Certain steps are required for the initialization of the MSP430 and the ADC.

For the MSP430, the following peripheral modules must be initialized:

- Clock source
- Port 2 - Interrupt
- Port 3 - USART

For the ADS1251/52, these actions must be taken:

- Reset the modulator of the ADC.
- Prepare the ADC for a data transfer.

The following discussion examines each of these steps in more detail.

3.1.1.1 INITIALIZING THE CLOCK SOURCE

void setup_osc(void)

The clock modules and the control system for the MSP430F449 are shown in Figure 8.

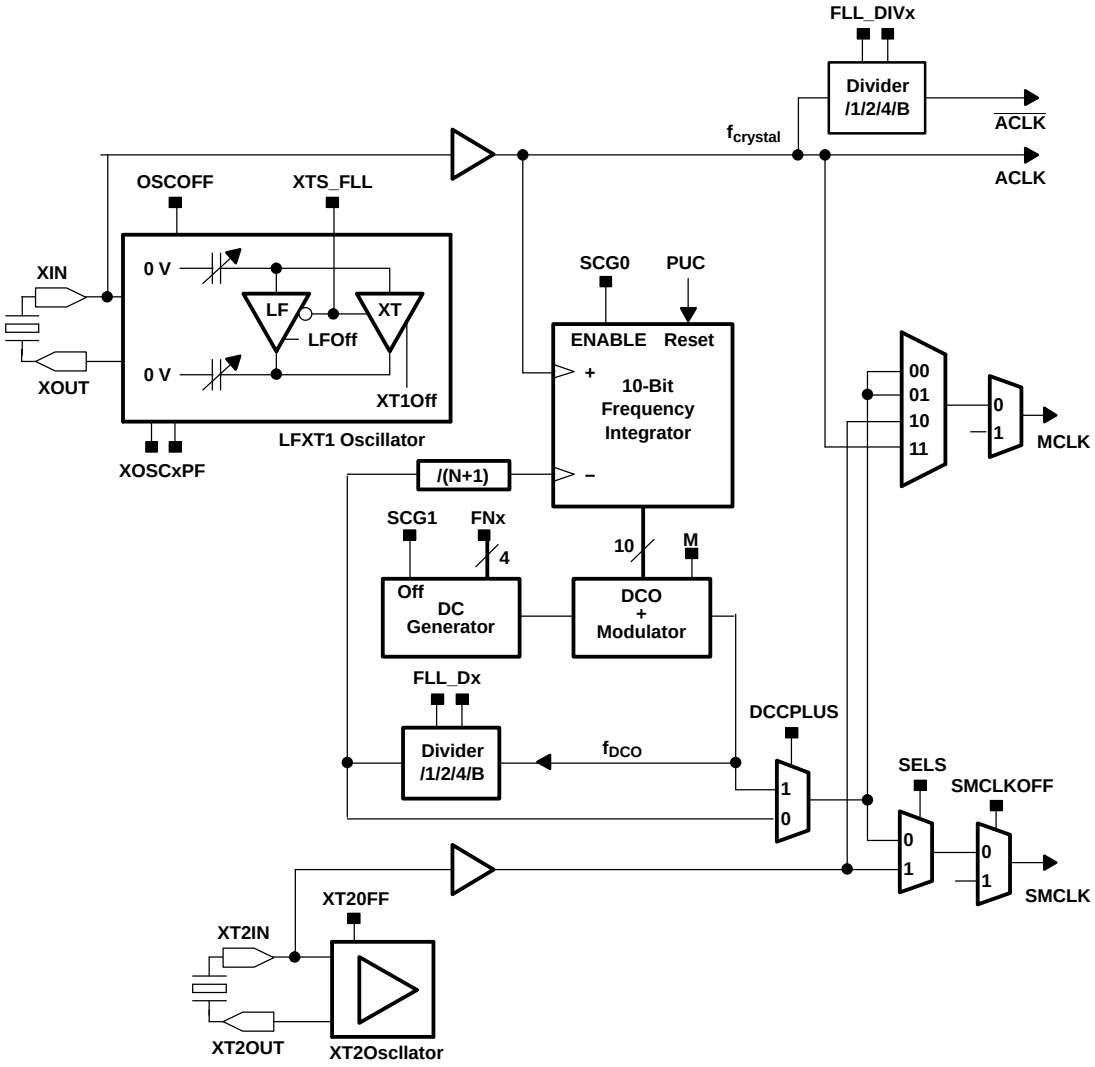


Figure 8. MSP430F449 Clock Modules and Control System

Two clocks are used in this application:

- LFXT1
- XT2

LFXT1 is a low-frequency crystal oscillator, sourced from the HPA449 assembly. LFXT1 has a frequency of 32.768 kHz. Reliable crystal operation depends on the proper load capacitance. This capacitance can be selected by software using the XCAPxPF bits. If the program fails to proceed past the Oscillator Flag check (LFOF), it may be necessary to change the XCAPxPF values in the program to ensure reliable function. The LFXT1 clock is used to provide the clock for timer A.

XT2 can be used as the source of MCLK and SMCLK as shown in Figure 8. In this application, XT2 is derived from an 8-MHz resonator connected between XT2IN and XT2OUT, and sourced by the user. (The resonator manufacturer used in this application is Murata, and the part number is CSTLS8M00G53-B0.) This resonator supplies the clocks required for the system, namely:

- MCLK – Master Clock used by the system.
- SMCLK – Sub-Main Clock, this clock is software selectable for individual peripheral modules.

SMCLK is the source clock for the USART (universal synchronous asynchronous receiver transmitter). The HPA449 board has a location available (marked as X2 on the silkscreen) where you can either solder the resonator directly to the HPA449 or use a single-in-line socket strip, in case it becomes necessary to change resonators.

It is useful to observe clock signals actually originating from the MSP430; this gives some assurance that the clocks required to operate the device are working as expected. On the MSP430F449, MCLK, SMCLK, and ACLK can be observed from select pins on the device, P1.1, P1.4, and P1.5, respectively. This can be achieved by setting P1.1, P1.4, and P1.5 as special function and then setting each bit as an output. This enables MCLK, SMCLK, and ACLK to be monitored from P1.1, P1.4, and P1.5, respectively. Choose suitable points on the HPA449 development board to probe the output signals.

3.1.1.2 INITIALIZING THE MSP430 PORTS

void setup_ports(void);

This procedure sets the direction of the I/O ports. Inspection of the HPA449 schematic indicates that the following bits should be set as inputs:

BIT 2 — Interrupt A

This is the interrupt from serial site A, the serial site for which the program has been coded.

BIT 7 — Interrupt B

This is the interrupt from serial site B, the other serial site. Because it may be used for future systems, also set this pin as an input. $\overline{\text{INTB}}$ is pulled HIGH through a 47-kΩ resistor, and because the system only responds to transitions, this causes no problems.

3.1.1.3 INITIALIZING THE MSP INTERRUPTS

void setup_ints(void)

Each pin on port 2 has interrupt capability. Any interrupt on P2 sources a single interrupt vector for port 2. Port 2, bit 2 of the MSP430 is set to receive an interrupt from the ADC; this interrupt is set to occur on a low-to-high transition.

3.1.1.4 USING TIMER A

An outline of the timer-A module and its control structure is shown in Figure 9. The timer is used in two situations:

- To determine enough time has elapsed for a valid reset of the modulator
- To determine sufficient time has elapsed for $\overline{\text{DRDY}}$ mode to expire before reading data out of the device

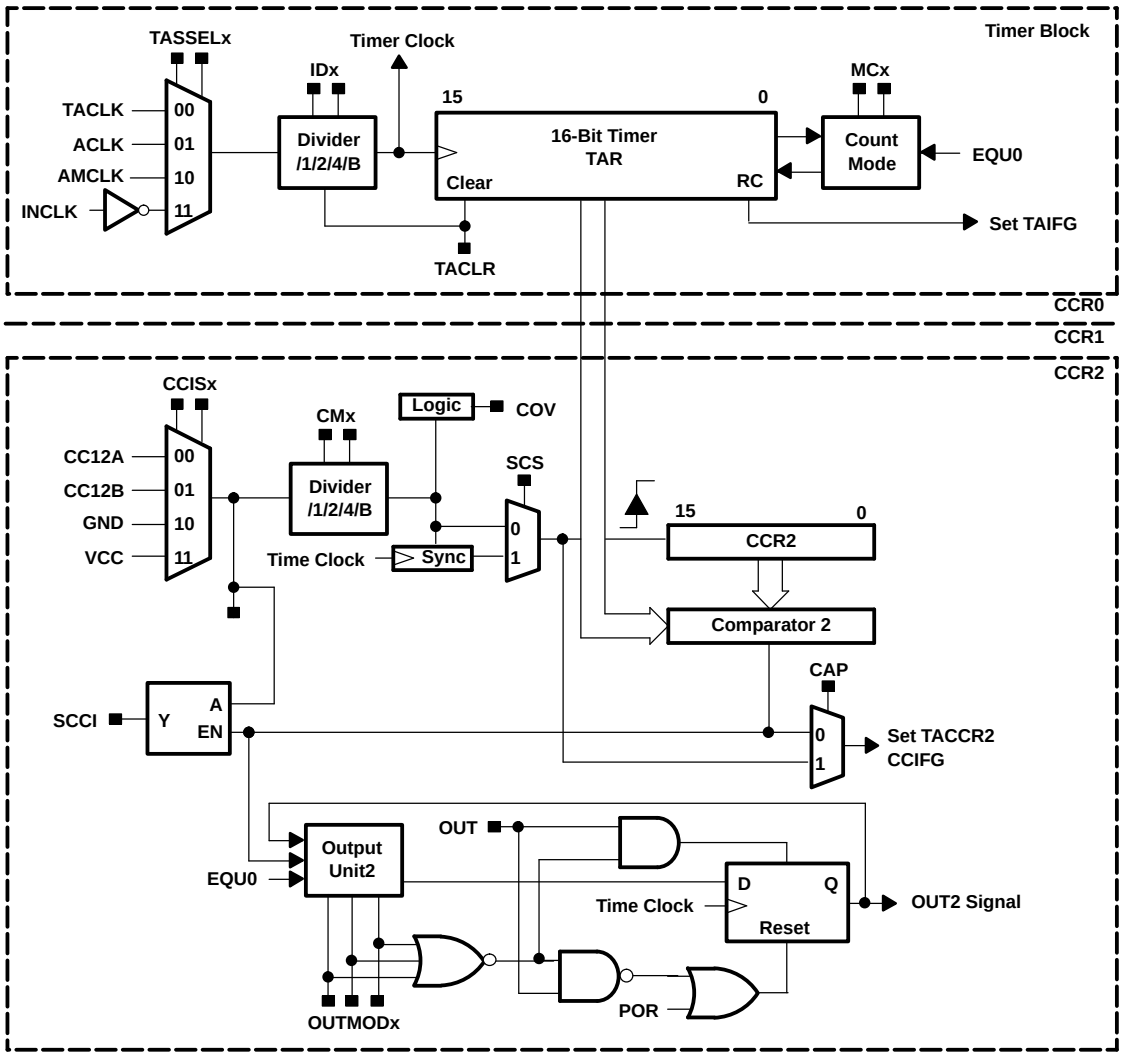


Figure 9. Timer-A Module Control Structure

Timer A is set to use ACLK as its clock source.

3.1.1.5 RESETTING THE ADC MODULATOR

void reset_adc(float MCLK_PRD, float ACLK_PRD);

With a running modulator clock and appropriate power applied to the converter, the DOUT/ $\overline{\text{DRDY}}$ pin continuously toggles between DOUT mode and $\overline{\text{DRDY}}$ mode. Therefore, it may be difficult for the host system to recognize an interrupt as an interrupt rather than recognize data as an interrupt.

This problem is solved by asserting the signal to the SCLK pin HIGH. When this signal is asserted for a sufficient time, the modulator enters a reset state. The modulator can then be released from the reset state in a controlled manner, by de-asserting the SCLK signal LOW. It is the action of de-asserting the SCLK signal that releases the modulator from reset and begins the DOUT/$\overline{\text{DRDY}}$ cycle.

Resetting the ADS1251/52 requires that the following condition be met:

$4^* t_{\overline{\text{DRDY}}} < [\text{SCLK} = 1] < 20^* t_{\overline{\text{DRDY}}}$

This condition indicates that the SCLK signal should be asserted HIGH for at least four consecutive DOUT/$\overline{\text{DRDY}}$ periods, but no more than 20 DOUT/$\overline{\text{DRDY}}$ periods, because more than 20 periods causes the modulator to power down. The DOUT/$\overline{\text{DRDY}}$ period is directly related to the modulator clock frequency as:

*$t_{\overline{DRDY}}$ (Conversion cycle) = 384 * modulator clock periods.*

A modulator clock of 32.768 kHz, as in this example, represents a clock period of 30.5 µs, and a $t_{\overline{DRDY}}$ time of 11.7 ms. Therefore, to ensure that the modulator is reset, SCLK must be asserted HIGH for at least 4 * $t_{\overline{DRDY}}$ cycles; this is 4 * 11.7 ms which is 46.8 ms.

This high-time reset is achieved by using the MSP430's timer-A module (see Figure 9) and determining a count of clocks based on the ACLK period. This time period changes if either the modulator clock for the ADS1251/52 is changed or the ACLK clock frequency is changed. Therefore, the calculation is achieved by passing two variables into the reset procedure:

- MCLK_PRD
- ACLK_PRD

The procedure determines the number of counts required based on timer A's clock source and the period of the clock supplied to the ADS1251/52 modulator.

The program tests the interrupt flag, CCIFG, until the count has reached the value loaded into TACCR.

Figure 10 shows the reset cycle in this case.

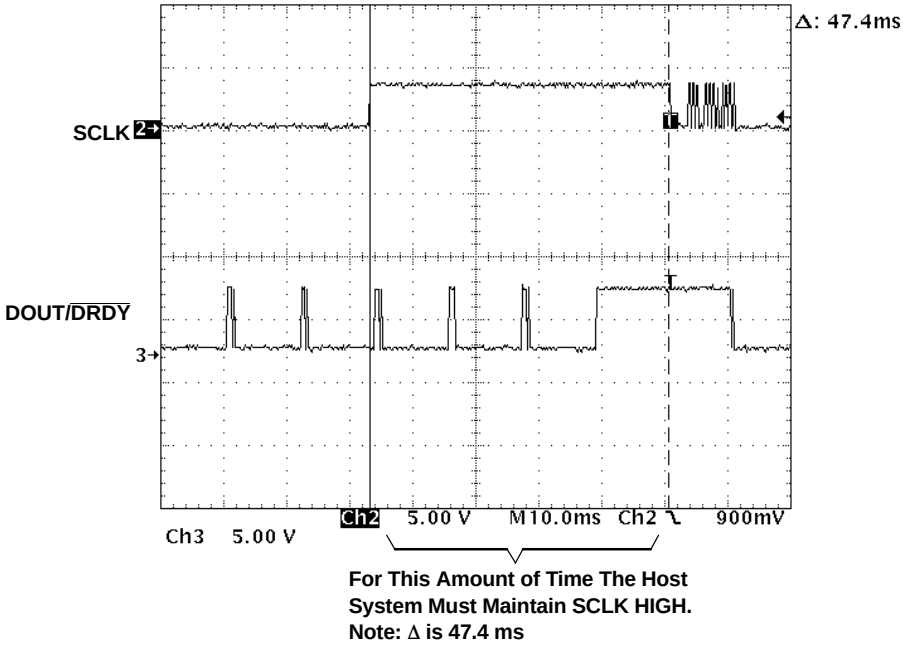
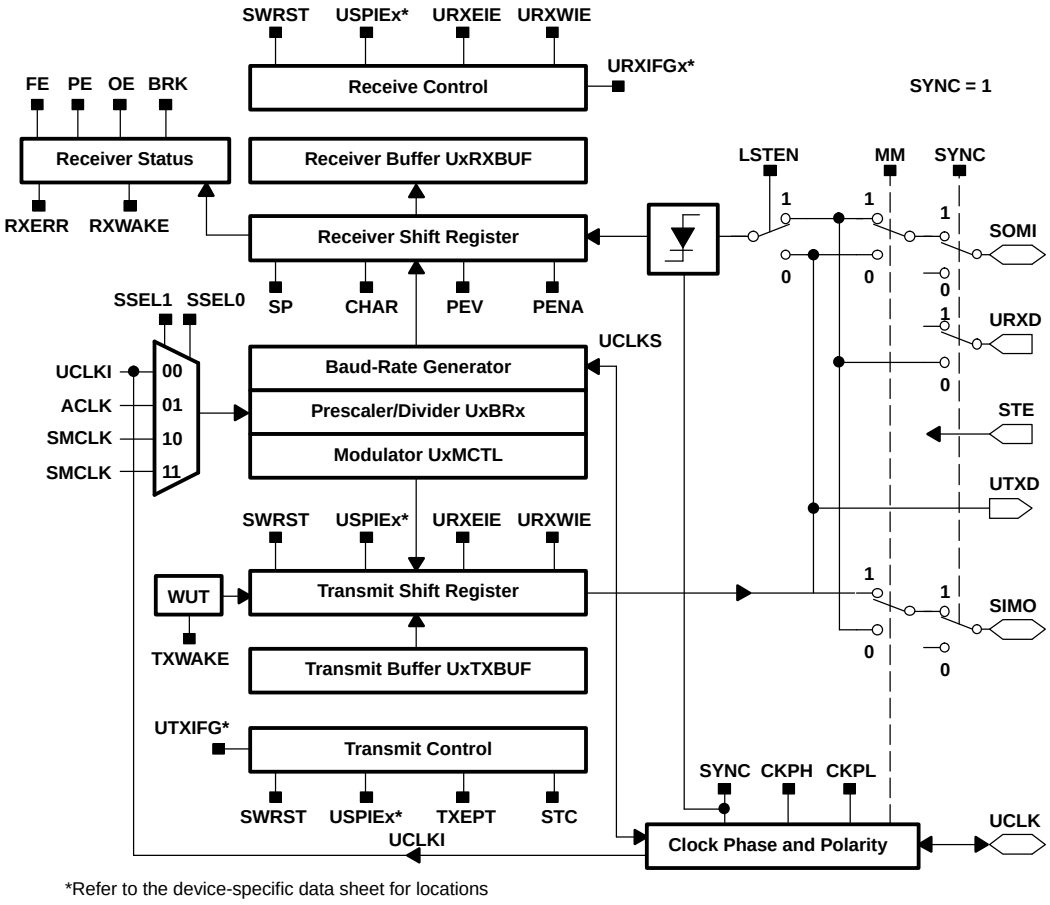


Figure 10. Reset Cycle

3.1.1.6 INITIALIZING THE USART

void setup_SPI(void)

Figure 11 shows the block diagram for the USART.



*Refer to the device-specific data sheet for locations

Figure 11. USART Block Diagram

In this application, the procedure is as follows:

- Reset the USART module.
- Enable SPI mode.
- Set the specific SPI function bits (see Table 3).

Table 3. USART BIT ASSIGNMENTS(1)

BIT POSITION	FUNCTION
Bit 1	SIMO (Data Out)
Bit 2	SOMI (Data In)
Bit 3(2)	UCLK (SCLK)

(1) Table 3 shows the bit assignment for the special function in the port 3 module.

(2) As soon as bit 3 is assigned to the UCLK, the signal changes from a GPIO pin to the UCLK function. This asserts UCLK (SCLK for the ADS1251/52) LOW and causes the modulator of the ADC to be released from its reset state; consequently, the ADC's continuous DOUT/$\overline{\text{DRDY}}$ cycle begins.

- Select SPI mode for the USART, in this case, master mode with sync = 1 and a character length of 8 bits.
- Select baud rate for the SPI clock, UCLK.
- Select transmission protocol.
- Release the USART module from reset mode.

3.2 RETRIEVING DATA

void make_array(void);

This function reads 512 words of data and stores them in an array called *ad_buffer*.

Following the LSB of data, the remaining time of DOUT mode is characterized by DOUT asserted LOW. The first rising edge after this time is the beginning of $\overline{\text{DRDY}}$ mode.

The process generally proceeds as follows:

- Wait until an interrupt occurs.
- After an interrupt occurs, go to the interrupt service routine (ISR).
- Wait for at least 36 MCLK periods, to ensure that the $\overline{\text{DRDY}}$ mode has concluded.
- Read first byte of data (most significant byte) and shift left 8 bits.
- Read second byte of data and shift left 8 bits.
- Read third byte of data (least significant byte) and shift left 8 bits.
- Increment the array index.
- Return and wait for the next interrupt.

A few additional comments are worthy of note regarding the ADC's operation within the program's boundary.

After the interrupt occurs, an aggregate delay time denoted in the ADS1251 and ADS1252 data sheets as $t_4 + t_2 + t_3$ occurs before the end of the $\overline{\text{DRDY}}$ mode and the beginning of the DOUT mode. This time is (24 + 6 + 6) MCLK periods; in this example, the time therefore is 36 * 30.5 µs = 1.1 ms. Therefore, wait at least 1.1 ms before attempting to read any data from the device; this delay is realized by timer A indicating that the CCIFG flag is set.

Figure 12 is a composite diagram showing the three sections of the $\overline{\text{DRDY}}$ phase:

- 24 MCLKs HIGH
- 6 MCLKs LOW
- 6 MCLKs HIGH

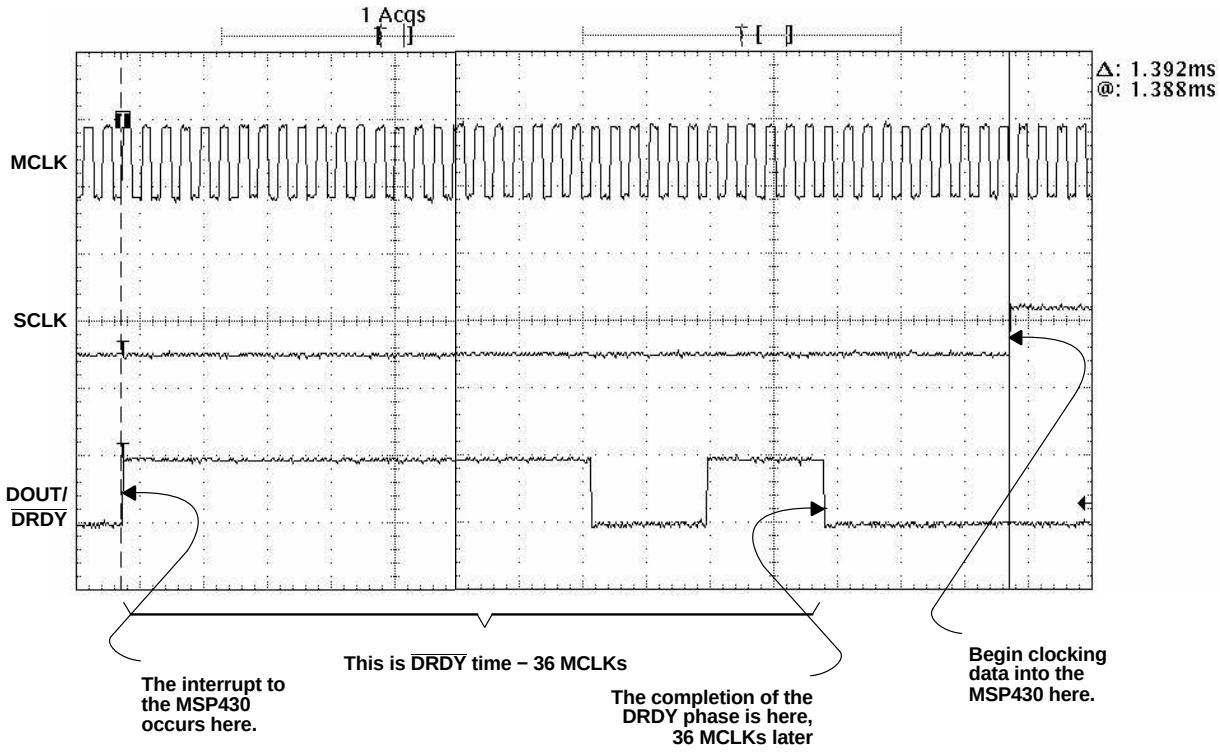


Figure 12. Three Sections of the $\overline{\text{DRDY}}$ Phase

Because each data word from the ADC is 24 bits long, and the SPI port of the MSP430 is only 8 bits wide, three bytes are required for one complete read. Each data word is constructed by waiting until the receive register indicates it is full (URXIFG0 = 1), reading the SPI receive register (U0RXBUF), storing the byte as appropriate, and repeating the procedure three times, to construct a 24-bit word.

One complete 24-bit transfer is shown in Figure 13.

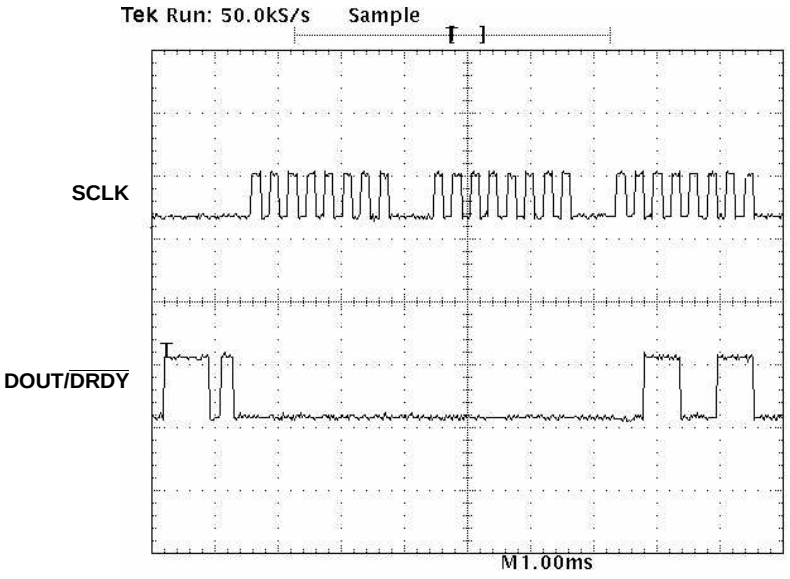


Figure 13. Complete 24-Bit Transfer

The result of the conversion is transmitted MSB first in an offset 2s-complement format.

In this example, both inputs to the ADC are at 0 V; therefore, the code resolved by the ADC is close to 0x000000. The transfer function for the ADC is shown in Figure 14.

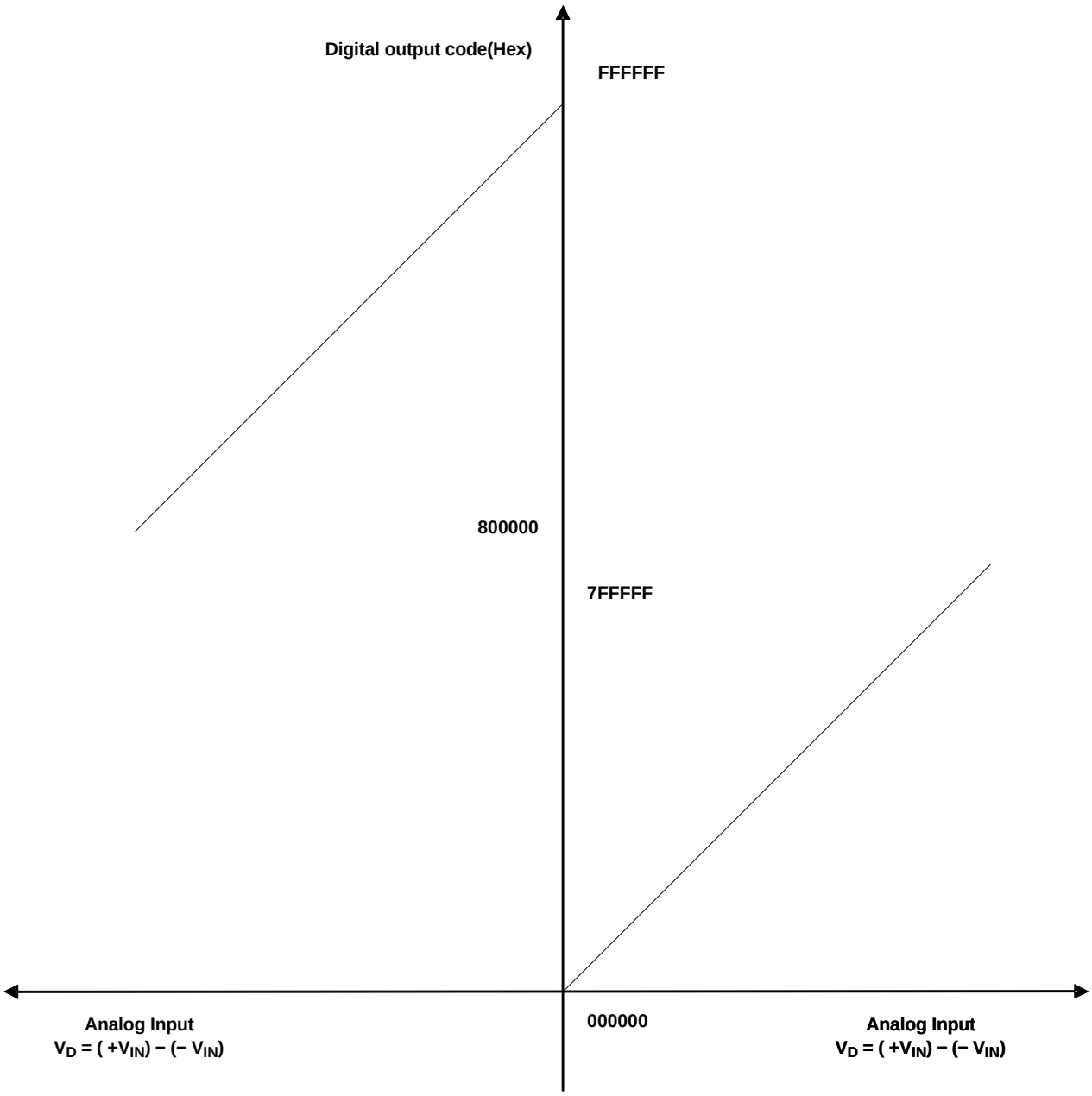


Figure 14. ADC Transfer Function

4 REFERENCES

1. *MSP430x4xx Family User's Guide*(SLAU056)
2. *MSP430x43x, MSP430x44x Mixed Signal Microcontroller* (SLAS344)
3. *ADS1251, 24-Bit, 20kHz Analog-to-Digital Converter*(SBAS184).
4. *ADS1252, 24-Bit, 40kHz Analog-to-Digital Converter* (SBAS127)
5. *A Spreadsheet Calculating the Frequency Response of the ADS1250–54* (SBAA103)
6. *Delta-Sigma Data Converters, Theory Design and Simulation*, R. Norsworthy, R. Schreier, and G.C. Temes, Wiley–IEEE Press, ISBN 0-7803-1045-4

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265