TMS320VC5507/5509 DSP Analog-to-Digital Converter (ADC) Reference Guide

Literature Number: SPRU586B
June 2004

Texas Instruments

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

Preface

Read This First

About This Manual

This manual describes the features and operation of the analog-to-digital converter that is on the TMS320VC5507 and TMS320VC5509 digital signal processors (DSPs) in the TMS320C55x™ (C55x™) DSP generation.

Notational Conventions

This document uses the following conventions:

- In most cases, hexadecimal numbers are shown with the suffix h. For example, the following number is a hexadecimal 40 (decimal 64):

  40h

- Binary numbers often are shown with the suffix b. For example, the following number is the decimal number 4 shown in binary form:

  0100b

Related Documentation From Texas Instruments

The following documents describe the C55x devices and related support tools. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

TMS320VC5507 Fixed-Point Digital Signal Processor Data Manual (literature number SPRS244) describes the features of the TMS320VC5507 fixed-point DSP and provides signal descriptions, pinouts, electrical specifications, and timings for the device.

TMS320VC5509 Fixed-Point Digital Signal Processor Data Manual (literature number SPRS163) describes the features of the TMS320VC5509 fixed-point DSP and provides signal descriptions, pinouts, electrical specifications, and timings for the device.

TMS320VC5509A Fixed-Point Digital Signal Processor Data Manual (literature number SPRS205) describes the features of the TMS320VC5509A fixed-point DSP and provides signal descriptions, pinouts, electrical specifications, and timings for the device.

TMS320C55x Technical Overview (literature number SPRU393) introduces the TMS320C55x DSPs, the latest generation of fixed-point DSPs in the TMS320C5000™ DSP platform. Like the previous generations, this processor is optimized for high performance and low-power operation. This book describes the CPU architecture, low-power enhancements, and embedded emulation features.

TMS320C55x DSP CPU Reference Guide (literature number SPRU371) describes the architecture, registers, and operation of the CPU for the TMS320C55x DSPs.

TMS320C55x DSP Peripherals Overview Reference Guide (literature number SPRU317) introduces the peripherals, interfaces, and related hardware that are available on TMS320C55x DSPs.

TMS320C55x DSP Algebraic Instruction Set Reference Guide (literature number SPRU375) describes the TMS320C55x DSP algebraic instructions individually. Also includes a summary of the instruction set, a list of the instruction opcodes, and a cross-reference to the mnemonic instruction set.

TMS320C55x DSP Mnemonic Instruction Set Reference Guide (literature number SPRU374) describes the TMS320C55x DSP mnemonic instructions individually. Also includes a summary of the instruction set, a list of the instruction opcodes, and a cross-reference to the algebraic instruction set.

TMS320C55x Optimizing C/C++ Compiler User's Guide (literature number SPRU281) describes the TMS320C55x C/C++ Compiler. This C/C++ compiler accepts ISO standard C and C++ source code and produces assembly language source code for TMS320C55x devices.

TMS320C55x Assembly Language Tools User's Guide (literature number SPRU280) describes the assembly language tools (assembler, linker, and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for TMS320C55x devices.

TMS320C55x DSP Programmer's Guide (literature number SPRU376) describes ways to optimize C and assembly code for the TMS320C55x DSPs and explains how to write code that uses special features and instructions of the DSPs.

Trademarks

TMS320, TMS320C5000, TMS320C55x, and C55x are trademarks of Texas Instruments.

Trademarks are the property of their respective owners.

Contents

1 Introduction to the ADC 7

2 Total Conversion Time 8

3 Initiating and Monitoring a Conversion Cycle 9

4 Conserving Power 9

5 ADC Registers 10
5.1 ADC Control Register (ADCCTL) 10
5.2 ADC Data Register (ADCDATA) 11
5.3 ADC Clock Divider Register (ADCCLKDIV) 13
5.4 ADC Clock Control Register (ADCCLKCTL) 15

6 Conversion Example 16

Revision History 18

Figures

1 ADC Block Diagram 7
2 ADC Total Conversion Time 8
3 ADC Control Register (ADCCTL) 10
4 ADC Data Register (ADCDATA) 12
5 ADC Clock Divider Register (ADCCLKDIV) 13
6 ADC Clock Control Register (ADCCLKCTL) 15
7 Total Conversion Time for the Conversion Example 17

Tables

1 ADC Registers 10
2 ADC Control Register (ADCCTL) Field Values 11
3 ADC Data Register (ADCDATA) Field Values 12
4 ADC Clock Divider Register (ADCCLKDIV) Field Values 13
5 ADC Clock Control Register (ADCCLKCTL) Field Values 15

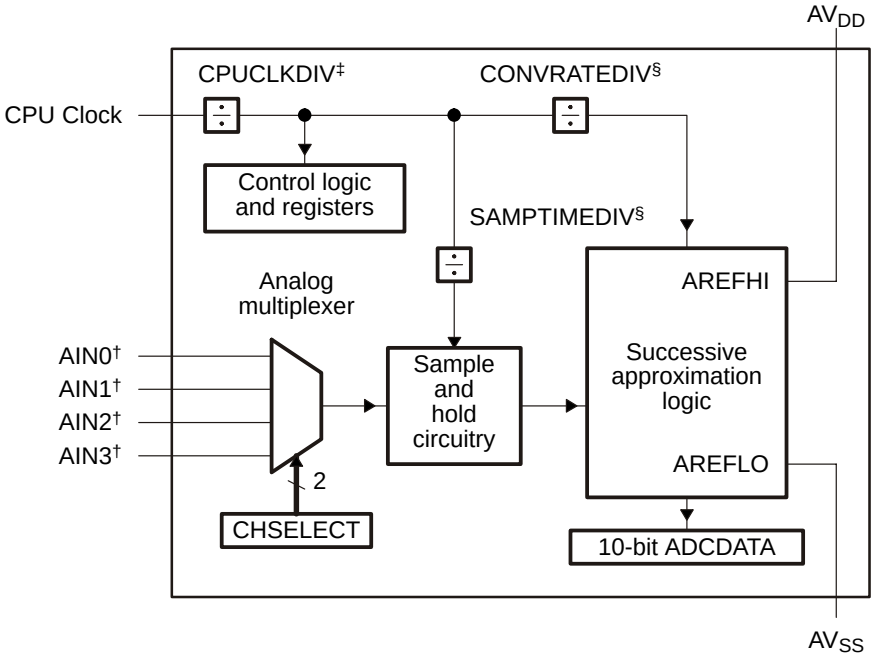
Analog-to-Digital Converter (ADC)

This chapter describes the 10-bit successive-approximation analog-to-digital converter (ADC) that is on TMS320VC5507/5509 DSPs.

1 Introduction to the ADC

The ADC (see Figure 1) converts an analog input signal to a digital value for use by the DSP. The ADC can sample one of up to four inputs (AIN0-AIN3) at a time, and generates a 10-bit digital representation (ADCDATA) of the samples. The maximum sampling rate of the ADC is 21.5 kHz. This performance makes the ADC suitable for sampling analog signals that change at a slow rate. For example, the ADC could be used to sample the voltage across a potentiometer on a user interface panel or to monitor the supply voltage drop on a battery operated circuit. The ADC is not intended for sampling data for signal processing.

Figure 1. ADC Block Diagram



† The TMS320VC5507/5509 DSPs are available in more than one type of package. Check the device-specific data manual to determine the number of analog input (AIN) pins available on a particular package.
‡ CPUCLKDIV is a field of the register ADCCLKCTL, which is described in section 5.4 (page 15).
§ CONVRATEDIV and SAMPTIMEDIV are fields of the register ADCCLKDIV, which is described in section 5.3 (page 13).

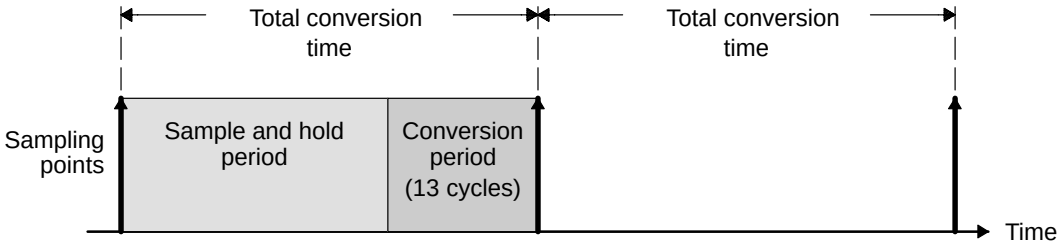
The ADC is based on a successive approximation architecture. A sample and hold feature is employed to help produce evenly spaced samples. The ADC uses external reference voltages on pins AV_{DD} and AV_{SS} to isolate the conversion process from the rest of the system. AV_{DD} supplies the high reference voltage, and AV_{SS} supplies the low reference voltage.

2 Total Conversion Time

The total conversion time of the ADC (see Figure 2) has two components—the sample and hold period, and the conversion period.

- The sample and hold period is the time required for an analog sample to be acquired by the sample and hold circuitry; this time period is greater than or equal to 40 μs.
- The conversion time period is the time required for the successive approximation process to convert one sample to the corresponding digital value. This conversion period requires 13 conversion clock cycles to complete. The internal conversion clock has a maximum frequency of 2 MHz.

Figure 2. ADC Total Conversion Time



The following equations describe the relationship between the ADC programmable clock dividers:

ADC Clock = (CPU Clock) / (CPUCLKDIV + 1)

ADC Conversion Clock = (ADC Clock) / (2 × (CONVRATEDIV + 1))
[must be less than or equal to 2 MHz]

ADC Sample and Hold Period =
(1 / (ADC Clock)) / (2 × (CONVRATEDIV + 1 + SAMPTIMEDIV))
[must be greater than or equal to 40 μs]

ADC Total Conversion Time =
(ADC Sample and Hold Period) + (13 × (ADC Conversion Clock Period))

3 Initiating and Monitoring a Conversion Cycle

The ADC operates in single-shot mode only; therefore, the DSP must initiate each conversion by writing a 1 to the ADCSTART bit in the ADC control register (ADCCTL).

Once a conversion is started, the DSP must wait until the conversion completes before selecting another channel or initiating a new conversion. The ADC does not support interrupts to the DSP or the DMA controller, so the DSP must poll the status of a conversion using the ADCBUSY bit in the ADC data register (ADCDATA).

After the conversion process completes, the ADCBUSY bit value changes from 1 to 0, indicating that the conversion data is available. The DSP can then read the data from the ADCDATA bits in ADCDATA. The value of the channel select (CHSELECT) bits in ADCCTL is reproduced in the register ADCDATA, so that the DSP can identify which samples were acquired from which channel.

4 Conserving Power

To conserve power, you can place the ADC into a low-power mode. A number of domains in the DSP can be individually turned on (made active) or turned off (made idle). The ADC is part of the peripherals domain. When the peripherals domain becomes idle, the IDLEEN bit of ADCCLKCTL determines whether the ADC becomes idle (IDLEEN = 1) or remains active (IDLEEN = 0).

5 ADC Registers

The ADC registers are listed in Table 1 and described in sections 5.1 through 5.4. These are registers are accessible at addresses in the I/O space of the DSP.

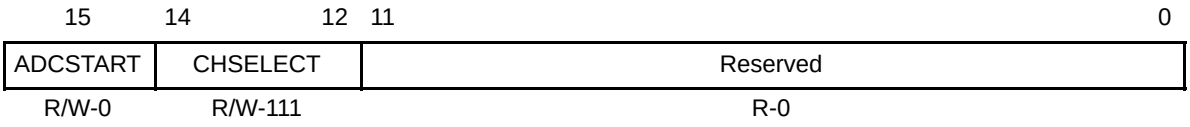
Table 1. ADC Registers

Address (Hex)	Name	Description
6800	ADCCTL	ADC control register
6801	ADCDATA	ADC data register
6802	ADCCLKDIV	ADC clock divider register
6803	ADCCLKCTL	ADC clock control register

5.1 ADC Control Register (ADCCTL)

The ADC control register (see Figure 3 and Table 2) is a read/write register used to select the analog input channel and to start a conversion.

Figure 3. ADC Control Register (ADCCTL)



Legend: R = Read; W = Write; -*n* = Value after reset

Table 2. ADC Control Register (ADCCTL) Field Values

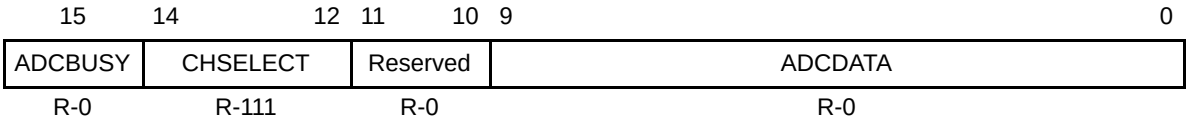
Bit	Field	Value	Description
15	ADCSTART		Start of conversion bit
		0	Writing 0 has no effect.
		1	Start conversion cycle. The ADCSTART bit is automatically cleared during the conversion cycle.
14-12	CHSELECT		Analog input channel select bits. These bits determine on which of four analog inputs the ADC performs a conversion.
		000b	Analog input AIN0† is selected
		001b	Analog input AIN1† is selected
		010b	Analog input AIN2† is selected.
		011b	Analog input AIN3† is selected.
		100b-111b	All analog switches are off.
11-0	Reserved		These reserved bits are always read as 0s.

† The TMS320VC5507/5509 DSPs are available in more than one type of package. Check the device-specific data manual to determine the number of analog input (AIN) pins available on a particular package.

5.2 ADC Data Register (ADCDATA)

The ADC data register (see Figure 4 and Table 3) is a read-only register that indicates whether a conversion is in process, records the actual digital data converted from the analog signal, and indicates from which channel the data came.

Figure 4. ADC Data Register (ADCDATA)



Legend: R = Read; W = Write; -*n* = Value after reset

Table 3. ADC Data Register (ADCDATA) Field Values

Bit	Field	Value	Description
15	ADCBUSY		ADC busy bit
		0	ADC data is available.
		1	ADC is busy performing a conversion. After ADCSTART bit is high, the ADCBUSY bit becomes high. This bit returns to 0 when the conversion is complete.
14-12	CHSELECT		Channel select bits. These bits indicate which analog input supplied the conversion data in the ADCDATA bits.
		000b	Analog input AIN0† is selected.
		001b	Analog input AIN1† is selected.
		010b	Analog input AIN2† is selected.
		011b	Analog input AIN3† is selected.
		100b-111b	Reserved
11-10	Reserved		These reserved bits are always read as 0s.
9-0	ADCDATA		ADC data bits. The 10-bit digital data converted from the analog signal.

† The TMS320VC5507/5509 DSPs are available in more than one type of package. Check the device-specific data manual to determine the number of analog input (AIN) pins available on a particular package.

5.3 ADC Clock Divider Register (ADCCLKDIV)

The ADC clock divider register (see Figure 5 and Table 4) is a read/write register that indicates the divider values for the conversion clock and the sample and hold time.

Figure 5. ADC Clock Divider Register (ADCCLKDIV)

15	8	7	4	3	0
SAMPTIMEDIV		Reserved		CONVRATEDIV	
R/W-0		R-0		R/W-1111	

Legend: R = Read; W = Write; -*n* = Value after reset

Table 4. ADC Clock Divider Register (ADCCLKDIV) Field Values

Bit	Field	Value	Description
15-8	SAMPTIMEDIV	0-255	Sample and hold time divider bits. This 8-bit value in conjunction with the CONVRATEDIV bits and the ADC clock period determines the sample and hold period as follows: ADC Sample and Hold Period = (ADC Clock Period) $\times$ (2 $\times$ (CONVRATEDIV + 1 + SAMPTIMEDIV))
7-4	Reserved		These reserved bits are always read as 0s.

Table 4. ADC Clock Divider Register (ADCCLKDIV) Field Values (Continued)

Bit	Field	Value	Description
3-0	CONVRATEDIV		Conversion clock rate divider bits. This 4-bit value determines the ADC clock divide down value required to produce the desired conversion clock: ADC Conversion Clock = (ADC Clock) / (2 × (CONVRATEDIV + 1)) Also, this 4-bit value in conjunction with the SAMPTIMEDIV bits and the ADC clock period determines the sample and hold period, as shown in the description for the SAMPTIMEDIV bits.
		0000b	Conversion clock = ADC clock divided by 2.
		0001b	Conversion clock = ADC clock divided by 4.
		0010b	Conversion clock = ADC clock divided by 6.
		0011b	Conversion clock = ADC clock divided by 8.
		0100b	Conversion clock = ADC clock divided by 10.
		0101b	Conversion clock = ADC clock divided by 12.
		0110b	Conversion clock = ADC clock divided by 14.
		0111b	Conversion clock = ADC clock divided by 16.
		1000b	Conversion clock = ADC clock divided by 18.
		1001b	Conversion clock = ADC clock divided by 20.
		1010b	Conversion clock = ADC clock divided by 22.
		1011b	Conversion clock = ADC clock divided by 24.
		1100b	Conversion clock = ADC clock divided by 26.
		1101b	Conversion clock = ADC clock divided by 28.
		1110b	Conversion clock = ADC clock divided by 30.
		1111b	Conversion clock = ADC clock divided by 32.

5.4 ADC Clock Control Register (ADCCLKCTL)

The ADC clock control register (see Figure 6 and Table 5) is a read/write register that holds the divider value for the CPU clock and determines whether the ADC can be put into a low-power (idle) state.

Figure 6. ADC Clock Control Register (ADCCLKCTL)

15	9	8	7	0
Reserved		IDLEEN	CPUCLKDIV	
R-0		R/W-0	R/W-07h	

Legend: R = Read; W = Write; -*n* = Value after reset

Table 5. ADC Clock Control Register (ADCCLKCTL) Field Values

Bit	Field	Value	Description
15-9	Reserved		These reserved bits are always read as 0s.
8	IDLEEN		Idle enable bit for the ADC. A number of domains in the DSP can be individually turned on (made active) or turned off (made idle). The ADC is part of the peripherals domain. The IDLEEN bit determines whether the ADC becomes idle when the peripherals domain becomes idle.
		0	The ADC cannot be placed in the idle state.
		1	If the peripherals domain is idle, the ADC is in the idle state.
7-0	CPUCLKDIV	0-255	CPU clock divider bits. These bits indicate the divider rate of the CPU clock.
			ADC Clock = (CPU Clock) / (CPUCLKDIV + 1)

6 Conversion Example

The clock dividers are used to derive the total conversion time from the CPU clock within the DSP. The following example shows through the programming of these clock dividers how to obtain the maximum sampling frequency in a system where the DSP operates at 144 MHz. After the ADC sampling rate has been programmed, the DSP can begin using the ADC for sampling analog inputs.

1) Divide down the CPU clock to generate the main clock to the ADC (ADC clock). It is desirable to program the ADC clock to as low a frequency as possible to minimize the power consumption of the ADC state machine.

   In this example, program the ADC clock to 4 MHz. To obtain the 4 MHz value, the CPU clock of 144 MHz must be divided by 36. An 8-bit divider, CPUCLKDIV bits in ADCCLKCTL, is provided. The calculation follows:

   ADC Clock = (CPU Clock) / (CPUCLKDIV + 1)
   ADC Clock = (144 MHz) / (CPUCLKDIV + 1)
   ADC Clock = (144 MHz) / (35 + 1) = 4 MHz

ADC Clock Control Register (ADCCLKCTL)

15 9	8	7 0
Reserved	IDLEEN	CPUCLKDIV = 0010 0011

The 4-MHz ADC clock is now divided to generate the two components of the total conversion time.

2) Divide down the 4-MHz ADC clock to generate the conversion clock.

   In this example, program the conversion clock rate divider to generate the maximum possible conversion clock frequency of 2 MHz. To obtain the 2-MHz conversion clock frequency, the ADC clock must be divided by the lowest value. A 5-bit divider, CONVRATEDIV bits in ADCCLKDIV, is provided. The calculation follows:

   ADC Conversion Clock = (ADC Clock) / (2 $\times$ (CONVRATEDIV + 1))
   ADC Conversion Clock = (4 MHz) / (2 $\times$ (CONVRATEDIV + 1))
   ADC Conversion Clock = (4 MHz) / (2 $\times$ (0 + 1)) = 2 MHz

ADC Clock Divider Register (ADCCLKDIV)

15 8	7 4	3 0
SAMPTIMEDIV	Reserved	CONVRATEDIV = 0000

The actual conversion time is 13 cycles of this clock, so the conversion time is 6.5 μs. The calculation follows:

ADC Conversion Time = 13 $\times$ (1 / ADC Conversion Clock)
ADC Conversion Time = 13 $\times$ (1 / (2 MHz)) = 6.5 μs

3) Program the clock divider for the sample and hold time. The sample and hold time must be greater than or equal to 40 µs.

   In this example, program the sample and hold time to 40 µs. An 8-bit divider, SAMPTIMEDIV bits in ADCCLKDIV, is used in conjunction with the conversion rate divider to obtain the sample and hold time from the ADC clock. The calculation follows:

   ADC Sample and Hold Period =
   (1 / (ADC Clock)) / (2 × (CONVRATEDIV + 1 + SAMPTIMEDIV))
   = (1 / (4 MHz)) / (2 × (0 + 1 + SAMPTIMEDIV))
   = 250 ns × (2 × (0 + 1 + 79)) = 40 µs

ADC Clock Divider Register (ADCCLKDIV)

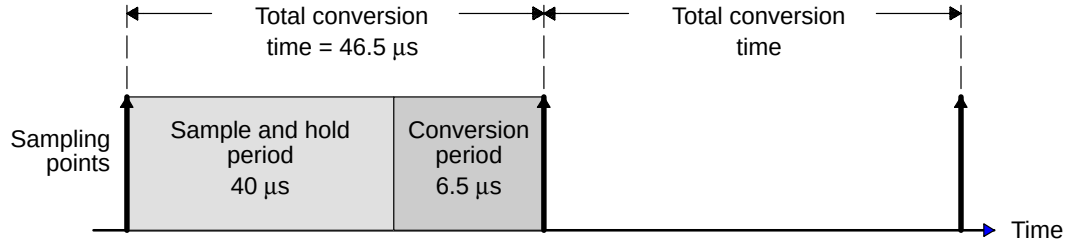
15 – 8	7 – 4	3 – 0
SAMPTIMEDIV= 0100 1111	Reserved	CONVRATEDIV = 0000

4) The final results of this example are summarized here and in Figure 7. The total conversion time is composed of a 40-µs sample and hold time plus a 6.5-µs conversion time. A new conversion can begin every 46.5 µs, giving a maximum sampling rate of 21.5 kHz. The calculations follow:

   Total Conversion Time = (Sample and Hold Period) + (Conversion Period)
   Total Conversion Time = 40 µs + 6.5 µs = 46.5 µs

   Sampling Frequency = 1 / (Total Conversion Time)
   Sampling Frequency = 1 / 46.5 µs = 21.5 kHz

Figure 7. Total Conversion Time for the Conversion Example



Revision History

This document was revised to SPRU586B from SPRU586A, which was released in November 2003.

The scope of this revision was limited to adding support for the TMS320VC5507 device.

The following changes were made in this revision:

Page	Additions/Modifications/Deletions
Global	Added the TMS320VC5507 device.

Index

A

ADC
- block diagram 7
- conversion example 16
- introduction 7
- registers 10
- total conversion time diagram 8

ADC busy bit (ADCBusy) 12

ADC channel select bits (ChSelect)
- of ADCR 10
- of ADDR 12

ADC clock enable bit (IDLEEN) 15

ADC conversion clock rate divider bits (ConvRateDiv) 13

ADC data bits (ADCData) 12

ADC sample and hold time divider bits (SampTimeDiv) 13

ADC start conversion bit (ADCStart) 10

ADC system clock divider bits (SystemClkDiv) 15

ADCBusy bit of ADDR 12

ADCCR 15

ADCData bits of ADDR 12

ADCDR 13

ADCR 10

ADCStart bit of ADCR 10

ADDR 11

C

ChSelect bits
- of ADCR 10
- of ADDR 12

clock control register of ADC (ADCCR) 15

clock divider register of ADC (ADCDR) 13

control register of ADC (ADCR) 10

conversion example for ADC 16

ConvRateDiv bits of ADCDR 13

D

data register of ADC (ADDR) 11

diagrams
- ADC 7
- ADC total conversion time 8

I

IDLEEN bit of ADCCR 15

R

registers, ADC 10

S

SampTimeDiv bits of ADCDR 13

SystemClkDiv bits of ADCCR 15