Preliminary

TMS320x2802x Piccolo Analog-to-Digital Converter and Comparator

Reference Guide

Literature Number: SPRUGE5
December 2008

Contents

List of Figures

List of Tables

Read This First

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h or with a leading 0x. For example, the following number is 40 hexadecimal (decimal 64): 40h or 0x40.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Related Documents From Texas Instruments

The following documents are available for download from the Texas Instruments website, www.ti.com.

Data Manual—

SPRS523— TMS320F28022, 28023, 28024, 28025, 28026, 28027 Microcontrollers (MCUs) contains the pinout, signal descriptions, as well as electrical and timing specifications for the 2802x devices.

SPRZ292— TMS320x28022, 28023, 28024, 28025, 28026, 28027 Piccolo DSP Silicon Errata describes known advisories on silicon and provides workarounds.

CPU User's Guides—

SPRU430— TMS320C28x DSP CPU and Instruction Set Reference Guide describes the central processing unit (CPU) and the assembly language instructions of the TMS320C28x fixed-point digital signal processors (DSPs). It also describes emulation features available on these DSPs.

Peripheral Guides—

SPRU566— TMS320x28xx, 28xxx DSP Peripheral Reference Guide describes the peripheral reference guides of the 28x digital signal processors (DSPs).

SPRUFN3— TMS320x2802x Piccolo System Control and Interrupts Reference Guide describes the various interrupts and system control features of the C2802x microcontrollers (MCUs).

SPRUFN6— TMS320x2802x Piccolo Boot ROM Reference Guide describes the purpose and features of the bootloader (factory-programmed boot-loading software) and provides examples of code. It also describes other contents of the device on-chip boot ROM and identifies where all of the information is located within that memory.

SPRUGE5— TMS320x2802x Piccolo Analog-to-Digital Converter (ADC) and Comparator Reference Guide describes how to configure and use the on-chip ADC module, which is a 12-bit pipelined ADC.

SPRUGE9— TMS320x2802x Piccolo Enhanced Pulse Width Modulator (ePWM) Module Reference Guide describes the main areas of the enhanced pulse width modulator that include digital motor control, switch mode power supply control, UPS (uninterruptible power supplies), and other forms of power conversion.

SPRUGE8— TMS320x2802x Piccolo High-Resolution Pulse Width Modulator (HRPWM) describes the operation of the high-resolution extension to the pulse width modulator (HRPWM).

SPRUGH1— TMS320x2802x Piccolo Serial Communications Interface (SCI) Reference Guide describes how to use the SCI.

SPRUFZ8— TMS320x2802x Piccolo Enhanced Capture (eCAP) Module Reference Guide describes the enhanced capture module. It includes the module description and registers.

SPRUG71— TMS320x2802x Piccolo Serial Peripheral Interface (SPI) Reference Guide describes the SPI - a high-speed synchronous serial input/output (I/O) port - that allows a serial bit stream of programmed length (one to sixteen bits) to be shifted into and out of the device at a programmed bit-transfer rate.

SPRUFZ9— TMS320x2802x Piccolo Inter-Integrated Circuit (I2C) Reference Guide describes the features and operation of the inter-integrated circuit (I2C) module.

Tools Guides—

SPRU513— TMS320C28x Assembly Language Tools User's Guide describes the assembly language tools (assembler and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for the TMS320C28x device.

SPRU514— TMS320C28x Optimizing C Compiler User's Guide describes the TMS320C28x™ C/C++ compiler. This compiler accepts ANSI standard C/C++ source code and produces TMS320 DSP assembly language source code for the TMS320C28x device.

SPRU608— The TMS320C28x Instruction Set Simulator Technical Overview describes the simulator, available within the Code Composer Studio for TMS320C2000 IDE, that simulates the instruction set of the C28x™ core.

Reference Guide
SPRUGE5–December 2008

Analog-to-Digital Converter and Comparator

The ADC module described in this reference guide is a Type 3 ADC and exists on the Piccolo™ family of devices. The Comparator function described in this reference guide is a Type 0 Comparator. See the *TMS320C28xx, 28xxx DSP Peripheral Reference Guide* (SPRU566) for a list of all devices with modules of the same type, to determine the differences between the types, and for a list of device-specific differences within a type.

1 Analog-to-Digital Converter (ADC)

The ADC module described in this reference guide is a 12-bit recyclic ADC; part SAR, part pipelined. The analog circuits of this converter, referred to as the "core" in this document, include the front-end analog multiplexers (MUXs), sample-and-hold (S/H) circuits, the conversion core, voltage regulators, and other analog supporting circuits. Digital circuits, referred to as the "wrapper" in this document, include programmable conversions, result registers, interface to analog circuits, interface to device peripheral bus, and interface to other on-chip modules.

1.1 Features

The core of the ADC contains a single 12-bit converter fed by two sample and hold circuits. The sample and hold circuits can be sampled simultaneously or sequentially. These, in turn, are fed by a total of up to 16 analog input channels. See the device datasheet for the specific number of channels available. The converter can be configured to run with an internal bandgap reference to create true-voltage based conversions or with a pair of external voltage references (VREFHI/LO) to create ratiometric based conversions.

Contrary to previous ADC types, this ADC is not sequencer based. It is easy for the user to create a series of conversions from a single trigger. However, the basic principle of operation is centered around the configurations of individual conversions, called SOC's, or Start-Of-Conversions.

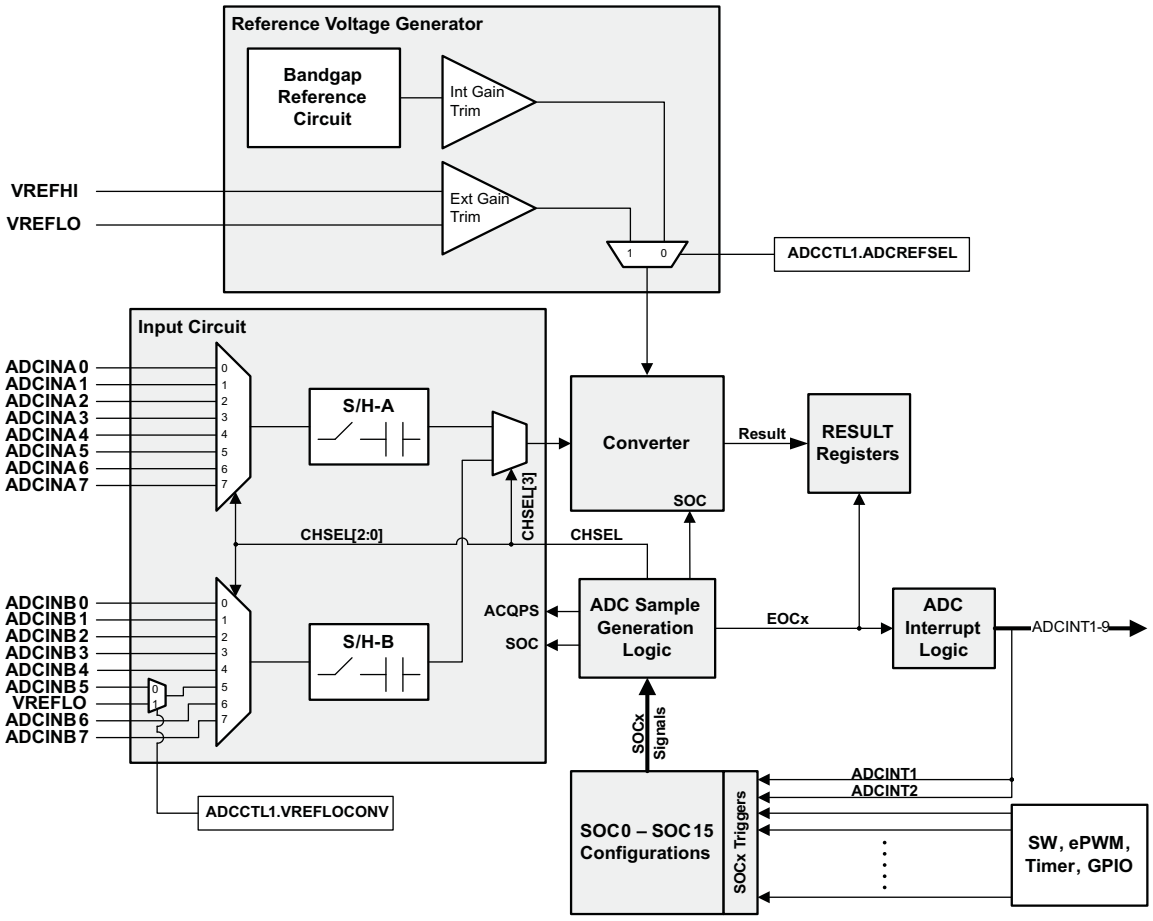
Functions of the ADC module include:

- 12-bit ADC core with built-in dual sample-and-hold (S/H)
- Simultaneous sampling or sequential sampling modes
- Full range analog input: 0 V to 3.3 V fixed, or VREFHI/VREFLO ratiometric
- Runs at full system clock, no prescaling required
- Up to 16-channel, multiplexed inputs
- 16 SOC's, configurable for trigger, sample window, and channel
- 16 result registers (individually addressable) to store conversion values
- Multiple trigger sources
  - S/W - software immediate start
  - ePWM 1-7
  - GPIO XINT2
  - CPU Timers 0/1/2
  - ADCINT1/2
- 9 flexible PIE interrupts, can configure interrupt request after any conversion

1.2 Block Diagram

Figure 1 shows the block diagram of the ADC module.

Figure 1. ADC Block Diagram

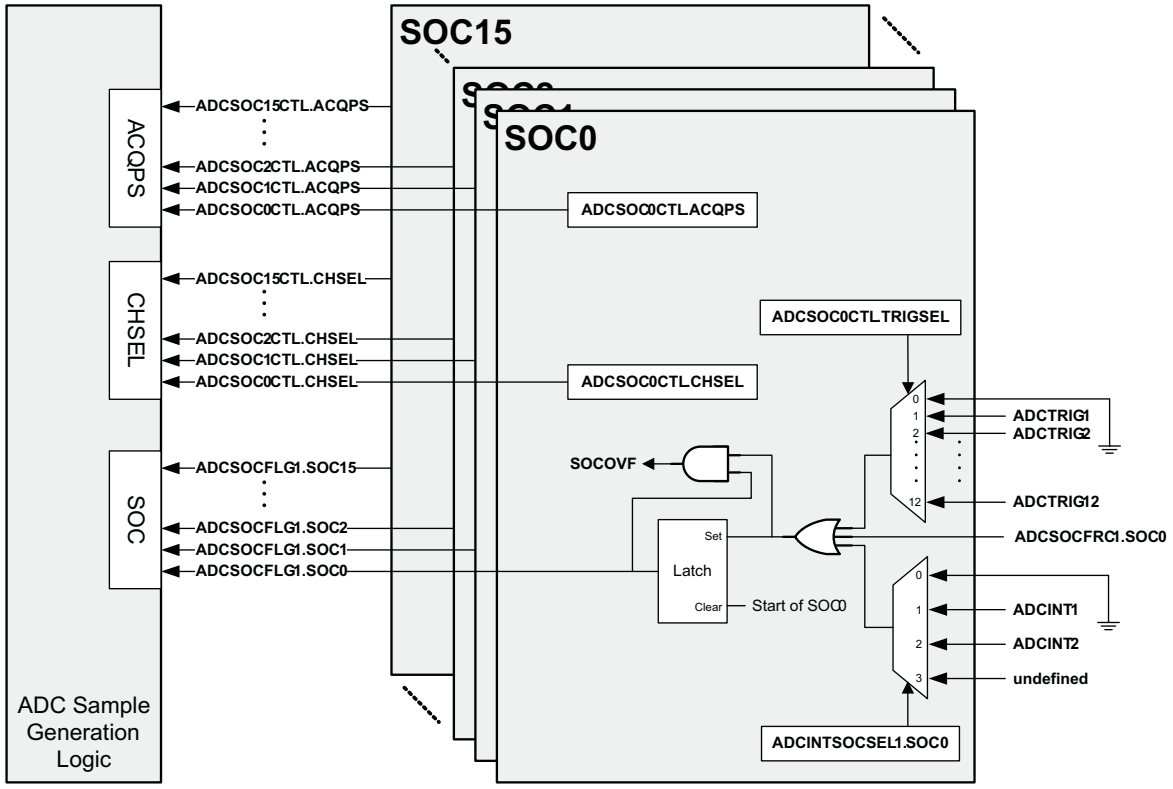


1.3 SOC Principle of Operation

Contrary to previous ADC types, this ADC is not sequencer based. Instead, it is SOC based. The term SOC is configuration set defining the single conversion of a single channel. In that set there are three configurations: the trigger source that starts the conversion, the channel to convert, and the acquisition (sample) window size. Each SOC is independently configured and can have any combination of the trigger, channel, and sample window size available. Multiple SOC's can be configured for the same trigger, channel, and/or acquisition window as desired. This provides a very flexible means of configurating conversions ranging from individual samples of different channels with different triggers, to oversampling the same channel using a single trigger, to creating your own series of conversions of different channels all from a single trigger.

The trigger source for SOCx is configured by a combination of the TRIGSEL field in the ADCSOCxCTL register and the appropriate bits in the ADCINTSOCSEL1 or ADCINTSOCSEL2 register. Software can also force an SOC event with the ADCSOCFRC1 register. The channel and sample window size for SOCx are configured with the CHSEL and ACQPS fields of the ADCSOCxCTL register.

Figure 2. SOC Block Diagram



For example, to configure a single conversion on channel ADCINA1 to occur when the ePWM3 timer reaches its period match you must first setup ePWM3 to output an SOCA or SOCB signal on a period match. See the TMS320x2802x Piccolo Enhanced Pulse Width Modulator Module User's Guide (SPRUGE9) on how to do this. In this case, we'll use SOCA. Then, setup one of the SOC's using its ADCSOCxCTL register. It makes no difference which SOC we choose, so we'll use SOC0. The fastest allowable sample window for the ADC is 7 cycles. Choosing the fastest time for the sample window, channel ADCINA1 for the channel to convert, and ePWM3 for the SOC0 trigger, we'll set the ACQPS field to 6, the CHSEL field to 1, and the TRIGSEL field to 9, respectively. The resulting value written into the register will be:

ADCSOC0CTL = 4846h; // (ACQPS=6, CHSEL=1, TRIGSEL=9)

When configured as such, a single conversion of ADCINA1 will be started on an ePWM3 SOCA event with the resulting value stored in the ADCRESULT0 register.

If instead ADCINA1 needed to be oversampled by 3X, then SOC1, SOC2, and SOC3 could all be given the same configuration as SOC0.

ADCSOC1CTL = 4846h; // (ACQPS=6, CHSEL=1, TRIGSEL=9)

ADCSOC2CTL = 4846h; // (ACQPS=6, CHSEL=1, TRIGSEL=9)

ADCSOC3CTL = 4846h; // (ACQPS=6, CHSEL=1, TRIGSEL=9)

When configured as such, four conversions of ADCINA1 will be started in series on an ePWM3 SOCA event with the resulting values stored in the ADCRESULT0 – ADCRESULT3 registers.

Another application may require 3 different signals to be sampled from the same trigger. This can be done by simply changing the CHSEL field for SOC0-SOC2 while leaving the TRIGSEL field unchanged.

```
ADCSOC0CTL = 4846h;   // (ACQPS=6, CHSEL=1, TRIGSEL=9)
ADCSOC1CTL = 4886h;   // (ACQPS=6, CHSEL=2, TRIGSEL=9)
ADCSOC2CTL = 48C6h;   // (ACQPS=6, CHSEL=3, TRIGSEL=9)
```

When configured this way, three conversions will be started in series on an ePWM3 SOCA event. The result of the conversion on channel ADCINA1 will show up in ADCRESULT0. The result of the conversion on channel ADCINA2 will show up in ADCRESULT1. The result of the conversion on channel ADCINA3 will show up in ADCRESULT2. The channel converted and the trigger have no bearing on where the result of the conversion shows up. The RESULT register is associated with the SOC.

Note: These examples are incomplete. Clocks must be enabled via the PCLKCR0 register and the ADC must be powered to work correctly. For a description of the PCLKCR0 register see the TMS320F2802x Piccolo System Control and Interrupts Reference Guide (SPRUFN3). For the power up sequence of the ADC, see Section 1.7.

1.3.1 ADC Acquisition (Sample and Hold) Window

External drivers vary in their ability to drive an analog signal quickly and effectively. Some circuits require longer times to properly transfer the charge into the sampling capacitor of an ADC. To address this, the ADC supports control over the sample window length for each individual SOC configuration. Each ADCSOCxCTL register has a 6-bit field, ACQPS, that determines the sample and hold (S/H) window size. The value written to this field is one less than the number of cycles desired for the sampling window for that SOC. Thus, a value of 15 in this field will give 16 clock cycles of sample time. The minimum number of sample cycles allowed is 7 (ACQPS=6). The total sampling time is found by adding the sample window size to the conversion time of the ADC, 13 ADC clocks. Examples of various sample times are shown below in Table 1.

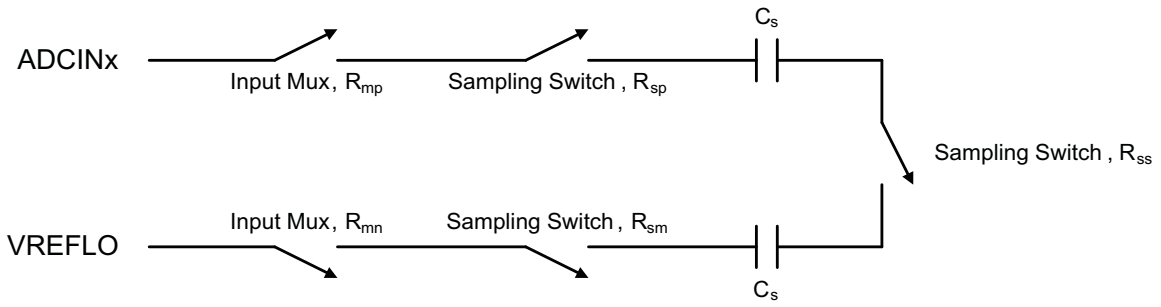
Table 1. Sample timings with different values of ACQPS

ADC Clock	ACQPS	Sample Window	Conversion Time (13 cycles)	Total Time to Process Analog Voltage(1)
60MHz	6	116.67ns	216.67ns	333.34ns
60MHz	8	150.00ns	216.67ns	366.67ns
60MHz	10	183.33ns	216.67ns	400.00ns
60MHz	14	250.00ns	216.67ns	466.67ns
60MHz	25	433.33ns	216.67ns	650.00ns
40MHz	6	175	325ns	500.00ns
40MHz	25	625	325ns	950.00ns

(1) The total times are for a single conversion and do not include pipelining effects that increase the average speed over time.

As shown in Figure 3, the ADCIN pins can be modeled as an RC circuit. With VREFLO connected to ground, a voltage swing from 0 to 3.3v on ADCIN yields a typical RC time constant of 2ns.

Figure 3. ADCINx Input Model



1.3.2 Trigger Operation

Each SOC can be configured to start on one of many input triggers. Multiple SOC's can be configured for the same channel if desired. Following is a list of the available input triggers:

- Software
- CPU Timers 0/1/2 interrupts
- XINT2 SOC
- ePWM1-7 SOCA and SOCB

See the ADCSOCxCTL Register Bit Definitions for the configuration details of these triggers.

Additionally ADCINT1 and ADCINT2 can be fed back to trigger another conversion. This configuration is controlled in the ADCINTSOCSEL1/2 registers. This mode is useful if a continuous stream of conversions is desired. See section 1.6 for information on the ADC interrupt signals.

1.3.3 Channel Selection

Each SOC can be configured to convert any of the available ADCIN input channels. When an SOC is configured for sequential sampling mode, the four bit CHSEL field of the ADCSOCxCTL register defines which channel to convert. When an SOC is configured for simultaneous sampling mode, the most significant bit of the CHSEL field is dropped and the lower three bits determine which pair of channels are converted.

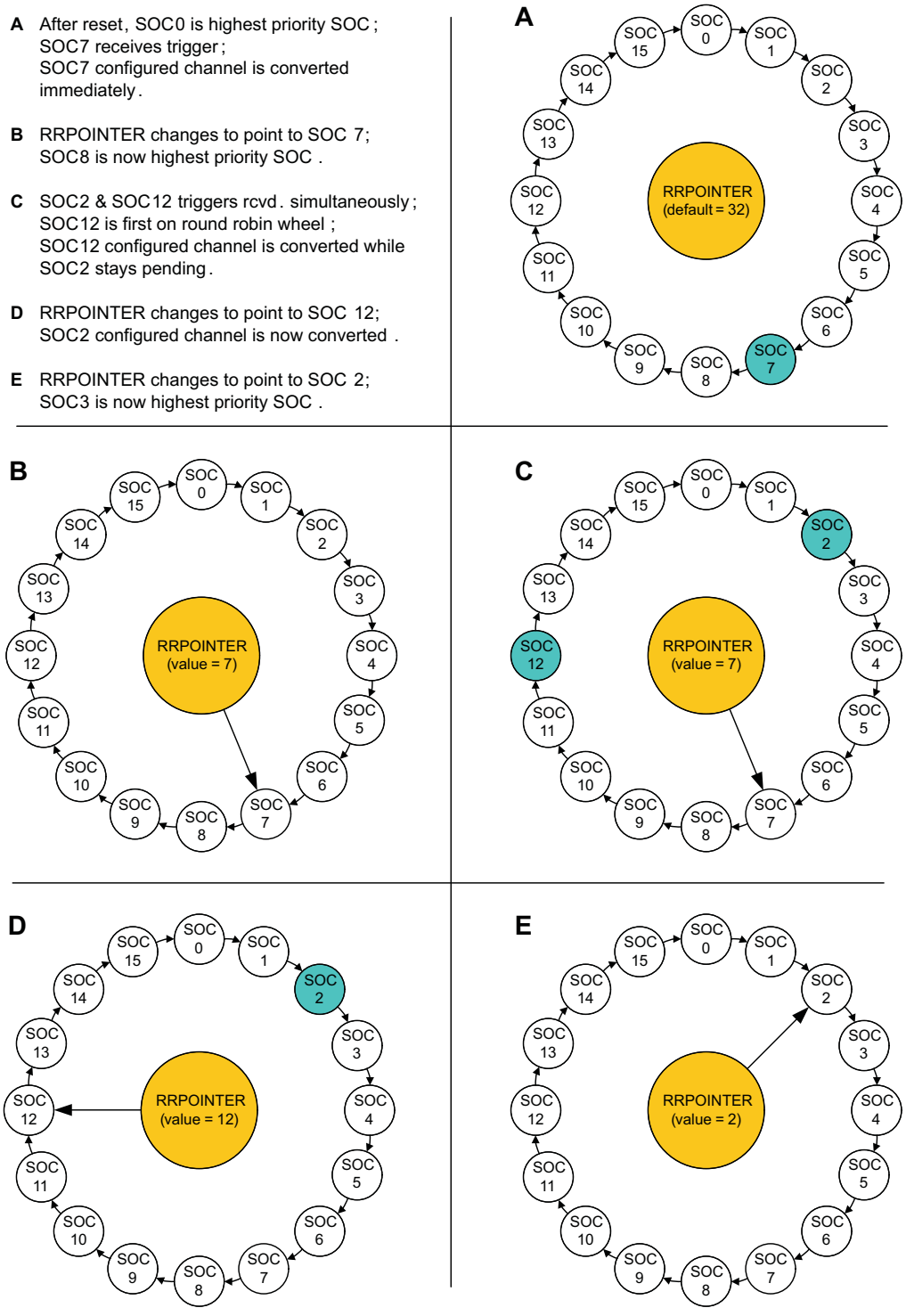
ADCINA0 is shared with VREFHI, and therefore cannot be used as a variable input source when using external reference voltage mode. See Section 1.9 for details on this mode.

1.4 ADC Conversion Priority

When multiple SOC flags are set at the same time, one of two forms of priority determines the order in which they are converted. The default priority method is round robin. In this scheme, no SOC has an inherent higher priority than another. Priority depends on the round robin pointer (RRPOINTER). The RRPOINTER reflected in the ADCSOCPRIORITYCTL register points to the last SOC converted. The highest priority SOC is given to the next value greater than the RRPOINTER value, wrapping around back to SOC0 after SOC15. At reset the value is 32 since 0 indicates a conversion has already occurred. When RRPOINTER equals 32 the highest priority is given to SOC0. The RRPOINTER is reset by a device reset, when the ADCCTL1.RESET bit is set, or when the SOCPRICTL register is written.

An example of the round robin priority method is given in Figure 4.

Figure 4. Round Robin Priority Example

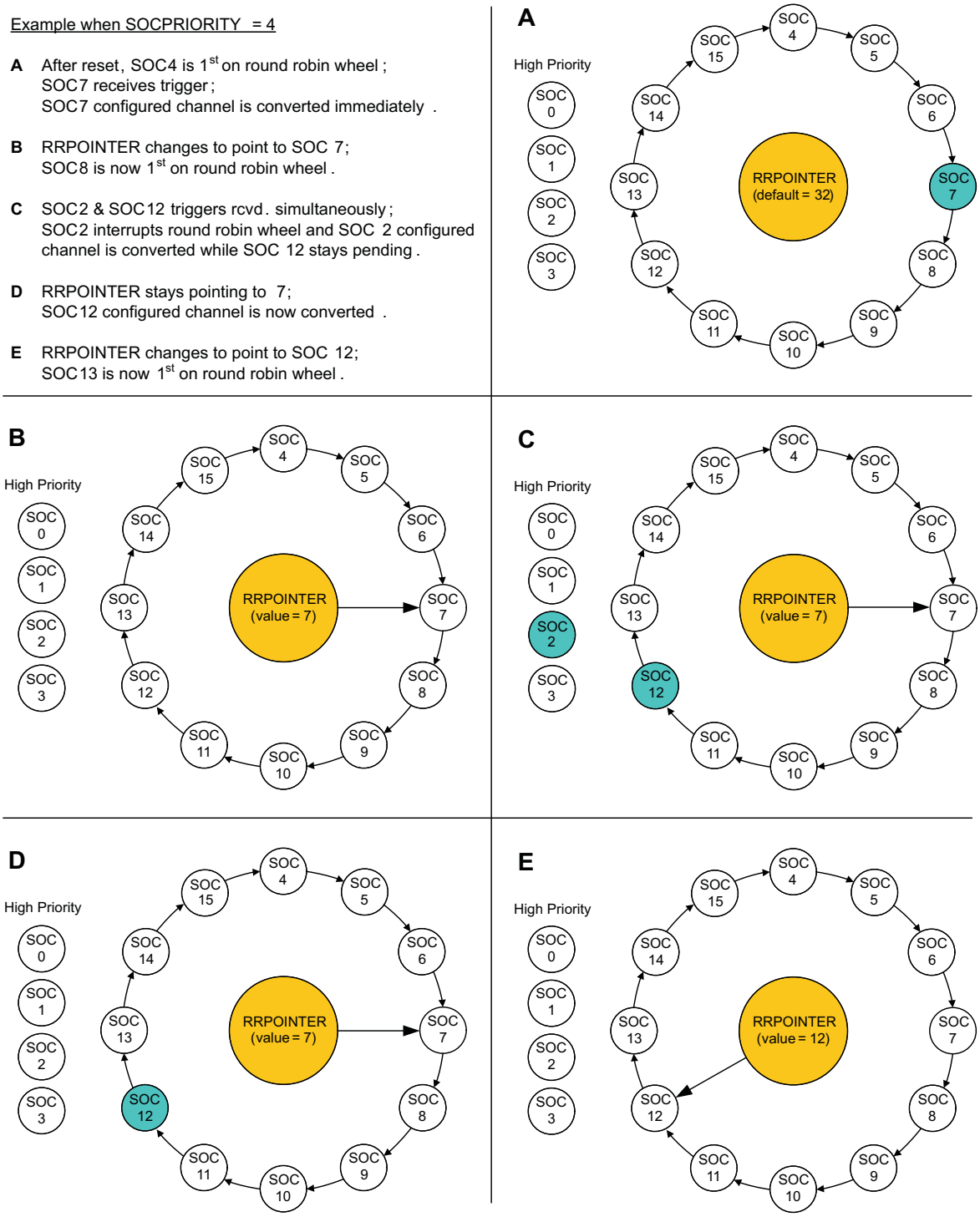


The SOCPRIORITY field in the ADCSOCPRIORITYCTL register can be used to assign high priority from a single to all of the SOC's. When configured as high priority, an SOC will interrupt the round robin wheel after any current conversion completes and insert itself in as the next conversion. After its conversion completes, the round robin wheel will continue where it was interrupted. If two high priority SOC's are triggered at the same time, the SOC with the lower number will take precedence.

High priority mode is assigned first to SOC0, then in increasing numerical order. The value written in the SOCPRIORITY field defines the first SOC that is not high priority. In other words, if a value of 4 is written into SOCPRIORITY, then SOC0, SOC1, SOC2, and SOC3 are defined as high priority, with SOC0 the highest.

An example using high priority SOC's is given in Figure 5.

Figure 5. High Priority Example



1.5 Simultaneous Sampling Mode

In some applications it is important to keep the delay between the sampling of two signals minimal. The ADC contains dual sample and hold circuits to allow two different channels to be sampled simultaneously. Simultaneous sampling mode is configured for a pair of SOCx's with the ADCSAMPLEMODE register. The even numbered SOCx and the following odd numbered SOCx (i.e., SOC0 and SOC1) are coupled together with one enable bit (SIMULEN0, in this case). The coupling behavior is as follows:

- Either SOCx's trigger will start a pair of conversions.
- The pair of channels converted will consist of the A-channel and the B-channel corresponding to the value of the CHSEL field of the triggered SOCx. The valid values in this mode are 0-7.
- Both channels will be sampled simultaneously.
- The A channel will always convert first.
- The even EOCx pulse will be generated based off of the A-channel conversion, the odd EOCx pulse will be generated off of the B-channel conversion. See Section 1.6 for an explanation of the EOCx signals.
- The result of the A-channel conversion is placed in the even ADCRESULTx register and the result of the B-channel conversion is written to the odd ADCRESULTx register.

For example, if the ADCSAMPLEMODE.SIMULEN0 bit is set, and SOC0 is configured as follows:

CHSEL = 2 (ADCINA2/ADCINB2 pair)

TRIGSEL = 5 (ADCTRIG5 = ePWM1.ADCSOCA)

When the ePWM1 sends out an ADCSOCA trigger, both ADCINA2 and ADCINB2 will be sampled simultaneously (assuming priority). Immediately after, the ADCINA2 channel will be converted and its value will be stored in the ADCRESULT0 register. Depending on the ADCCTL1.INTPULSEPOS setting, the EOC0 pulse will either occur when the conversion of ADCINA2 begins or completes. Then the ADCINB2 channel will be converted and its value will be stored in the ADCRESULT1 register. Depending on the ADCCTL1.INTPULSEPOS setting, the EOC1 pulse will either occur when the conversion of ADCINB2 begins or completes.

Typically in an application it is expected that only the even SOCx of the pair will be used. However, it is possible to use the odd SOCx instead, or even both. In the latter case, both SOCx triggers will start a conversion. Therefore, caution is urged as both SOCx's will store their results to the same ADCRESULTx registers, possibly overwriting each other.

The rules of priority for the SOCx's remain the same as in sequential sampling mode.

Section 1.11 shows the timing of simultaneous sampling mode.

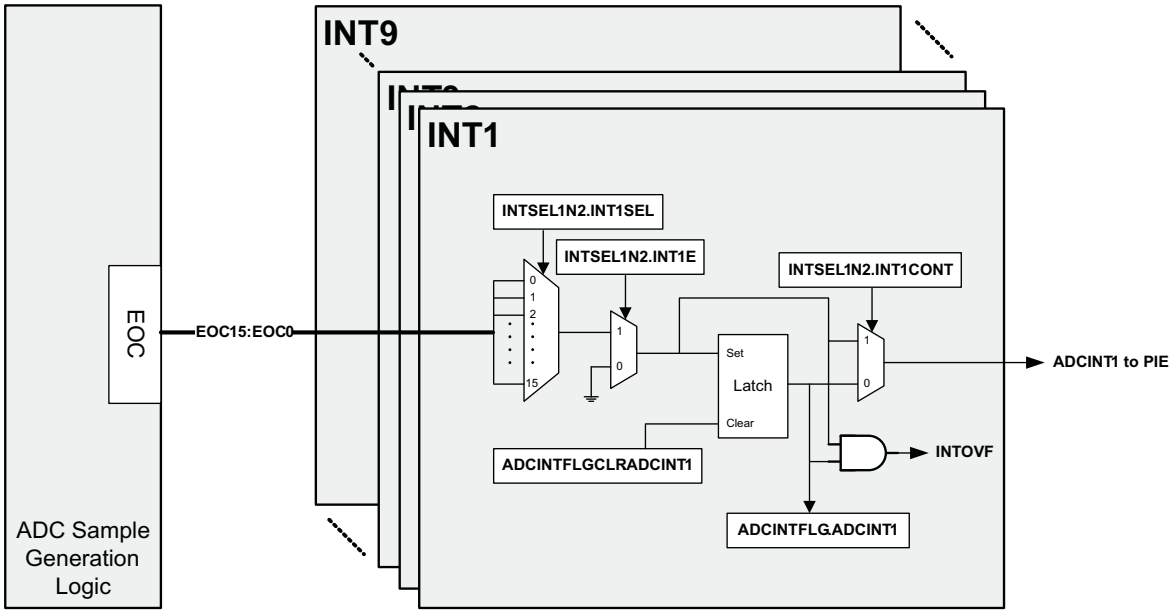
1.6 EOC and Interrupt Operation

Just as there are 16 independent SOCx configuration sets, there are 16 EOCx pulses. In sequential sampling mode, the EOCx is associated directly with the SOCx. In simultaneous sampling mode, the even and the following odd EOCx pair are associated with the even and the following odd SOCx pair, as described in Section 1.5. Depending on the ADCCTL1.INTPULSEPOS setting, the EOCx pulse will occur either at the beginning of a conversion or the end. See section 1.11 for exact timings on the EOCx pulses.

The ADC contains 9 interrupts that can be flagged and/or passed on to the PIE. Each of these interrupts can be configured to accept any of the available EOCx signals as its source. The configuration of which EOCx is the source is done in the INTSELxNy registers. Additionally, the ADCINT1 and ADCINT2 signals can be configured to generate an SOCx trigger. This is beneficial to creating a continuous stream of conversions.

Figure 6 shows a block diagram of the interrupt structure of the ADC.

Figure 6. Interrupt Structure



1.7 Power Up Sequence

The ADC resets to the ADC off state. Before writing to any of the ADC registers the ADCENCLK bit in the PCLKCR0 register must be set. For a description of the PCLKCR0 register see the TMS320F2802x Piccolo System Control Reference Guide (SPRUFN3). When powering up the ADC, use the following sequence:

1. If an external reference is desired, enable this mode using bit 3 (ADCREFSEL) in the ADCCTL1 register.
2. Power up the reference, bandgap, and analog circuits together by setting bits 7-5 (ADCPWDN, ADCBGPWD, ADCREFPWD) in the ADCCTL1 register. Intermediary states are not currently supported.
3. Enable the ADC by setting bit 14 (ADCENABLE) of the ADCCTL1 register.
4. Before performing the first conversion, a delay of 1 millisecond after step 2 is required.

Alternatively, steps 1 through 3 can be performed simultaneously.

When powering down the ADC, all three bits in step 2 can be cleared simultaneously. The ADC power levels must be controlled via software and they are independent of the state of the device power modes.

Note: This type ADC requires a 1ms delay after all of the circuits are powered up. This differs from the previous type ADC's.

1.8 ADC Calibration

Inherent in any converter is a zero offset error and a full scale gain error. The ADC is factory calibrated at 25-degrees Celsius to correct both of these while allowing the user to modify the offset correction for any application environmental effects, such as the ambient temperature. Except under certain emulation conditions, or unless a modification from the factory settings is desired, the user is not required to perform any specific action. The ADC will be properly calibrated during the device boot process.

1.8.1 Factory Settings and Calibration Function

During the fabrication and test process Texas Instruments calibrates several ADC settings along with a couple of internal oscillator settings. These settings are embedded into the TI reserved OTP memory as part of a C-callable function named Device_cal(). Called during the startup boot procedure in the Boot ROM this function writes the factory settings into their respective active registers. Until this occurs, the ADC and the internal oscillators will not adhere to their specified parameters. If the boot process is skipped during emulation, the user must ensure the trim settings are written to their respective registers to ensure the ADC and the internal oscillators meet the specifications in the datasheet. This can be done either by calling this function manually or in the application itself, or by a direct write via CCS. A gel function is provided as part of the *C2802x C/C++ Header Files and Peripheral Examples* (SPRC823) to accomplish this.

For more information on the Device_cal() function refer to the *TMS320x2802x Boot ROM Reference Guide* (SPRUFN6).

Texas Instruments cannot guarantee the parameters specified in the datasheet if a value other than the factory settings contained in the TI reserved OTP memory is written into the ADC trim registers.

1.8.2 ADC Zero Offset Calibration

Zero offset error is defined as the resultant digital value that occurs when converting a voltage at VREFLO. This base error affects all conversions of the ADC and together with the full scale gain and linearity specifications, determine the DC accuracy of a converter. The zero offset error can be positive, meaning that a positive digital value is output when VREFLO is presented, or negative, meaning that a voltage higher than a one step above VREFLO still reads as a digital zero value. To correct this error, the two's complement of the error is written into the ADCOFFTRIM register. The value contained in this register will be applied before the results are available in the ADC result registers. This operation is fully contained within the ADC core, so the timing for the results will not be affected and the full dynamic range of the ADC will be maintained for any trim value. Calling the Device_cal() function writes the ADCOFFTRIM register with the factory calibrated offset error correction, but the user can modify the ADCOFFTRIM register to compensate for additional offset error induced by the application environment. This can be done without sacrificing an ADC channel by using the VREFLOCONV bit in the ADCCTRL1 register.

Use the following procedure to re-calibrate the ADC offset:

1. **Set ADCOFFTRIM to 80 (50h)**. This adds an artificial offset to account for negative offset that may reside in the ADC core.
2. **Set ADCCTL1.VREFLOCONV to 1.** This internally connects VREFLO to input channel B5. See the ADCCTL1 register description for more details.
3. **Perform multiple conversions on B5 (i.e. sample VREFLO) and take an average to account for board noise.** See Section 1.3 on how to setup and initiate the ADC to sample B5.
4. **Set ADCOFFTRIM to 80 (50h) minus the average obtained in step 3.** This removes the artificial offset from step 1 and creates a two's compliment of the offset error.

1.8.3 ADC Full Scale Gain Calibration

Gain error occurs as an incremental error as the voltage input is increased. Full scale gain error occurs at the maximum input voltage. As in offset error, gain error can be positive or negative. A positive full scale gain error means that the full scale digital result is reached before the maximum voltage is input. A negative full scale error implies that the full digital result will never be achieved. The calibration function Device_cal() writes a factory trim value to correct the ADC full scale gain error into the ADCREFTRIM register. This register should not be modified after the Device_cal() function is called.

1.8.4 ADC Bias Current Calibration

To further increase the accuracy of the ADC, the calibration function Device_cal() also writes a factory trim value to an ADC register for the ADC bias currents. This register should not be modified after the Device_cal() function is called.

1.9 Internal/External Reference Voltage Selection

1.9.1 Internal Reference Voltage

The ADC can operate in two different reference modes, selected by the ADCCTL1.ADCREFSEL bit. By default the internal bandgap is chosen to generate the reference voltage for the ADC. This will convert the voltage presented according to a fixed scale 0 to 3.3v range. The equation governing conversions in this mode is:

Digital Value = 0	when Input ≤ 0v
Digital Value = 4096 [(Input – VREFLO)/3.3v]	when 0v < Input < 3.3v
Digital Value = 4095,	when Input ≥ 3.3v

*All fractional values are truncated

**VREFLO must be tied to ground in this mode. This is done internally on some devices.

1.9.2 External Reference Voltage

To convert the voltage presented as a ratiometric signal, the external VREFHI/VREFLO pins should be chosen to generate the reference voltage. In contrast with the fixed 0 to 3.3v input range of the internal bandgap mode, the ratiometric mode has an input range from VREFLO to VREFHI. Converted values will scale to this range. For instance, if VREFLO is set to 0.5v and VREFHI is 3.0v, a voltage of 1.75v will be converted to the digital result of 2048. See the device datasheet for the allowable ranges of VREFLO and VREFHI. On some devices VREFLO is tied to ground internally, and hence limited to 0v. The equation governing the conversions in this mode is:

Digital Value = 0	when Input ≤ VREFLO
Digital Value = 4096 [(Input – VREFLO)/(VREFHI – VREFLO)]	when VREFLO < Input < VREFHI
Digital Value = 4095,	when Input ≥ VREFHI

*All fractional values are truncated

1.10 ADC Registers

This section contains the ADC registers and bit definitions with the registers grouped by function. All of the ADC registers are located in Peripheral Frame 2 except the ADCRESULTx registers, which are found in Peripheral Frame 0. See the device datasheet for specific addresses.

Table 2. ADC Configuration & Control Registers (AdcRegs and AdcResult):

Register Name	Address Offset	Size (x16)	Description
ADCCTL1	0x00	1	Control 1 Register(1)
ADCINTFLG	0x04	1	Interrupt Flag Register
ADCINTFLGCLR	0x05	1	Interrupt Flag Clear Register
ADCINTOVF	0x06	1	Interrupt Overflow Register
ADCINTOVFCLR	0x07	1	Interrupt Overflow Clear Register
ADCINTSEL1AND2	0x08	1	Interrupt 1 and 2 Selection Register(1)
ADCINTSEL3AND4	0x09	1	Interrupt 3 and 4 Selection Register(1)
ADCINTSEL5AND6	0x0A	1	Interrupt 5 and 6 Selection Register(1)
ADCINTSEL7AND8	0x0B	1	Interrupt 7 and 8 Selection Register(1)
ADCINTSEL9AND10	0x0C	1	Interrupt 9 Selection Register (reserved Interrupt 10 Selection)(1)
SOCPRICTL	0x10	1	SOC Priority Control Register(1)
ADCSAMPLEMODE	0x12	1	Sampling Mode Register(1)
ADCINTSOCSEL1	0x14	1	Interrupt SOC Selection 1 Register (for 8 channels)(1)
ADCINTSOCSEL2	0x15	1	Interrupt SOC Selection 2 Register (for 8 channels)(1)
ADCSOCFLG1	0x18	1	SOC Flag 1 Register (for 16 channels)
ADCSOCFRC1	0x1A	1	SOC Force 1 Register (for 16 channels)
ADCSOCOVF1	0x1C	1	SOC Overflow 1 Register (for 16 channels)
ADCSOCOVFCLR1	0x1E	1	SOC Overflow Clear 1 Register (for 16 channels)
ADCSOC0CTL - ADCSOC15CTL	0x20 - 0x2F	1	SOC0 Control Register to SOC15 Control Register(1)
ADCREFTRIM	0x40	1	Reference Trim Register(1)
ADCOFFTRIM	0x41	1	Offset Trim Register(1)
ADCREV – reserved	0x4F	1	Revision Register
ADCRESULT0 - ADCRESULT15	0x00 - 0x0F(2)	1	ADC Result 0 Register to ADC Result 15 Register

(1) This register is EALLOW protected.
(2) The base address of the ADCRESULT registers differs from the base address of the other ADC registers. In the header files, the ADCRESULT registers are found in the AdcResult register file, not AdcRegs.

1.10.1 ADC Control Register 1 (ADCCTL1)

Note: The following ADC Control Register is EALLOW protected.

Figure 7. ADC Control Register 1 (ADCCTL1) (Address Offset 00h)

15	14	13	12				8
RESET	ADCENABLE	ADCBSY	ADCBSYCHN				
R-0/W-1	R/W-0	R-0	R-0				

7	6	5	4	3	2	1	0
ADCPWN	ADCBGPWD	ADCREFPWD	Reserved	ADCREFSEL	INTPULSEPOS	VREFLO CONV	TEMPCONV
R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; R-0/W-1 = always read as 0, write 1 to set; -*n* = value after reset

Table 3. ADC Control Register 1 (ADCCTL1) Field Descriptions

Bit	Field	Value	Description
15	RESET		ADC module software reset. This bit causes a master reset on the entire ADC module. All register bits and state machines are reset to the initial state as occurs when the device reset pin is pulled low (or after a power-on reset). This is a one-time-effect bit, meaning this bit is self-cleared immediately after it is set to 1. Read of this bit always returns a 0. Also, the reset of ADC has a latency of two clock cycles (that is, other ADC control register bits should not be modified until two clock cycles after the instruction that resets the ADC.
		0	no effect
		1	Resets entire ADC module (bit is then set back to 0 by ADC logic)
			Note: The ADC module is reset during a system reset. If an ADC module reset is desired at any other time, you can do so by writing a 1 to this bit. After two clock cycles, you can then write the appropriate values to the ADCCTL1 register bits. Assembly code:
			MOV ADCCTL1, #1xxxxxxxxxxxxxxb ; Resets the ADC (RESET = 1)
			RPT #1|| NOP ; Delay two cycles
			MOV ADCCTL1, #0xxxxxxxxxxxxxxb ; Set to user-desired value
			Note: The second MOV is not required if the default configuration is sufficient.
14	ADCENABLE		ADC Enable
		0	ADC disabled (**does not** power down ADC)
		1	ADC Enabled. Musts set before an ADC conversion (recommend that it be set directly after setting ADC power-up bits
13	ADCBSY		ADC Busy
			Set when ADC SOC is generated, cleared per below. Used by the ADC state machine to determine if ADC is avaliable to sample.
			Sequential Mode: Cleared 4 ADC clocks after negative edge of S/H pulse
			Simultaneous Mode: Cleared 14 ADC clocks after negative edge of S/H pulse
		0	ADC is busy and cannot sample another channel
		1	ADC is available to sample next channel
12-8	ADCBSYCHN		Set when ADC SOC for current channel is generated
			When ADCBSY = 0: holds the value of the last converted channel
			When ADCBSY = 1: reflects channel currently being processed
		00h	ADCINA0 is currently processing or was last channel converted
		01h	ADCINA1 is currently processing or was last channel converted
		02h	ADCINA2 is currently processing or was last channel converted
		03h	ADCINA3 is currently processing or was last channel converted
		04h	ADCINA4 is currently processing or was last channel converted
		05h	ADCINA5 is currently processing or was last channel converted
		06h	ADCINA6 is currently processing or was last channel converted
		07h	ADCINA7 is currently processing or was last channel converted
		08h	ADCINB0 is currently processing or was last channel converted
		09h	ADCINB1 is currently processing or was last channel converted
		0Ah	ADCINB2 is currently processing or was last channel converted
		0Bh	ADCINB3 is currently processing or was last channel converted
		0Ch	ADCINB4 is currently processing or was last channel converted
		0Dh	ADCINB5 is currently processing or was last channel converted
		0Eh	ADCINB6 is currently processing or was last channel converted
		0Fh	ADCINB7 is currently processing or was last channel converted
		1xh	Invalid value
7	ADCPWDN		ADC power down (active low).
			This bit controls the power up and power down of all the analog circuitry inside the analog core except the bandgap and reference circuitry
		0	All analog circuitry inside the core except the bandgap and referencce circuitry is powered down
		1	The analog circuitry inside the core is powered up

Table 3. ADC Control Register 1 (ADCCTL1) Field Descriptions (continued)

Bit	Field	Value	Description
6	ADCBGPWD		Bandgap circuit power down (active low)
		0	Bandgap circuitry is powered down
		1	Bandgap buffer's curcuitry inside core is powered up
5	ADCREFPWD		Reference buffers circuit power down (active low)
		0	Reference buffers circuitry is powered down
		1	Reference buffers circuitry inside the core is powered up
4	Reserved	0	Reads return a zero; Writes have no effect.
3	ADCREFSEL		Internal/external reference select
		0	Internal Bandgap used for reference generation
		1	External VREFHI/VREFLO pins used for reference generation. On some devices the VREFHI pin is shared with ADCINA0. In this case ADCINA0 will not be available for conversions in this mode. On some devices the VREFLO pin is shared with VSSA. In this case the VREFLO voltage cannot be varied.
2	INTPULSEPOS		INT Pulse Generation control
		0	INT pulse generation occurs when ADC begins conversion (neg edge of sample pulse od the sampled signal)
		1	INT pulse generation occurs 1 cycle prior to ADC result latching into its result register
1	VREFLOCONV		VREFLO Convert. When enabled, internally connects VREFLO to the ADC channel B5 and disconnects the ADCINB5 pin from the ADC. Whether the pin ADCINB5 exists on the device does not affect this function. Any external circuitry on the ADCINB5 pin is unaffected by this mode.
		0	VREFLO connection disabled
		1	VREFLO internally connected to the ADC for sampling
0	TEMPCONV		Temperature Sensor Convert. Currently not supported. Reserved for possible temperature sensor in the future.
		0	Only valid setting.
		1	Invalid setting. Do not use.

1.10.2 ADC Interrupt Registers

Figure 8. ADC Interrupt Flag Register (ADCINTFLG) (Address Offset 04h)

15						9	8
Reserved							ADCINT9
R-0							R-0

7	6	5	4	3	2	1	0
ADCINT8	ADCINT7	ADCINT6	ADCINT5	ADCINT4	ADCINT3	ADCINT2	ADCINT1
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 4. ADC Interrupt Flag Register (ADCINTFLG) Field Descriptions

Bit	Field	Value	Description
15-9	Reserved	0	Reads return a zero; Writes have no effect.
8-0	ADCINTx (x = 9 to 1)		ADC Interrupt Flag Bits: Reading this bit indicates if an ADCINT pulse was generated
		0	No ADC interrupt pulse generated
		1	ADC Interrupt pulse generated
			If the ADC interrupt is placed in continuous mode (INTSELxNy register) then further interrupt pulses are generated whenever a selected EOC event occurs even if the flag bit is set. If the continuous mode is not enabled, then no further interrupt pulses are generated until the user clears this flag bit using the ADCINTFLGCLR register. Rather, an ADC interrupt overflow event occurs in the ADCINTOVF register.

Figure 9. ADC Interrupt Flag Clear Register (ADCINTFLGCLR) (Address Offset 05h)

15						9	8
Reserved							ADCINT9
R-0							R/W-0

7	6	5	4	3	2	1	0
ADCINT8	ADCINT7	ADCINT6	ADCINT5	ADCINT4	ADCINT3	ADCINT2	ADCINT1
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 5. ADC Interrupt Flag Clear Register (ADCINTFLGCLR) Field Descriptions

Bit	Field	Value	Description
15-9	Reserved	0	Reads return a zero; Writes have no effect.
8-0	ADCINTx (x = 9 to 1)		ADC interrupt Flag Clear Bit
		0	No action.
		1	Clears respective flag bit in the ADCINTFLG register. If software tries to set this bit on the same clock cycle that hardware tries to set the flag bit in the ADCINTFLG register, then hardware has priority and the ADCINTFLG bit will be set. In this case the overflow bit in the ADCINTOVF register will not be affected regardless of whether the ADCINTFLG bit was previously set or not.

Figure 10. ADC Interrupt Overflow Register (ADCINTOVF) (Address Offset 06h)

15						9	8
Reserved							ADCINT9
R-0							R-0

7	6	5	4	3	2	1	0
ADCINT8	ADCINT7	ADCINT6	ADCINT5	ADCINT4	ADCINT3	ADCINT2	ADCINT1
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 6. ADC Interrupt Overflow Register (ADCINTOVF) Field Descriptions

Bit	Field	Value	Description
15-9	Reserved	0	Reserved
8-0	ADCINT9 (x = 9 to 1)		ADC Interrupt Overflow Bits. Indicates if an overflow occurred when generating ADCINT pulses. If the respective ADCINTFLG bit is set and a selected additional EOC trigger is generated, then an overflow condition occurs. 0 No ADC interrupt overflow event detected. 1 ADC Interrupt overflow event detected. The overflow bit does not care about the continuous mode bit state. An overflow condition is generated irrespective of this mode selection.

Figure 11. ADC Interrupt Overflow Clear Register (ADCINTOVFCLR) (Address Offset 07h)

15	14	13	12	11	10	9	8
Reserved							ADCINT9
R-0							R-0/W-1

7	6	5	4	3	2	1	0
ADCINT8	ADCINT7	ADCINT6	ADCINT5	ADCINT4	ADCINT3	ADCINT2	ADCINT1
R-0/W-1	R-0/W-1	R-0/W-1	R-0/W-1	R-0/W-1	R-0/W-1	R-0/W-1	R-0/W-1

LEGEND: R/W = Read/Write; R = Read only; R-0/W-1 =always read 0, write 1 to set; *-n* = value after reset

Table 7. ADC Interrupt Overflow Clear Register (ADCINTOVFCLR) Field Descriptions

Bit	Field	Value	Description
15-9	Reserved	0	Reads return a zero; Writes have no effect.
8-0	ADCINT9 (x = 9 to 1)		ADC Interrupt Overflow Clear Bits. 0 No action. 1 Clears the respective overflow bit in the ADCINTOVF register. If software tries to set this bit on the same clock cycle that hardware tries to set the overflow bit in the ADCINTOVF register, then hardware has priority and the ADCINTOVF bit will be set.

Note: The following Interrupt Select Registers are EALLOW protected.

Figure 12. Interrupt Select 1 And 2 Register (INTSEL1N2) (Address Offset 08h)

15	14	13	12	11	10	9	8
Reserved	INT2CONT	INT2E	INT2SEL				
R-0	R/W-0	R/W-0	R/W-0				

7	6	5	4	3	2	1	0
Reserved	INT1CONT	INT1E	INT1SEL				
R-0	R/W-0	R/W-0	R/W-0				

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Figure 13. Interrupt Select 3 And 4 Register (INTSEL3N4) (Address Offset 09h)

15	14	13	12	8
Reserved	INT4CONT	INT4E	INT4SEL	
R-0	R/W-0	R/W-0	R/W-0	

7	6	5	4	0
Reserved	INT3CONT	INT3E	INT3SEL	
R-0	R/W-0	R/W-0	R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 14. Interrupt Select 5 And 6 Register (INTSEL5N6) (Address Offset 0Ah)

15	14	13	12	8
Reserved	INT6CONT	INT6E	INT6SEL	
R-0	R/W-0	R/W-0	R/W-0	

7	6	5	4	0
Reserved	INT5CONT	INT5E	INT5SEL	
R-0	R/W-0	R/W-0	R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 15. Interrupt Select 7 And 8 Register (INTSEL7N8) (Address Offset 0Bh)

15	14	13	12	8
Reserved	INT8CONT	INT8E	INT8SEL	
R-0	R/W-0	R/W-0	R/W-0	

7	6	5	4	0
Reserved	INT7CONT	INT7E	INT7SEL	
R-0	R/W-0	R/W-0	R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 16. Interrupt Select 9 And 10 Register (INTSEL9N10) (Address Offset 0Ch)

15	8
Reserved	
R-0	

7	6	5	4	0
Reserved	INT9CONT	INT9E	INT9SEL	
R-0	R/W-0	R/W-0	R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 8. INTSELxNy Register Field Descriptions

Bit	Field	Value	Description
15	Reserved	0	Reserved
14	INTyCONT		ADCINTy Continuous Mode Enable
		0	No further ADCINTy pulses are generated until ADCINTy flag (in ADCINTFLG register) is cleared by user.
		1	ADCINTy pulses are generated whenever an EOC pulse is generated irrespective if the flag bit is cleared or not.
13	INTyE		ADCINTy Interrupt Enable
		0	ADCINTy is disabled.
		1	ADCINTy is enabled.
12-8	INTySEL		ADCINTy EOC Source Select
		00h	EOC0 is trigger for ADCINTy
		01h	EOC1 is trigger for ADCINTy
		02h	EOC2 is trigger for ADCINTy
		03h	EOC3 is trigger for ADCINTy
		04h	EOC4 is trigger for ADCINTy
		05h	EOC5 is trigger for ADCINTy
		06h	EOC6 is trigger for ADCINTy
		07h	EOC7 is trigger for ADCINTy
		08h	EOC8 is trigger for ADCINTy
		09h	EOC9 is trigger for ADCINTy
		0Ah	EOC10 is trigger for ADCINTy
		0Bh	EOC11 is trigger for ADCINTy
		0Ch	EOC12 is trigger for ADCINTy
		0Dh	EOC13 is trigger for ADCINTy
		0Eh	EOC14 is trigger for ADCINTy
		0Fh	EOC15 is trigger for ADCINTy
		1xh	Invalid value.
7	Reserved	0	Reads return a zero; Writes have no effect.
6	INTxCONT		ADCINTx Continuous Mode Enable.
		0	No further ADCINTx pulses are generated until ADCINTx flag (in ADCINTFLG register) is cleared by user.
		1	ADCINTx pulses are generated whenever an EOC pulse is generated irrespective if the flag bit is cleared or not.
5	INTxE		ADCINTx Interrupt Enable
		0	ADCINTx is disabled.
		1	ADCINTx is enabled .

Table 8. INTSELxNy Register Field Descriptions (continued)

Bit	Field	Value	Description
4-0	INTxSEL		ADCINTx EOC Source Select
		00h	EOC0 is trigger for ADCINTx
		01h	EOC1 is trigger for ADCINTx
		02h	EOC2 is trigger for IADCNTx
		03h	EOC3 is trigger for ADCINTx
		04h	EOC4 is trigger for ADCINTx
		05h	EOC5 is trigger for ADCINTx
		06h	EOC6 is trigger for ADCINTx
		07h	EOC7 is trigger for ADCINTx
		08h	EOC8 is trigger for ADCINTx
		09h	EOC9 is trigger for ADCINTx
		0Ah	EOC10 is trigger for ADCINTx
		0Bh	EOC11 is trigger for ADCINTx
		0Ch	EOC12 is trigger for ADCINTx
		.0Dh	EOC13 is trigger for ADCINTx
		0Eh	EOC14 is trigger for ADCINTx
		0Fh	EOC15 is trigger for ADCINTx
		1xh	Invalid value.

1.10.3 ADC Priority Register

Note: The following SOC Priority Control Register is EALLOW protected.

Figure 17. ADC Start of Conversion Priority Control Register (SOCPRICTL)

15 11	10 5	4 0
Reserved	RRPOINTER	SOCPRIORITY
R-0	R-20h	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 9. SOCPRICTL Register Field Descriptions

Bit	Field	Value	Description
15-11	Reserved		Reads return a zero; Writes have no effect.
10-5	RRPOINTER		Round Robin Pointer. Holds the value of the last converted round robin SOCx to be used by the round robin scheme to determine order of conversions.
		00h	SOC0 was last round robin SOC to convert. SOC1 is highest round robin priority.
		01h	SOC1 was last round robin SOC to convert. SOC2 is highest round robin priority.
		02h	SOC2 was last round robin SOC to convert. SOC3 is highest round robin priority.
		03h	SOC3 was last round robin SOC to convert. SOC4 is highest round robin priority.
		04h	SOC4 was last round robin SOC to convert. SOC5 is highest round robin priority.
		05h	SOC5 was last round robin SOC to convert. SOC6 is highest round robin priority.
		06h	SOC6 was last round robin SOC to convert. SOC7 is highest round robin priority.
		07h	SOC7 was last round robin SOC to convert. SOC8 is highest round robin priority.
		08h	SOC8 was last round robin SOC to convert. SOC9 is highest round robin priority.
		09h	SOC9 was last round robin SOC to convert. SOC10 is highest round robin priority.
		0Ah	SOC10 was last round robin SOC to convert. SOC11 is highest round robin priority.
		0Bh	SOC11 was last round robin SOC to convert. SOC12 is highest round robin priority.
		0Ch	SOC12 was last round robin SOC to convert. SOC13 is highest round robin priority.
		0Dh	SOC13 was last round robin SOC to convert. SOC14 is highest round robin priority.
		0Eh	SOC14 was last round robin SOC to convert. SOC15 is highest round robin priority.
		0Fh	SOC15 was last round robin SOC to convert. SOC0 is highest round robin priority.
		1xh	Invalid value
		20h	Reset value to indicate no SOC has been converted. SOC0 is highest round robin priority. Set to this value when the device is reset, when the ADCCTL1.RESET bit is set, or when the SOCPRICTL register is written. In the latter case, if a conversion is currently in progress, it will complete and then the new priority will take effect.
		Others	Invalid selection.
4-0	SOCPRIORITY		SOC Priority.
			Determines the cutoff point for priority mode and round robin arbitration for SOCx
		00h	SOC priority is handled in round robin mode for all channels.
		01h	SOC0 is high priority, rest of channels are in round robin mode.
		02h	SOC0-SOC1 are high priority, SOC2-SOC15 are in round robin mode.
		03h	SOC0-SOC2 are high priority, SOC3-SOC15 are in round robin mode.
		04h	SOC0-SOC3 are high priority, SOC4-SOC15 are in round robin mode.
		05h	SOC0-SOC4 are high priority, SOC5-SOC15 are in round robin mode.
		06h	SOC0-SOC5 are high priority, SOC6-SOC15 are in round robin mode.
		07h	SOC0-SOC6 are high priority, SOC7-SOC15 are in round robin mode.
		08h	SOC0-SOC7 are high priority, SOC8-SOC15 are in round robin mode.
		09h	SOC0-SOC8 are high priority, SOC9-SOC15 are in round robin mode.
		0Ah	SOC0-SOC9 are high priority, SOC10-SOC15 are in round robin mode.
		0Bh	SOC0-SOC10 are high priority, SOC11-SOC15 are in round robin mode.
		0Ch	SOC0-SOC11 are high priority, SOC12-SOC15 are in round robin mode.
		0Dh	SOC0-SOC12 are high priority, SOC13-SOC15 are in round robin mode.
		0Eh	SOC0-SOC13 are high priority, SOC14-SOC15 are in round robin mode.
		0Fh	SOC0-SOC14 are high priority, SOC15 is in round robin mode.
		10h	All SOCs are in high priority mode, arbitrated by SOC number
		Others	Invalid selection.

1.10.4 ADC SOC Registers

Note: The following ADC Sample Mode Register is EALLOW protected.

Figure 18. ADC Sample Mode Register (ADCSAMPLEMODE) (Address Offset 12h)

15							8
Reserved							
R-0							

7	6	5	4	3	2	1	0
SIMULEN14	SIMULEN12	SIMULEN10	SIMULEN8	SIMULEN6	SIMULEN4	SIMULEN2	SIMULEN0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 10. ADC Sample Mode Register (ADCSAMPLEMODE) Field Descriptions

Bit	Field	Value	Description
15:8	Reserved	0	Reserved
7	SIMULEN14		Simultaneous sampling enable for SOC14/SOC15. Couples SOC14 and SOC15 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC14 or SOC15.
		0	Single sample mode set for SOC14 and SOC15. All bits of CHSEL field define channel to be converted. EOC14 associated with SOC14. EOC15 associated with SOC15. SOC14's result placed in ADCRESULT14 register. SOC15's result placed in ADCRESULT15.
		1	Simultaneous sample for SOC14 and SOC15. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC14 and EOC15 associated with SOC14 and SOC15 pair. SOC14's and SOC15's results will be placed in ADCRESULT14 and ADCRESULT15 registers, respectively.
6	SIMULEN12		Simultaneous sampling enable for SOC12/SOC13. Couples SOC12 and SOC13 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC12 or SOC13.
		0	Single sample mode set for SOC12 and SOC13. All bits of CHSEL field define channel to be converted. EOC12 associated with SOC12. EOC13 associated with SOC13. SOC12's result placed in ADCRESULT12 register. SOC13's result placed in ADCRESULT13.
		1	Simultaneous sample for SOC12 and SOC13. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC12 and EOC13 associated with SOC12 and SOC13 pair. SOC12's and SOC13's results will be placed in ADCRESULT12 and ADCRESULT13 registers, respectively.
5	SIMULEN10		Simultaneous sampling enable for SOC10/SOC11. Couples SOC10 and SOC11 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC10 or SOC11.
		0	Single sample mode set for SOC10 and SOC11. All bits of CHSEL field define channel to be converted. EOC10 associated with SOC10. EOC11 associated with SOC11. SOC10's result placed in ADCRESULT10 register. SOC11's result placed in ADCRESULT11.
		1	Simultaneous sample for SOC10 and SOC11. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC10 and EOC11 associated with SOC10 and SOC11 pair. SOC10's and SOC11's results will be placed in ADCRESULT10 and ADCRESULT11 registers, respectively.
4	SIMULEN8		Simultaneous sampling enable for SOC8/SOC9. Couples SOC8 and SOC9 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC8 or SOC9.
		0	Single sample mode set for SOC8 and SOC9. All bits of CHSEL field define channel to be converted. EOC8 associated with SOC8. EOC9 associated with SOC9. SOC8's result placed in ADCRESULT8 register. SOC9's result placed in ADCRESULT9.
		1	Simultaneous sample for SOC8 and SOC9. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC8 and EOC9 associated with SOC8 and SOC9 pair. SOC8's and SOC9's results will be placed in ADCRESULT8 and ADCRESULT9 registers, respectively.

Table 10. ADC Sample Mode Register (ADCSAMPLEMODE) Field Descriptions (continued)

Bit	Field	Value	Description
3	SIMULEN6		Simultaneous sampling enable for SOC6/SOC7. Couples SOC6 and SOC7 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC6 or SOC7.
		0	Single sample mode set for SOC6 and SOC7. All bits of CHSEL field define channel to be converted. EOC6 associated with SOC6. EOC7 associated with SOC7. SOC6's result placed in ADCRESULT6 register. SOC7's result placed in ADCRESULT7.
		1	Simultaneous sample for SOC6 and SOC7. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC6 and EOC7 associated with SOC6 and SOC7 pair. SOC6's and SOC7's results will be placed in ADCRESULT6 and ADCRESULT7 registers, respectively.
2	SIMULEN4		Simultaneous sampling enable for SOC4/SOC5. Couples SOC4 and SOC5 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC4 or SOC5.
		0	Single sample mode set for SOC4 and SOC5. All bits of CHSEL field define channel to be converted. EOC4 associated with SOC4. EOC5 associated with SOC5. SOC4's result placed in ADCRESULT4 register. SOC5's result placed in ADCRESULT5.
		1	Simultaneous sample for SOC4 and SOC5. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC4 and EOC5 associated with SOC4 and SOC5 pair. SOC4's and SOC5's results will be placed in ADCRESULT4 and ADCRESULT5 registers, respectively.
1	SIMULEN2		Simultaneous sampling enable for SOC2/SOC3. Couples SOC2 and SOC3 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC2 or SOC3.
		0	Single sample mode set for SOC2 and SOC3. All bits of CHSEL field define channel to be converted. EOC2 associated with SOC2. EOC3 associated with SOC3. SOC2's result placed in ADCRESULT2 register. SOC3's result placed in ADCRESULT3.
		1	Simultaneous sample for SOC2 and SOC3. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC2 and EOC3 associated with SOC2 and SOC3 pair. SOC2's and SOC3's results will be placed in ADCRESULT2 and ADCRESULT3 registers, respectively.
0	SIMULEN0		Simultaneous sampling enable for SOC0/SOC1. Couples SOC0 and SOC1 in simultaneous sampling mode. See section 1.5 for details. This bit should not be set when the ADC is actively converting SOC0 or SOC1.
		0	Single sample mode set for SOC0 and SOC1. All bits of CHSEL field define channel to be converted. EOC0 associated with SOC0. EOC1 associated with SOC1. SOC0's result placed in ADCRESULT0 register. SOC1's result placed in ADCRESULT1.
		1	Simultaneous sample for SOC0 and SOC1. Lowest three bits of CHSEL field define the pair of channels to be converted. EOC0 and EOC1 associated with SOC0 and SOC1 pair. SOC0's and SOC1's results will be placed in ADCRESULT0 and ADCRESULT1 registers, respectively.

Note: The following ADC Interrupt SOC Select Registers are EALLOW protected.

Figure 19. ADC Interrupt Trigger SOC Select 1 Register (ADCINTSOCSEL1) (Address Offset 14h)

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
SOC7	SOC6	SOC5	SOC4	SOC3	SOC2	SOC1	SOC0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 11. ADC Interrupt Trigger SOC Select 1 Register (ADCINTSOCSEL1) Register Field Descriptions

Bit	Field	Value	Description
15--0	SOCx (x = 7 to 0)		SOCx ADC Interrupt Trigger Select. Selects which, if any, ADCINT triggers SOCx. This field overrides the TRIGSEL field in the ADCSOCxCTL register.
		00	No ADCINT will trigger SOCx. TRIGSEL field determines SOCx trigger.
		01	ADCINT1 will trigger SOCx. TRIGSEL field is ignored.
		10	ADCINT2 will trigger SOCx. TRIGSEL field is ignored.
		11	Invalid selection.

Figure 20. ADC Interrupt Trigger SOC Select 2 Register (ADCINTSOCSEL2) (Address Offset 15h)

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
SOC15	SOC14	SOC13	SOC12	SOC11	SOC10	SOC9	SOC8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 12. ADC Interrupt Trigger SOC Select 2 Register (ADCINTSOCSEL2) Field Descriptions

Bit	Field	Value	Description
15-0	SOCx (x = 15 to 8)		SOCx ADC Interrupt Trigger Select. Selects which, if any, ADCINT triggers SOCx. This field overrides the TRIGSEL field in the ADCSOCxCTL register.
		00	No ADCINT will trigger SOCx. TRIGSEL field determines SOCx trigger.
		01	ADCINT1 will trigger SOCx. TRIGSEL field is ignored.
		10	ADCINT2 will trigger SOCx. TRIGSEL field is ignored.
		11	Invalid selection.

Figure 21. ADC SOC Flag 1 Register (ADCSOCFLG1) (Address Offset 18h)

15	14	13	12	11	10	9	8
SOC15	SOC14	SOC13	SOC12	SOC11	SOC10	SOC9	SOC8
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
7	6	5	4	3	2	1	0
SOC7	SOC6	SOC5	SOC4	SOC3	SOC2	SOC1	SOC0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 13. ADC SOC Flag 1 Register (ADCSOCFLG1) Field Descriptions

Bit	Field	Value	Description
15-0	SOCx (x = 15 to 0)		SOCx Start of Conversion Flag. Indicates the state of individual SOC conversions.
		0	No sample pending for SOCx.
		1	Trigger has been received and sample is pending for SOCx.
			The bit will be automatically cleared when the respective SOCx conversion is started. If contention exists where this bit receives both a request to set **and** a request to clear on the same cycle, regardless of the source of either, this bit will be set and the request to clear will be ignored. In this case the overflow bit in the ADCSOCOVF1 register will not be affected regardless of whether this bit was previously set or not.

Figure 22. ADC SOC Force 1 Register (ADCSOCFRC1) (Address Offset 1Ah)

15	14	13	12	11	10	9	8
SOC15	SOC14	SOC13	SOC12	SOC11	SOC10	SOC9	SOC8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
SOC7	SOC6	SOC5	SOC4	SOC3	SOC2	SOC1	SOC0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 14. ADC SOC Force 1 Register (ADCSOCFRC1) Field Descriptions

Bit	Field	Value	Description
15-0	SOCx (x = 15 to 0)		SOCx Force Start of Conversion Flag. Writing a 1 will force to 1 the respective SOCx flag bit in the ADCSOCFLG1 register. This can be used to initiate a software initiated conversion. Writes of 0 are ignored.
		0	No action.
		1	Force SOCx flag bit to 1. This will cause a conversion to start once priority is given to SOCx.
			If software tries to set this bit on the same clock cycle that hardware tries to clear the SOCx bit in the ADCSOCFLG1 register, then software has priority and the ADCSOCFLG1 bit will be set. In this case the overflow bit in the ADCSOCOVF1 register will not be affected regardless of whether the ADCSOCFLG1 bit was previously set or not.

Figure 23. ADC SOC Overflow 1 Register (ADCSOCOVF1) (Address Offset 1Ch)

15	14	13	12	11	10	9	8
SOC15	SOC14	SOC13	SOC12	SOC11	SOC10	SOC9	SOC8
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

7	6	5	4	3	2	1	0
SOC7	SOC6	SOC5	SOC4	SOC3	SOC2	SOC1	SOC0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 15. ADC SOC Overflow 1 Register (ADCSOCOVF1) Field Descriptions

Bit	Field	Value	Description
15-0	SOCx (x = 15 to 0)		SOCx Start of Conversion Overflow Flag. Indicates an SOCx event was generated while an existing SOCx event was already pending.
		0	No SOCx event overflow
		1	SOCx event overflow
			An overflow condition does not stop SOCx events from being processed. It simply is an indication that a trigger was missed

Figure 24. ADC SOC Overflow Clear 1 Register (ADCSOCOVFCLR1) (Address Offset 1Eh)

15	14	13	12	11	10	9	8
SOC15	SOC14	SOC13	SOC12	SOC11	SOC10	SOC9	SOC8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

7	6	5	4	3	2	1	0
SOC7	SOC6	SOC5	SOC4	SOC3	SOC2	SOC1	SOC0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 16. ADC SOC Overflow Clear 1 Register (ADCSOCOVFCLR1) Field Descriptions

Bit	Field	Value	Description
15-0	SOCx (x = 15 to 0)		SOCx Clear Start of Conversion Overflow Flag. Writing a 1 will clear the respective SOCx overflow flag in the ADCSOCOVF1 register. Writes of 0 are ignored.
		0	No action.
		1	Clear SOCx overflow flag.
			If software tries to set this bit on the same clock cycle that hardware tries to set the overflow bit in the ADCSOCOVF1 register, then hardware has priority and the ADCSOCOVF1 bit will be set.

Note: The following ADC SOC0 - SOC15 Control Registers are EALLOW protected.

Figure 25. ADC SOC0 - SOC15 Control Registers (ADCSOCxCTL) (Address Offset 20h - 2Fh)

15 11	10	9 6	5 0
TRIGSEL	Reserved	CHSEL	ACQPS
R/W-0	R-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 17. ADC SOC0 - SOC15 Control Registers (ADCSOCxCTL) Register Field Descriptions

Bit	Field	Value	Description
15-11	TRIGSEL		SOCx Trigger Source Select. Configures which trigger will set the respective SOCx flag in the ADCSOCFLG1 register to intiate a conversion to start once priority is given to SOCx. This setting can be overridden by the respective SOCx field in the ADCINTSOCSEL1 or ADCINTSOCSEL2 register.
		00h	ADCTRIG0 - Software only.
		01h	ADCTRIG1 - CPU Timer 0, TINT0n
		02h	ADCTRIG2 - CPU Timer 1, TINT1n
		03h	ADCTRIG3 - CPU Timer 2, TINT2n
		04h	ADCTRIG4 – XINT2, XINT2SOC
		05h	ADCTRIG5 – ePWM1, ADCSOCA
		06h	ADCTRIG6 – ePWM1, ADCSOCB
		07h	ADCTRIG7 – ePWM2, ADCSOCA
		08h	ADCTRIG8 – ePWM2, ADCSOCB
		09h	ADCTRIG9 – ePWM3, ADCSOCA
		0Ah	ADCTRIG10 – ePWM3, ADCSOCB
		0Bh	ADCTRIG11 – ePWM4, ADCSOCA
		0Ch	ADCTRIG12 – ePWM4, ADCSOCB
		0Dh	ADCTRIG13 – ePWM5, ADCSOCA
		0Eh	ADCTRIG14 – ePWM5, ADCSOCB
		0Fh	ADCTRIG15 – ePWM6, ADCSOCA
		10h	ADCTRIG16 – ePWM6, ADCSOCB
		11h	ADCTRIG17 - ePWM7, ADCSOCA
		12h	ADCTRIG18 - ePWM7, ADCSOCB
		Others	Invalid selection.
10	Reserved		Reads return a zero; Writes have no effect.

Table 17. ADC SOC0 - SOC15 Control Registers (ADCSOCxCTL) Register Field Descriptions (continued)

Bit	Field	Value	Description
9-6	CHSEL		SOCx Channel Select. Selects the channel to be converted when SOCx is received by the ADC.
			Sequential Sampling Mode (SIMULENx = 0):
		0h	ADCINA0
		1h	ADCINA1
		2h	ADCINA2
		3h	ADCINA3
		4h	ADCINA4
		5h	ADCINA5
		6h	ADCINA6
		7h	ADCINA7
		8h	ADCINB0
		9h	ADCINB1
		Ah	ADCINB2
		Bh	ADCINB3
		Ch	ADCINB4
		Dh	ADCINB5
		Eh	ADCINB6
		Fh	ADCINB7
			Simultaneous Sampling Mode (SIMULENx = 1):
		0h	ADCINA0/ADCINB0 pair
		1h	ADCINA1/ADCINB1 pair
		2h	ADCINA2/ADCINB2 pair
		3h	ADCINA3/ADCINB3 pair
		4h	ADCINA4/ADCINB4 pair
		5h	ADCINA5/ADCINB5 pair
		6h	ADCINA6/ADCINB6 pair
		7h	ADCINA7/ADCINB7 pair
		8h	Invalid selection.
		9h	Invalid selection.
		Ah	Invalid selection.
		Bh	Invalid selection.
		Ch	Invalid selection.
		Dh	Invalid selection.
		Eh	Invalid selection.
		Fh	Invalid selection.

Table 17. ADC SOC0 - SOC15 Control Registers (ADCSOCxCTL) Register Field Descriptions (continued)

Bit	Field	Value	Description
5-0	ACQPS		SOCx Acquisition Prescale. Controls the sample and hold window for SOCx. Minimum value allowed is 6.
		00h	Invalid selection.
		01h	Invalid selection.
		02h	Invalid selection.
		03h	Invalid selection.
		04h	Invalid selection.
		05h	Invalid selection.
		06h	Sample window is 7 cycles long (6 + 1 clock cycles).
		07h	Sample window is 8 cycles long (7 + 1 clock cycles).
		08h	Sample window is 9 cycles long (8 + 1 clock cycles).
		09h	Sample window is 10 cycles long (9 + 1 clock cycles).
		...	...
		3Fh	Sample window is 64 cycles long (63 + 1 clock cycles).

1.10.5 ADC Calibration Registers

Note: The following ADC Calibration Registers are EALLOW protected.

Figure 26. ADC Reference/Gain Trim Register (ADCREFTRIM) (Address Offset 40h)

15 13	12 8	7 4	3 0
Reserved	EXTREF_FINE_TRIM	BG_COARSE_TRIM	BG_FINE_TRIM
R-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 18. ADC Reference/Gain Trim Register (ADCREFTRIM) Field Descriptions

Bit	Field	Value	Description
15-13	Reserved		Reads return a zero; Writes have no effect.
12-8	EXTREF_FINE_TRIM		ADC External reference Fine Trim. These bits should not be modified after device boot code loads them with the factory trim setting.
7-4	BG_COARSE_TRIM		ADC Internal Bandgap Fine Trim. These bits should not be modified after device boot code loads them with the factory trim setting.
3-0	BG_FINE_TRIM		ADC Internal Bandgap Coarse Trim. A maximum value of 30 is supported. These bits should not be modified after device boot code loads them with the factory trim setting.

Figure 27. ADC Offset Trim Register (ADCOFFTRIM) (Address Offset 41h)

15 9	8 0
Reserved	OFFTRIM
R-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 19. ADC Offset Trim Register (ADCOFFTRIM) Field Descriptions

Bit	Field	Value	Description
15-9	Reserved		Reads return a zero; Writes have no effect.
8-0	OFFTRIM		ADC Offset Trim. 2's complement of ADC offset. Range is -256 to +255. These bits are loaded by device boot code with a factory trim setting. Modification of this default setting can be made to correct any board induced offset.

1.10.6 ADC Revision Register

Figure 28. ADC Revision Register (ADCREV) (Address Offset 4Fh)

15	8
REV	
R-x	

7	0
TYPE	
R-3h	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 20. ADC Revision Register (ADCREV) Field Descriptions

Bit	Field	Value	Description
15-8	REV		ADC Revision. To allow documentation of differences between revisions. First version is labeled as 00h.
7-0	TYPE	3	ADC Type. Always set to 3 for this type ADC

1.10.7 ADC Result Registers

The ADC Result Registers are found in Peripheral Frame 0 (PF0). In the header files, the ADCRESULTx registers are located in the AdcResult register file, not AdcRegs.

Figure 29. ADC RESULT0 - RESULT15 Registers (ADCRESULTx) (PF1 Block Address Offset 00h - 0Fh)

15 12	11 0
Reserved	RESULT
R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 21. ADC RESULT0 - ADCRESULT15 Registers (ADCRESULTx) Field Descriptions

Bit	Field	Value	Description
15-12	Reserved		Reads return a zero; Writes have no effect.
11-0	RESULT		12-bit right-justified ADC result Sequential Sampling Mode (SIMULENx = 0): After the ADC completes a conversion of an SOCx, the digital result is placed in the corresponding ADCRESULTx register. For example, if SOC4 is configured to sample ADCINA1, the completed result of that conversion will be placed in ADCRESULT4. Simultaneous Sampling Mode (SIMULENx = 1): After the ADC completes a conversion of a channel pair, the digital results are found in the corresponding ADCRESULTx and ADCRESULTx+1 registers (assuming x is even). For example, for SOC4, the completed results of those conversions will be placed in ADCRESULT4 and ADCRESULT5. See 1.11 for timings of when this register is written.

1.11 ADC Timings

Figure 30. Timing Example For Sequential Mode / Late Interrupt Pulse

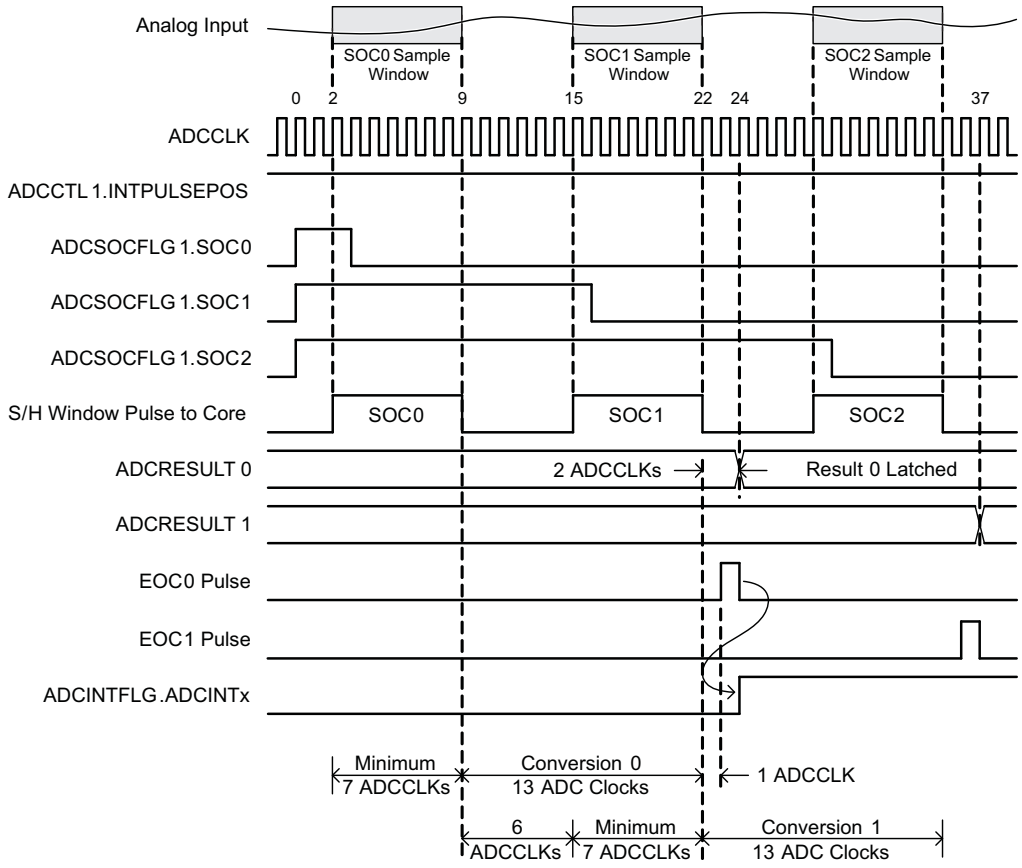


Figure 31. Timing Example For Sequential Mode / Early Interrupt Pulse

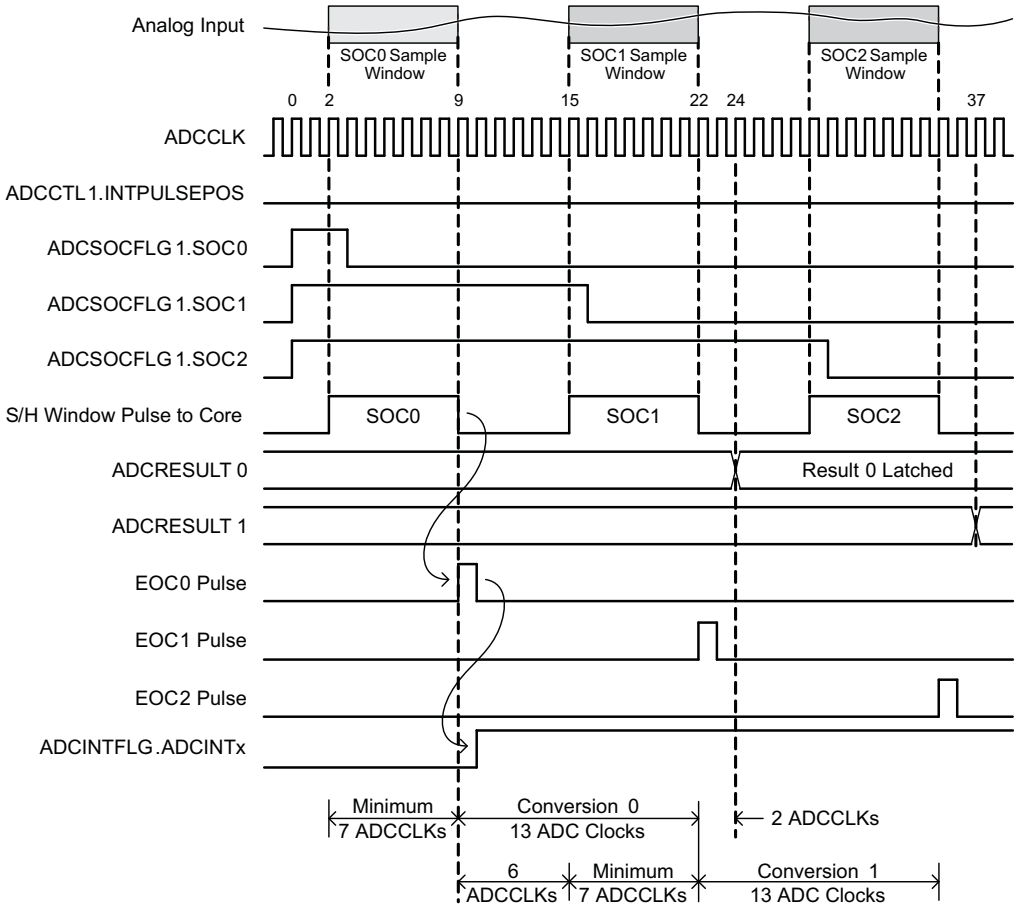


Figure 32. Timing Example For Simultaneous Mode / Late Interrupt Pulse

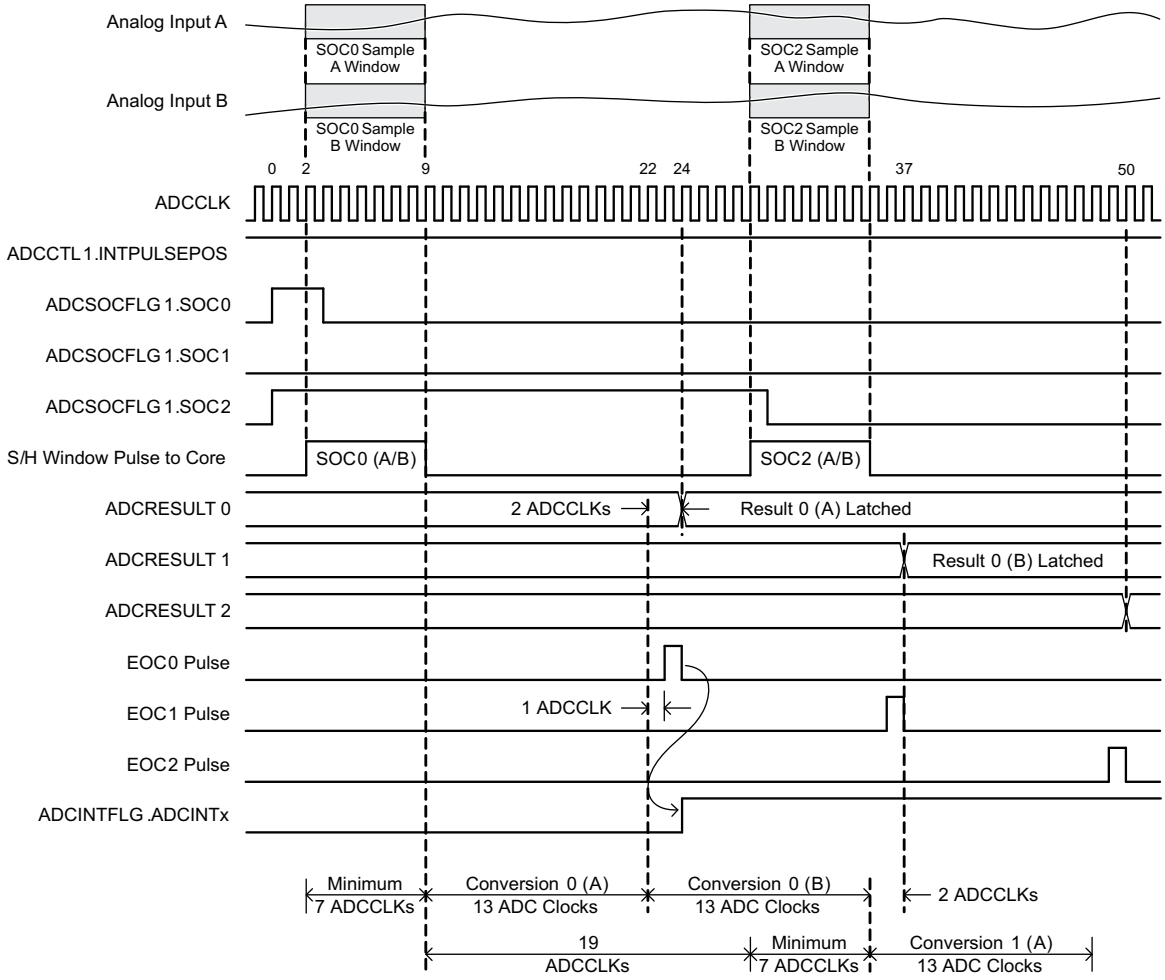
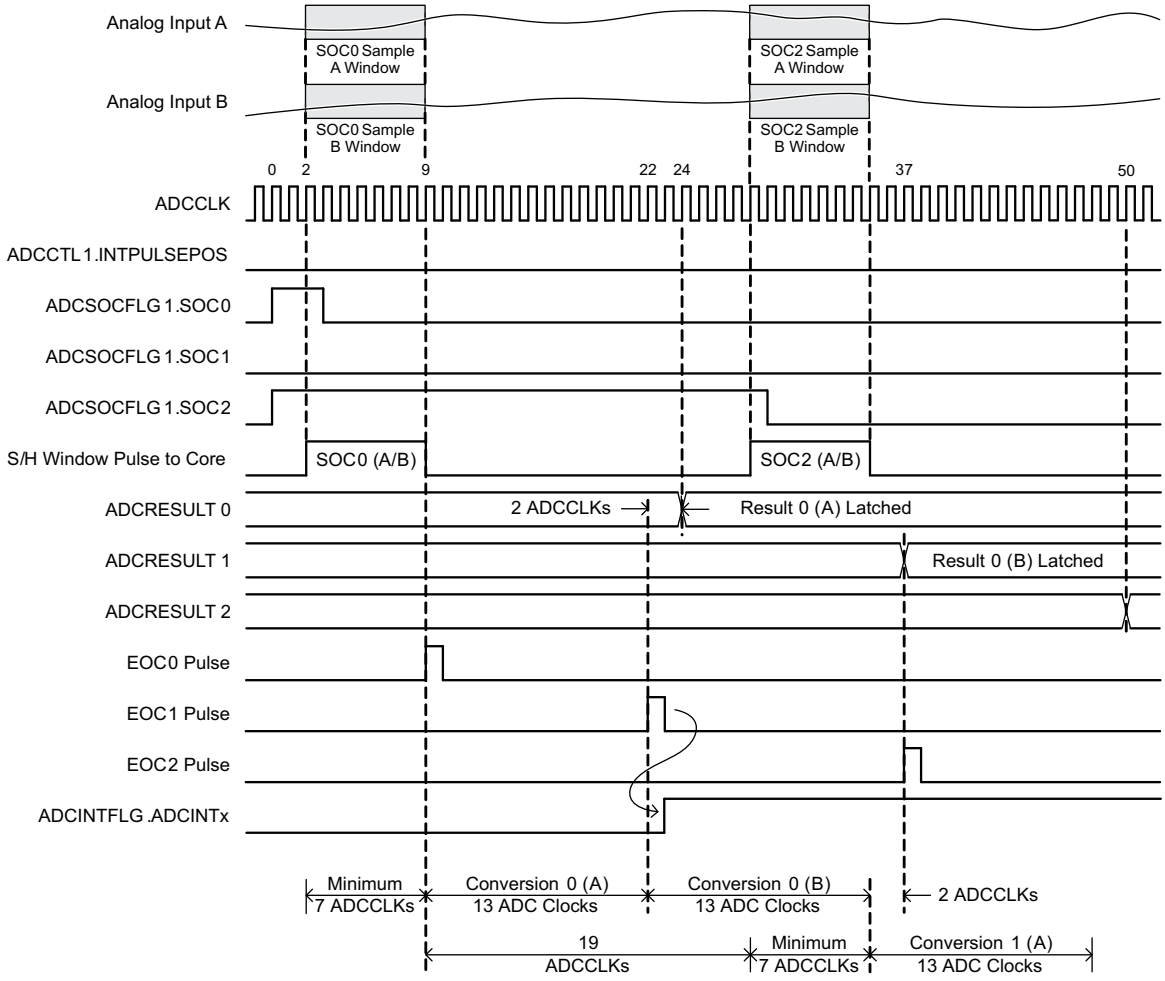


Figure 33. Timing Example For Simultaneous Mode / Early Interrupt Pulse



2 Comparator Block

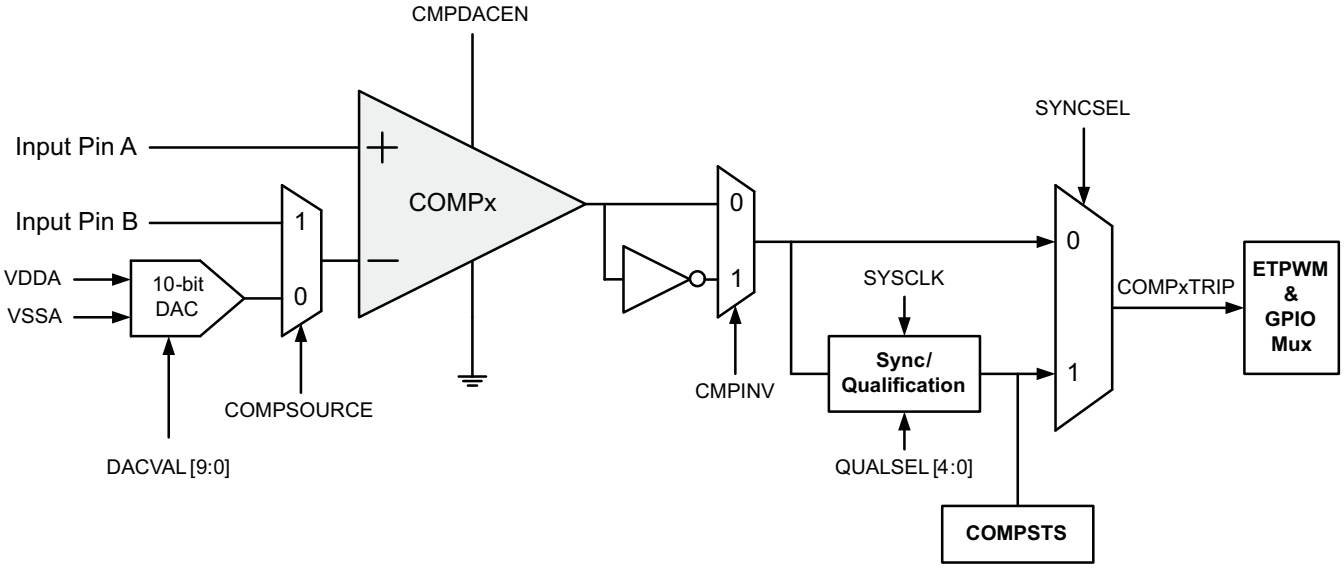
The comparator module described in this reference guide is a true analog voltage comparator in the VDDA domain. The analog portion of the block include the comparator, its inputs and outputs, and the internal DAC reference. The digital circuits, referred to as the wrapper in this document, include the DAC controls, interface to other on-chip logic, output qualification block, and the control signals.

2.1 Features

The comparator block can accommodate two external analog inputs or one external analog input using the internal DAC reference for the other input. The output of the comparator can be passed asynchronously or qualified and synchronized to the system clock period. The comparator output is routed to both the ePWM Trip Zone modules, as well as the GPIO output multiplexer.

2.2 2.2 Block Diagram

Figure 34. Comparator Block Diagram



2.3 Comparator Function

The comparator in each comparator block is an analog comparator module, and as such its output is asynchronous to the system clock. The truth table for the comparator is shown in Table 22.

Figure 35. Comparator

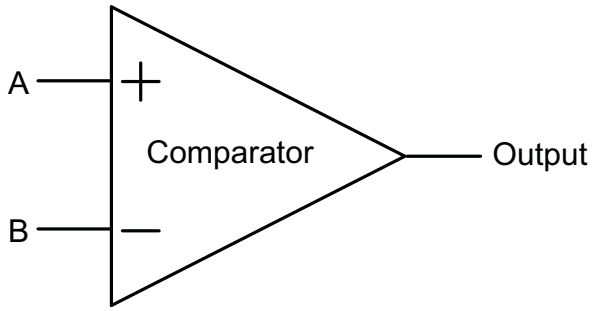


Table 22. Comparator Truth Table

Voltages	Output
Voltage A > Voltage B	1
Voltage B > Voltage A	0

There is no definition for the condition Voltage A = Voltage B since there is hysteresis in the response of the comparator output. Refer to the device datasheet for the value of this hysteresis. This also limits the sensitivity of the comparator output to noise on the input voltages.

The output state of the comparator, after qualification, is reflected by the COMPSTS bit in the COMPSTS register. Since this bit is part of the wrapper, clocks must be enabled to the comparator block for the COMPSTS bit to actively show the comparator state.

2.4 DAC Reference

Each comparator block contains a 10-bit voltage DAC reference that can used to supply the inverting input (B side input) of the comparator. The voltage output of the DAC is controlled by the DACVAL bit field in the DACVAL register. The output of the DAC is given by the equation:

$$V = \frac{DACVAL * (VDDA-VSSA)}{1023}$$

Since the DAC is also in the analog domain it does not require a clock to maintain its voltage output. A clock is required, however, to modify the digital inputs that control the DAC.

2.5 Initialization

There are 2 steps that must be performed prior to using the comparator block:

1. Enable the Band Gap inside the ADC by writing a 1 to the ADCBGPWD bit inside ADCTRL1.
2. Enable the comparator block by writing a 1 to the COMPDACEN bit in the COMPCTL register.

2.6 Digital Domain Manipulation

At the output of the comparator there are two more functional blocks that can be used to influence the behavior of the comparator output. They are:

1. Inverter circuit: Controlled by the CMPINV bit in the COMPCTL register; will apply a logical NOT to the output of the comparator. This function is asynchronous, while its control requires a clock present in order to change its value.
2. Qualification block: Controlled by the QUALSEL bit field in the COMPCTL register, and gated by the SYNCSEL bit in the COMPCTL register. This block can be used as a simple filter to only pass the output of the comparator once it is synchronized to the system clock. and qualified by the number of system clocks defined in QUALSEL bit field.

2.7 Comparator Registers

F280x2x devices have two comparators COMP1 and COMP2. Table 23 lists the registers for these modules.

Name	Address Range	Size(x16)	Description
COMP1	6400h – 641Fh	1	Comparator
COMP2	6420h – 642Fh	1	Comparator

Table 23. Comparator Module Registers

Name	Address Range(base)	Size(x16)	Description
COMPCTL	0x0000 0000	1	comparator control
Reserved	0x0000 0001	1	Reserved
COMPSTS	0x0000 0002	1	compare output status
Reserved	0x0000 0003	1	Reserved
Reserved	0x0000 0004	1	Reserved
Reserved	0x0000 0005	1	Reserved
DACVAL	0x0000 0006	1	10-bit DAC Value
Reserved	0x0000 0007 0x0000 001F	25	Reserved

2.7.1 Comparator Control (COMPCTL) Register

Figure 36. Comparator Control (COMPCTL) Register

15	9	8
Reserved		SYNCSEL
R-0		R/W-0

7	3	2	1	0
QUALSEL		CMPINV	COMPSOURCE	COMPDACE
R/W-0		R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 24. COMPCTL Register Field Descriptions

Bit	Field	Value	Description
15-9	Reserved		Reads return a 0; Writes have no effect.
8	SYNCSEL		Synchronization select for output of the comparator before being passed to ETPWM/GPIO blocks
		0	Asynchronous version of Comparator output is passed
		1	Synchronous version of comparator output is passed
7-3	QUALSEL		Qualification Period for synchronized output of the comparator
		0h	Synchronized value of comparator is passed through
		1h	Input to the block must be consistent for 2 consecutive clocks before output of Qual block can change
		2h	Input to the block must be consistent for 3 consecutive clocks before output of Qual block can change
		...	...
		Fh	Input to the block must be consistent for 15 consecutive clocks before output of Qual block can change

Table 24. COMPCTL Register Field Descriptions (continued)

Bit	Field	Value	Description
2	CMPINV		Invert select for Comparator
		0	Output of comparator is passed
		1	Inverted output of comparator is passed
1	COMPSOURCE		Source select for comparator inverting input
		0	Inverting input of comparator connected to internal DAC
		1	Inverting input connected to external pin
0	COMPDACE		Comparator/DAC Enable
		0	Comparator/DAC logic is powered down.
		1	Comparator/DAC logic is powered up.

2.7.2 Compare Output Status (COMPSTS) Register

Figure 37. Compare Output Status (COMPSTS) Register

15	1	0
Reserved		COMPSTS
R-0		R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 25. Compare Output Status (COMPSTS) Register Field Descriptions

Bit	Field	Value	Description
15-1	Reserved		Reads return zero and writes have no effect.
0	COMPSTS		Logical latched value of the comparator

2.7.3 DAC Value (DACVAL) Register

Figure 38. DAC Value (DACVAL) Register

15	10	9	0
Reserved		DACVAL	
R-0		R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 26. DAC Value (DACVAL) Register Field Descriptions

Bit	Field	Value	Description
15-10	Reserved		Reads return zero and writes have no effect.
9-0	DACVAL	0-3FFh	DAC Value bits, scales the output of the DAC from 0 – 1023.

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 TI 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. Information of third parties may be subject to additional restrictions.

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.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

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
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
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
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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