Section 44. High-Speed 10-Bit ADC

HIGHLIGHTS

This section of the manual contains the following major topics:

44.1 INTRODUCTION

This section describes the features and associated operational modes of the high-speed 10-bit Analog-to-Digital Converter (ADC) available on the dsPIC33F family of devices.

The High-Speed 10-Bit ADC module has the following key features:

- 10-bit resolution
- 4 Msps conversion rate at 3.3V (devices with two Successive Approximation Registers (SARs))
- 2 Msps conversion rate at 3.3V (devices with one SAR)
- Independent Start of Conversion (SoC) trigger selection for each analog input pair
- Up to four dedicated Sample and Hold (S&H) circuits with asynchronous sampling option
- Two shared S&H circuits on devices with two SARs
- One shared S&H circuit on devices with one SAR
- Dedicated result register for each analog input
- Unipolar inputs

Power conversion applications often require voltage and current measurements for each control loop. Therefore, the 26 analog inputs of the High-Speed 10-Bit ADC module are grouped in 13 pairs. A pair is a combination of even and odd numbered analog inputs such as AN0 and AN1, AN2, AN3 and so on. The ADC always converts a single pair of analog inputs at a time. Whether the conversion happens in parallel or sequential manner depends on the number of SAR converters available on the device.

Note:	The available analog inputs and SAR converters may vary depending on the device variant. Refer to the specific device data sheet for details.

Each analog input pair (for example, Pair 0 (AN0, AN1), Pair 1 (AN2, AN3)) receives a separate conversion request. The conversion request can be selected from a variety of sources (see Figure 44-7). If multiple analog input pairs receive a conversion request at the same time, the conversion requests are prioritized. Analog input Pair 0 has the highest priority, and analog input Pair 12 has the lowest priority.

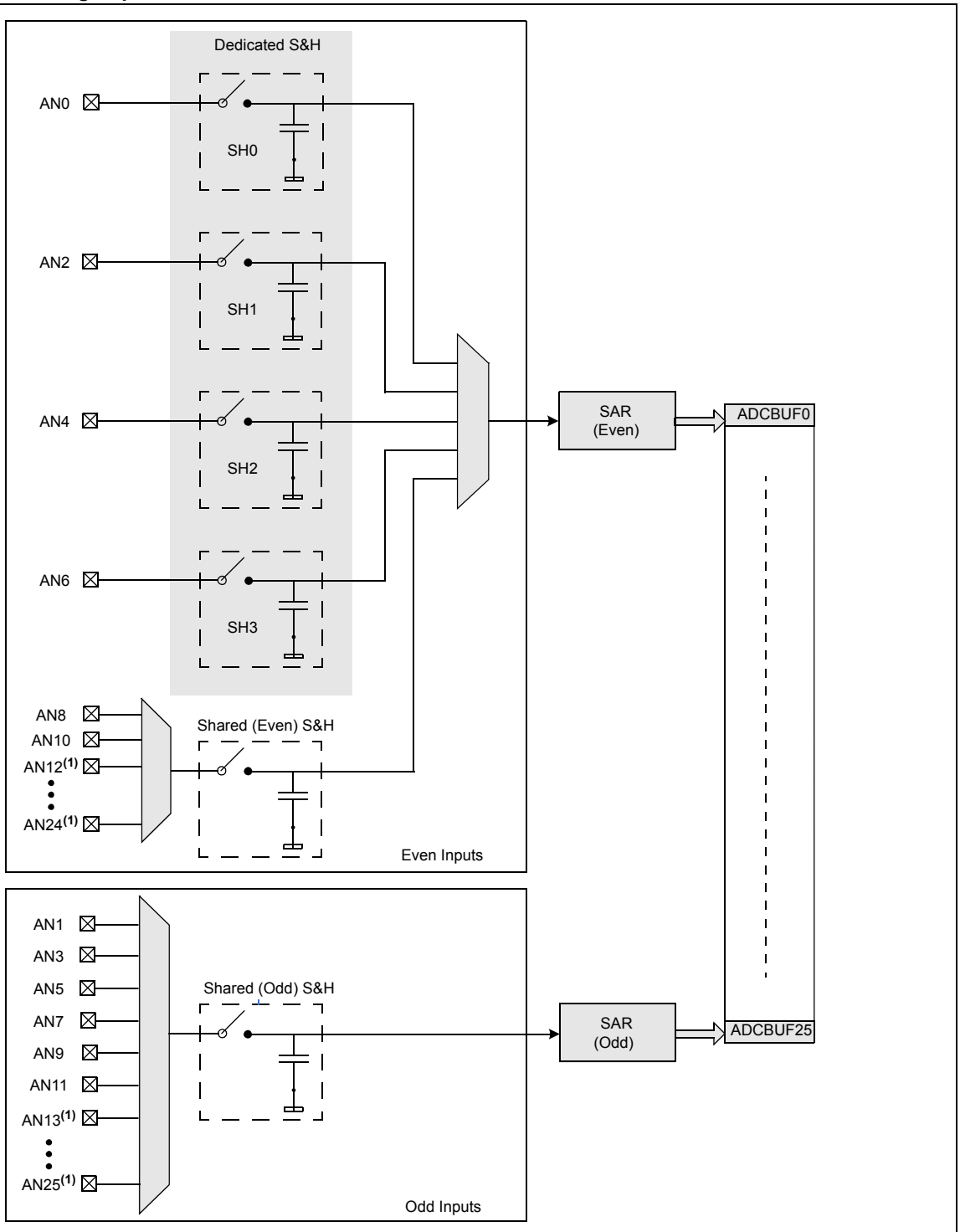
Figure 44-1 shows a block diagram of the High-Speed 10-Bit ADC with a dual SAR converter. In this module, the even and odd numbered analog inputs are converted in parallel, thereby providing 4 Msps throughput using two 2 Msps SAR converters. The even numbered analog inputs are converted by one SAR, and the odd numbered analog inputs are converted by another SAR. The dual SAR device has a separate shared S&H circuit for even and odd numbered analog inputs to keep the analog input constant for the respective SAR during conversion.

The separate shared S&H circuit for even and odd numbered analog inputs also provides the option to sample both the inputs (the even and odd input) in a pair simultaneously, thus preserving the relative phase information between the signals on both analog inputs.

Figure 44-2 shows a block diagram of the High-Speed 10-Bit ADC with a single SAR converter. In this module, the even and odd numbered analog inputs are converted sequentially. Unlike a dual SAR device, it has a single shared S&H circuit for even and odd numbered analog inputs. Therefore, the analog input pairs that use the shared S&H circuit for both inputs are sampled sequentially.

Each of the first four analog input pairs in both the single and dual SAR device has a dedicated S&H circuit for even numbered analog inputs (AN0, AN2, AN4 and AN6). The dedicated S&H circuit allows the respective analog input to be sampled on a conversion request without any latency (zero latency).

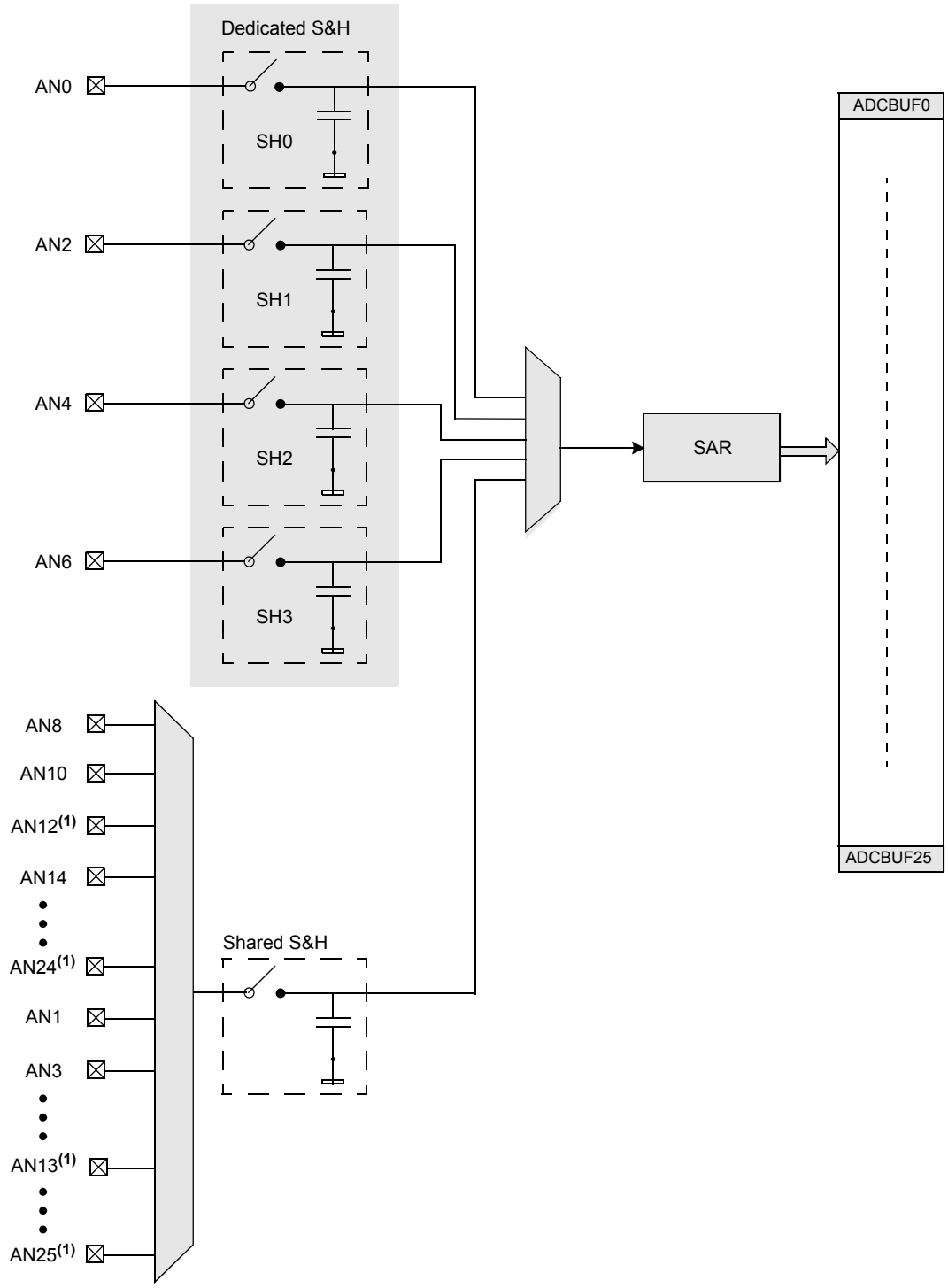
Figure 44-1: High-Speed 10-Bit ADC with Two SAR Converters



Note 1: Depending on the device variant, these inputs may be connected to EXTREF or Internal Voltage Reference. Refer to the specific device data sheet for more details.

2: The available analog inputs and the dedicated S&H circuit may vary depending on the device variant. Refer to the specific device data sheet for more details.

Figure 44-2: High-Speed 10-Bit ADC with One SAR Converter



Note 1: Depending on the device variant, these inputs may be connected to EXTREF or Internal Voltage Reference. Refer to the specific device data sheet for more details.

2: The available analog inputs and the dedicated S&H circuit may vary depending on the device variant. Refer to the specific device data sheet for more details.

44.2 CONTROL REGISTERS

This section outlines the specific functions of each register that controls the operation of the High-Speed 10-Bit ADC module.

Note:	Not all control registers are available on all devices. Refer to the specific device data sheet for more information.

- **ADCON: A/D Control Register**

  This register configures the sample conversion sequence, enables the ADC module and is used to set up the clock divider for the ADC clock.

- **ADSTAT: A/D Status Register**

  This register contains the Pair Data Ready (PxRDY) flag to indicate the analog input pair that caused the common ADC interrupt. The Pair Data Ready flag is cleared in the specific pair handler.

- **ADBASE: A/D Base Register**

  This register contains a unique offset value based on the analog input pair that caused the common ADC interrupt. It is read in the common ADC interrupt to branch to the specific analog pair handler.

- **ADPCFG: A/D Port Configuration Register**

  This register configures the analog input pins as analog inputs or digital I/O.

- **ADPCFG2: A/D Port Configuration Register 2**

  This register configures the analog input pins as analog inputs or digital I/O.

- **ADCPC0: A/D Convert Pair Control Register 0**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 0 and Pair 1.

- **ADCPC1: A/D Convert Pair Control Register 1**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 2 and Pair 3.

- **ADCPC2: A/D Convert Pair Control Register 2**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 4 and Pair 5.

- **ADCPC3: A/D Convert Pair Control Register 3**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 6 and Pair 7.

- **ADCPC4: A/D Convert Pair Control Register 4**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 8 and Pair 9.

- **ADCPC5: A/D Convert Pair Control Register 5**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 10 and Pair 11.

- **ADCPC6: A/D Convert Pair Control Register 6**

  This register selects the trigger source, enables the common ADC interrupt and allows software trigger generation for Analog Input Pair 12.

Register 44-1: ADCON: A/D Control Register

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
ADON	—	ADSIDL	SLOWCLK(1)	—	GSWTRG	—	FORM(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-1	R/W-1
EIE(1)	ORDER(1)	SEQSAMP(1)	ASYNCSAMP(1)	—	ADCS<2:0>(1)		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **ADON:** A/D Operating Mode bit
1 = A/D converter module is operating
0 = A/D converter is off

bit 14 **Unimplemented:** Read as '0'

bit 13 **ADSIDL:** Stop in Idle Mode bit
1 = Discontinue module operation when device enters Idle mode
0 = Continue module operation in Idle mode

bit 12 **SLOWCLK:** Enable Slow Clock Divider bit(1)
1 = ADC is clocked by the auxiliary PLL (ACLK)
0 = ADC is clock by the primary PLL (FVCO)

bit 11 **Unimplemented:** Read as '0'

bit 10 **GSWTRG:** Global Software Trigger bit
When this bit is set, it triggers conversions if selected by the TRGSRC<4:0> bits in the ADCPCx registers. This bit is automatically cleared in hardware.

bit 9 **Unimplemented:** Read as '0'

bit 8 **FORM:** Data Output Format bit(1)
1 = Fractional (DOUT = dddd dddd dd00 0000)
0 = Integer (DOUT = 0000 00dd dddd dddd)

bit 7 **EIE:** Early Interrupt Enable bit(1)
1 = Interrupt is generated after first conversion is completed
0 = Interrupt is generated after second conversion is completed

bit 6 **ORDER:** Conversion Order bit(1)
1 = Odd numbered analog input is converted first, followed by conversion of even numbered input
0 = Even numbered analog input is converted first, followed by conversion of odd numbered input

bit 5 **SEQSAMP:** Sequential Sample Enable bit(1)
1 = Shared Sample and Hold (S&H) circuit is sampled at the start of the second conversion if ORDER = 0. If ORDER = 1, then the shared S&H circuit is sampled at the start of the first conversion
0 = Shared S&H circuit is sampled at the same time the dedicated S&H circuit is sampled, if the shared S&H circuit is not currently busy with an existing conversion process. If the shared S&H circuit is busy at the time the dedicated S&H circuit is sampled, then the shared S&H circuit will sample at the start of the new conversion cycle

bit 4 **ASYNCSAMP:** Asynchronous Dedicated S&H Sampling Enable bit(1)
1 = The dedicated S&H circuit is constantly sampling and then terminates sampling as soon as the trigger pulse is detected
0 = The dedicated S&H circuit starts sampling when the trigger event is detected and completes the sampling process in two ADC clock cycles

Note 1: This control bit can only be changed while the ADC module is disabled (ADON = 0).

Register 44-1: ADCON: A/D Control Register (Continued)

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **ADCS<2:0>:** A/D Conversion Clock Divider Select bits(1)

111 = FADC/8
110 = FADC/7
101 = FADC/6
100 = FADC/5
011 = FADC/4 (default)
010 = FADC/3
001 = FADC/2
000 = FADC/1

Note 1: This control bit can only be changed while the ADC module is disabled (ADON = 0).

Register 44-2: ADSTAT: A/D Status Register

U-0	U-0	U-0	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S
—	—	—	P12RDY	P11RDY	P10RDY	P9RDY	P8RDY
bit 15							bit 8

R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S	R/C-0 H-S
P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY	P1RDY	P0RDY
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown
C = Cleared in software	H-S = Set by hardware		

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **P12RDY:** Conversion Data for Pair 12 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 11 **P11RDY:** Conversion Data for Pair 11 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 10 **P10RDY:** Conversion Data for Pair 10 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 9 **P9RDY:** Conversion Data for Pair 9 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 8 **P8RDY:** Conversion Data for Pair 8 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 7 **P7RDY:** Conversion Data for Pair 7 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 6 **P6RDY:** Conversion Data for Pair 6 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 5 **P5RDY:** Conversion Data for Pair 5 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 4 **P4RDY:** Conversion Data for Pair 4 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 3 **P3RDY:** Conversion Data for Pair 3 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 2 **P2RDY:** Conversion Data for Pair 2 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 1 **P1RDY:** Conversion Data for Pair 1 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

bit 0 **P0RDY:** Conversion Data for Pair 0 Ready bit
Bit is set when data is ready in buffer, cleared when a '0' is written to this bit.

Note: Not all PxRDY bits are available on all devices. See the specific device data sheet for the available analog inputs.

Register 44-3: ADBASE: A/D Base Register

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADBASE<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
ADBASE<7:1>							—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-1 **ADBASE<15:1>: ADC Base Register bits**

This register contains the base address of the user's ADC Interrupt Service Routine jump table. This register, when read, contains the sum of the ADBASE register contents and the encoded value of the PxRDY Status bits.

The encoder logic provides the bit number of the highest priority PxRDY bits where P0RDY is the highest priority and P12RDY is lowest priority.

bit 0 **Unimplemented:** Read as '0'

Note 1: The encoding results are shifted left two bits so bits 1-0 of the result are always zero.

2: As an alternative to using the ADBASE Register, the ADCP0-12 ADC Pair Conversion Complete Interrupts can be used to invoke A to D conversion completion routines for individual ADC input pairs.

Register 44-4: ADPCFG: A/D Port Configuration Register

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **PCFG<15:0>:** A/D Port Configuration Control bits

1 = Port pin in Digital mode, port read input enabled, A/D input multiplexer connected to AVSS
0 = Port pin in Analog mode, port read input disabled, A/D samples pin voltage

Note: Not all PxRDY bits are available on all devices. See the specific device data sheet for the available analog inputs.

Register 44-5: ADPCFG2: A/D Port Configuration Register 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as '0'

bit 7-0 **PCFG<23:16>:** A/D Port Configuration Control bits

1 = Port pin in Digital mode, port read input enabled, A/D input multiplexer connected to AVSS

0 = Port pin in Analog mode, port read input disabled, A/D samples pin voltage

Note: Not all PxRDY bits are available on all devices. See the specific device data sheet for the available analog inputs.

Register 44-6: ADCPC0: A/D Convert Pair Control Register 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN0	PEND0	SWTRG0	TRGSRC0<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN1:** Interrupt Request Enable 1 bit
1 = Enable the IRQ generation when requested conversion of channels AN3 and AN2 is completed
0 = IRQ is not generated

bit 14 **PEND1:** Pending Conversion Status 1 bit
1 = Conversion of channels AN3 and AN2 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG1:** Software Trigger 1 bit
1 = Start conversion of AN3 and AN2 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND1 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC1<4:0>:** Trigger 1 Source Selection bits
Selects trigger source for conversion of analog channels AN3 and AN2.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-6: ADCPC0: A/D Convert Pair Control Register 0 (Continued)

bit 7 **IRQEN0:** Interrupt Request Enable 0 bit
1 = Enable the IRQ generation when requested conversion of channels AN1 and AN0 is completed
0 = IRQ is not generated

bit 6 **PEND0:** Pending Conversion Status 0 bit
1 = Conversion of channels AN1 and AN0 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG0:** Software Trigger 0 bit
1 = Start conversion of AN1 and AN0 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND0 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC0<4:0>:** Trigger 0 Source Selection bits
Selects trigger source for conversion of analog channels AN1 and AN0.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-7: ADCPC1: A/D Convert Pair Control Register 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN2	PEND2	SWTRG2	TRGSRC2<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN3:** Interrupt Request Enable 3 bit
1 = Enable the IRQ generation when requested conversion of channels AN7 and AN6 is completed
0 = IRQ is not generated

bit 14 **PEND3:** Pending Conversion Status 3 bit
1 = Conversion of channels AN7 and AN6 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG3:** Software Trigger 3 bit
1 = Start conversion of AN7 and AN6 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND3 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC3<4:0>:** Trigger 3 Source Selection bits
Selects trigger source for conversion of analog channels AN7 and AN6.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-7: ADCPC1: A/D Convert Pair Control Register 1 (Continued)

bit 7 **IRQEN2:** Interrupt Request Enable 2 bit
1 = Enable the IRQ generation when requested conversion of channels AN5 and AN4 is completed
0 = IRQ is not generated

bit 6 **PEND2:** Pending Conversion Status 2 bit
1 = Conversion of channels AN5 and AN4 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG2:** Software Trigger 2 bit
1 = Start conversion of AN5 and AN4 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND2 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC2<4:0>:** Trigger 2 Source Selection bits
Selects trigger source for conversion of analog channels AN5 and AN4.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-8: ADCPC2: A/D Convert Pair Control Register 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN4	PEND4	SWTRG4	TRGSRC4<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN5:** Interrupt Request Enable 5 bit
1 = Enable the IRQ generation when requested conversion of channels AN11 and AN10 is completed
0 = IRQ is not generated

bit 14 **PEND5:** Pending Conversion Status 5 bit
1 = Conversion of channels AN11 and AN10 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG5:** Software Trigger 5 bit
1 = Start conversion of AN11 and AN10 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND5 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC5<4:0>:** Trigger 5 Source Selection bits
Selects trigger source for conversion of analog channels AN11 and AN10.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-8: ADCPC2: A/D Convert Pair Control Register 2 (Continued)

bit 7 **IRQEN4:** Interrupt Request Enable 4 bit
1 = Enable the IRQ generation when requested conversion of channels AN9 and AN8 is completed
0 = IRQ is not generated

bit 6 **PEND4:** Pending Conversion Status 4 bit
1 = Conversion of channels AN9 and AN8 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG4:** Software Trigger 4 bit
1 = Start conversion of AN9 and AN8 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND4 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC4<4:0>:** Trigger 4 Source Selection bits
Selects trigger source for conversion of analog channels AN9 and AN8.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-9: ADCPC3: A/D Convert Pair Control Register 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN6	PEND6	SWTRG6	TRGSRC6<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN7:** Interrupt Request Enable 7 bit
1 = Enable the IRQ generation when requested conversion of channels AN15 and AN14 is completed
0 = IRQ is not generated

bit 14 **PEND7:** Pending Conversion Status 7 bit
1 = Conversion of channels AN15 and AN14 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG7:** Software Trigger 7 bit
1 = Start conversion of AN15 and AN14 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND7 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC7<4:0>:** Trigger 7 Source Selection bits
Selects trigger source for conversion of analog channels AN15 and AN14.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-9: ADCPC3: A/D Convert Pair Control Register 3 (Continued)

bit 7 **IRQEN6:** Interrupt Request Enable 6 bit
1 = Enable the IRQ generation when requested conversion of channels AN13 and AN12 is completed
0 = IRQ is not generated

bit 6 **PEND6:** Pending Conversion Status 6 bit
1 = Conversion of channels AN13 and AN12 is pending. This is set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG6:** Software Trigger 6 bit
1 = Start conversion of AN13 and AN12 (if selected in TRGSRC bits). This bit is automatically cleared by hardware when the PEND6 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC6<4:0>:** Trigger 6 Source Selection bits
Selects trigger source for conversion of analog channels AN13 and AN12.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-10: ADCPC4: A/D Convert Pair Control Register 4

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN9	PEND9	SWTRG9	TRGSRC9<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN8	PEND8	SWTRG8	TRGSRC8<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN9:** Interrupt Request Enable 9 bit
1 = Enable IRQ generation when requested conversion of channels AN19 and AN18 is completed
0 = IRQ is not generated

bit 14 **PEND9:** Pending Conversion Status 9 bit
1 = Conversion of channels AN19 and AN18 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG9:** Software Trigger 9 bit
1 = Start conversion of AN19 and AN18. This bit is automatically cleared by hardware when the PEND9 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC9<4:0>:** Trigger 9 Source Selection bits
Selects trigger source for conversion of analog channels AN19 and AN18.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-10: ADCPC4: A/D Convert Pair Control Register 4 (Continued)

bit 7 **IRQEN8:** Interrupt Request Enable 8 bit
1 = Enable IRQ generation when requested conversion of channels AN17 and AN16 is completed
0 = IRQ is not generated

bit 6 **PEND8:** Pending Conversion Status 8 bit
1 = Conversion of channels AN17 and AN16 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG8:** Software Trigger 8 bit
1 = Start conversion of AN17 and AN16. This bit is automatically cleared by hardware when the PEND8 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC8<4:0>:** Trigger 8 Source Selection bits
Selects trigger source for conversion of analog channels AN17 and AN16.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-11: ADCPC5: A/D Convert Pair Control Register 5

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN11	PEND11	SWTRG11	TRGSRC11<4:0>				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN10	PEND10	SWTRG10	TRGSRC10<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **IRQEN11:** Interrupt Request Enable 11 bit
1 = Enable IRQ generation when requested conversion of channels AN23 and AN22 is completed
0 = IRQ is not generated

bit 14 **PEND11:** Pending Conversion Status 11 bit
1 = Conversion of channels AN23 and AN22 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 13 **SWTRG11:** Software Trigger 11 bit
1 = Start conversion of AN23 and AN22. This bit is automatically cleared by hardware when the PEND11 bit is set
0 = Conversion is not started

bit 12-8 **TRGSRC11<4:0>:** Trigger 11 Source Selection bits
Selects trigger source for conversion of analog channels AN23 and AN22.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-11: ADCPC5: A/D Convert Pair Control Register 5 (Continued)

bit 7 **IRQEN10:** Interrupt Request Enable 10 bit
1 = Enable IRQ generation when requested conversion of channels AN21 and AN20 is completed
0 = IRQ is not generated

bit 6 **PEND10:** Pending Conversion Status 10 bit
1 = Conversion of channels AN21 and AN20 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG10:** Software Trigger 10 bit
1 = Start conversion of AN21 and AN20. This bit is automatically cleared by hardware when the PEND10 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC10<4:0>:** Trigger 10 Source Selection bits
Selects trigger source for conversion of analog channels AN21 and AN20.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10110 = Reserved
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

Register 44-12: ADCPC6: A/D Convert Pair Control Register 6

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQEN12	PEND12	SWTRG12	TRGSRC12<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as '0'

bit 7 **IRQEN12:** Interrupt Request Enable 12 bit
1 = Enable IRQ generation when requested conversion of channels AN25 and AN24 is completed
0 = IRQ is not generated

bit 6 **PEND12:** Pending Conversion Status 12 bit
1 = Conversion of channels AN25 and AN24 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG12:** Software Trigger 12 bit
1 = Start conversion of AN25 and AN24. This bit is automatically cleared by hardware when the PEND12 bit is set
0 = Conversion is not started

bit 4-0 **TRGSRC12<4:0>:** Trigger 12 Source Selection bits
Selects trigger source for conversion of analog channels AN25 and AN24.
00000 = No conversion enabled
00001 = Individual software trigger selected
00010 = Global software trigger selected
00011 = PWM Special Event Trigger selected
00100 = PWM Generator 1 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01011 = PWM Generator 8 primary trigger selected
01100 = Timer1 period match
01101 = Reserved
01110 = PWM Generator 1 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10110 = PWM Generator 9 secondary trigger selected
10111 = PWM Generator 1 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11110 = PWM Generator 8 current-limit ADC trigger
11111 = Timer2 period match

44.3 ADC CONFIGURATION

44.3.1 ADC Clock Selection

The input clock source for the ADC module can be selected from the Auxiliary clock generator (ACLK) or the output of the Primary PLL (FVCO).

44.3.1.1 AUXILIARY CLOCK GENERATOR AS INPUT CLOCK FOR THE ADC MODULE

The Primary Oscillator Clock (POSCCLK) and Internal FRC Clock (FRCCLK) can be used with an auxiliary PLL to obtain the Auxiliary Clock (ACLK). The auxiliary PLL has a fixed 16x multiplication factor.

The Auxiliary Clock Control (ACLKCON) register selects the reference clock and enables the auxiliary PLL and output dividers for obtaining the necessary auxiliary clock. Equation 44-1 gives the relationship between the Reference Clock (REFCLK) input frequency and the Auxiliary Clock (ACLK) frequency.

Equation 44-1:

$$ACLK = (REFCLK * M)/N$$

Where,

$REFCLK$ = Internal FRC clock frequency (7.37 MHz), if the internal FRC is selected as the clock source.

$REFCLK$ = Primary Oscillator Clock frequency (POSCCLK), if the primary oscillator is selected as the clock source.

M = 16, if the auxiliary PLL is enabled by setting the ENAPLL (ACLKCON<15>) bit.

M = 1, if the auxiliary PLL is disabled.

N = Postscaler ratio selected by the Auxiliary Postscaler (APSTSCLR<2:0>) bits in the Auxiliary Clock Control (ACLKCON<2:0>) register.

The auxiliary clock for the ADC module can be derived from the system clock when the device is running in the primary PLL mode. Equation 44-2 gives the relationship between Primary PLL Clock (PLLCLK) frequency and Auxiliary Clock (ACLK) frequency.

Equation 44-2:

$$ACLK = (PLLCLK)/N$$

Where,

N = Postscaler ratio selected by the Auxiliary Postscaler (APSTSCLR<2:0>) bits in the Auxiliary Clock Control (ACLKCON<2:0>) register.

Note:	Some devices require that the primary PLL be configured to operate at a maximum of 30 MIPS or less if the primary PLL is selected as the clock source for the auxiliary clock. Check the device data sheet if this requirement applies to a particular device.

44.3.1.2 OUTPUT OF PRIMARY PLL (FVCO) AS INPUT CLOCK FOR THE ADC MODULE

The OSCCON register selects the Reference Clock (REFCLK) input frequency and enables the primary PLL. The PLLFBD register selects the PLL feedback divider while the CLKDIV register selects the PLL prescaler to generate the Primary PLL Clock (FVCO)

Equation 44-3:

$$FVCO = REFCLK * (M/N1)$$

Where,

REFCLK = Internal FRC clock frequency (7.37 MHz) if the internal FRC is selected as the clock source.

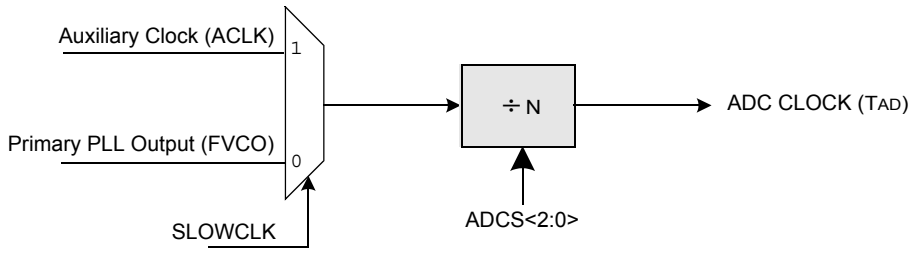
REFCLK = Primary Oscillator Clock frequency (POSCCLK), if the primary oscillator is selected as the clock source.

M = PLL Feedback Divider selection from the PLLFBD register (PLLDIV<8:0>).

N1 = PLL Phase Detector Input Divider Select bits from the CLKDIV register (PLLPRE<4:0>).

Refer to **Section 42. "Oscillator (Part IV)"** (DS70307) for more information on configuring the Auxiliary clock generator.

Figure 44-3: ADC Clock Generation



Note 1: Clock divider ratio selected using ADCS<2:0> bits.

Figure 44-3 shows the logic for ADC clock generation. The block diagram shows two ADC clock sources for the High-Speed 10-Bit ADC module. The input clock to the High-Speed 10-Bit ADC module is selected using the Enable The Slow Clock Divider (SLOWCLK) bit in the A/D Control Register (ADCON<12>) register.

- When SLOWCLK = 0, the primary PLL is chosen as the input clock to the High-Speed 10-Bit ADC module.
- When SLOWCLK = 1, the auxiliary clock is chosen as the input clock to the High-Speed 10-Bit ADC module.

The clock divider ratio is controlled by the ADC Conversion Clock Select (ADCS<2:0>) bits in the ADC Control (ADCON<2:0>) register. See Register 44-1 for more details on clock divider bit settings.

Note: The ADC clock period (TAD) should be within a range as specified in the "Electrical Characteristics" section of the device data sheet.

44.3.1.3 CONFIGURING ANALOG PORT PINS

The Analog/Digital Pin Configuration (ADPCFG and ADCPCFG2) and Port I/O Data Direction (TRISx) registers control the operation of the analog input pins. Refer to **Section 10. "I/O Ports"** (DS70193), for more information on the port I/O registers.

To configure a port pin as an analog input:

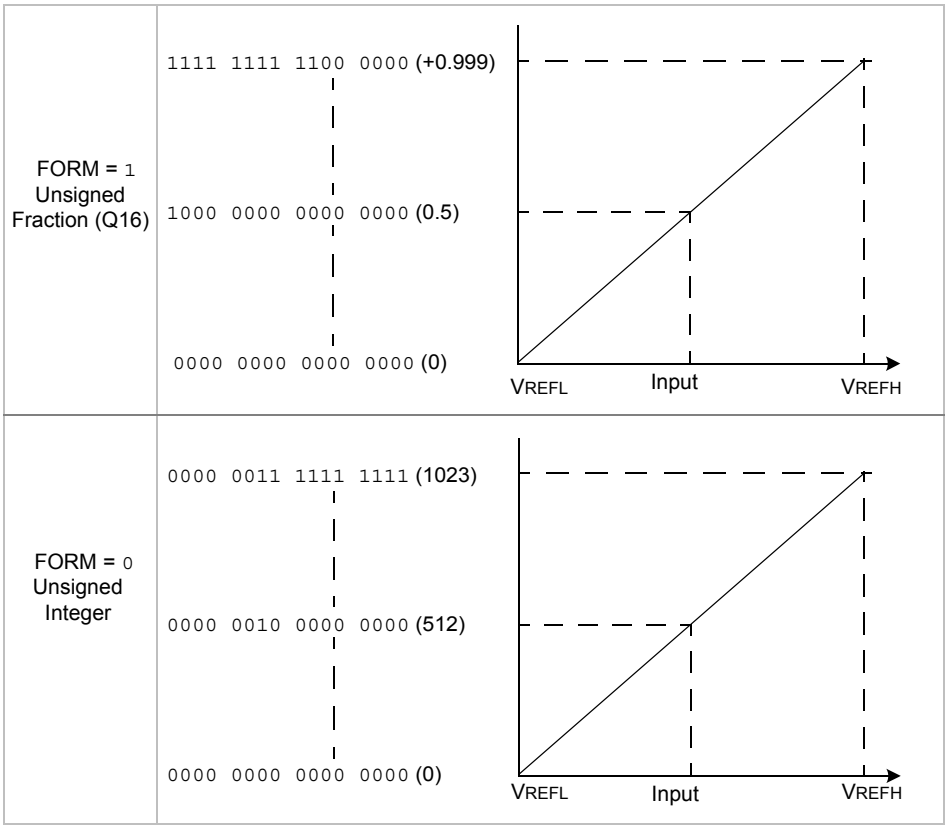
1. Clear the Pin Configuration bit (PCFGn = 0) in the ADPCFG and ADPCFG2 registers.
2. Set the Port I/O Direction bit (TRISn = 1) in the TRISx register.

Note 1:	When a port pin is configured as an analog input (PCFGn = 0), the Digital I/O Port register reads the pin as '0'.
2:	When a port pin is configured as a digital input (PCFGn = 1), the user application should apply digital input levels (V_{IL} and V_{IH}) only.

44.3.2 Selecting Output Data Format

The ADC result is available in two different numerical formats: Unsigned Integer and Unsigned Fraction (see Figure 44-4). The Data Output Format (FORM) bit in the ADC Control (ADCON<8>) register selects the output data format.

Figure 44-4: ADC Output Format



Note: The positive reference voltage is AV_{DD} (V_{REFH}). The negative reference voltage is AV_{SS} (V_{REFL}).

44.3.3 Enabling the High-Speed 10-Bit ADC Module

When the A/D Operating Mode (ADON) bit in the A/D Control Register (ADCON<15>) is set to '1', the module is in Active mode and is fully powered and functional. When the ADON bit is set to '0', the module is disabled. The digital and analog portions of the circuit are turned off for maximum current savings.

After enabling the High-Speed 10-Bit ADC module, the user application must wait for the analog stages to stabilize before starting the conversion. For the stabilization time, refer to the "Electrical Characteristics" section of the device data sheet.

Note:	The ASYNCSAMP, SEQSAMP, ORDER, EIE, ADCS, SLOWCLK and FORM bits should not be modified while ADON = 1. This would lead to indeterminate results.

44.3.4 Voltage Reference

The High-Speed 10-Bit ADC module uses analog supply pins (AV_{DD} and AV_{SS}) as voltage reference pins. The positive reference voltage is AV_{DD} ($V_{REF}+$) and the negative reference voltage is AV_{SS} ($V_{REF}-$). Refer to the "Electrical Characteristics" section in the device data sheet for specific information on the maximum and minimum values of AV_{DD} and AV_{SS}.

Note:	The High-Speed 10-Bit ADC module does not have external reference voltage pins.

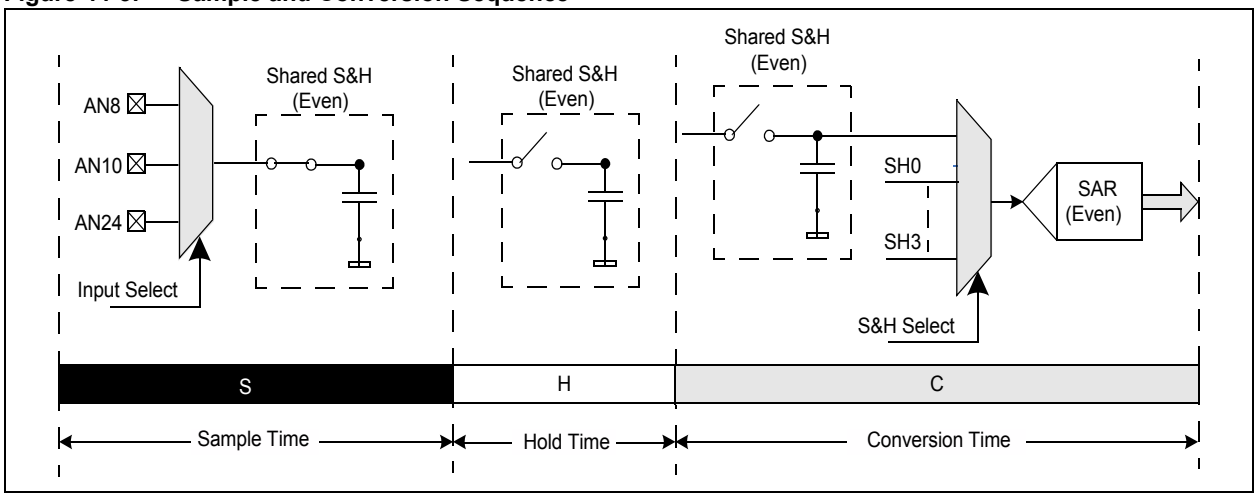
44.4 ADC CONVERSION

44.4.1 Basic Sample and Conversion Sequence

The analog-to-digital conversion is a three step process. Figure 44-5 illustrates each step of the process for an even numbered analog input that uses the shared (even) S&H circuit available on the dual SAR converter.

1. **Sample Time:** The analog multiplexer selects an analog input. The selected input is connected to the shared S&H circuit.
2. **Hold Time:** The shared S&H circuit is disconnected from the analog multiplexer. It now holds the analog input for a conversion.
3. **Conversion Time:** The analog input stored in the S&H circuit is converted to equivalent digital bits.

Figure 44-5: Sample and Conversion Sequence



44.4.1.1 SAMPLE TIME

During the sampling time, the selected analog input is connected to the S&H circuit capacitor. There is a minimum sample time to ensure that the S&H circuit provides the desired accuracy for the analog-to-digital conversion (refer to **44.10 "Transfer Function for 10-Bit ADC"**).

The following sampling modes are used in the High-Speed 10-Bit ADC module:

- Asynchronous Sampling Mode: In this mode, when not performing a conversion, the dedicated S&H circuit continuously samples the analog input. On a pair conversion request, the sampling process is terminated and the S&H circuit enters a hold state.
- Synchronous Sampling Mode: In this mode, the shared S&H circuit samples the analog input only on an ADC pair conversion request. The sampling time is 2 TAD clock cycles, where TAD is the ADC clock period.

44.4.1.2 CONVERSION TIME

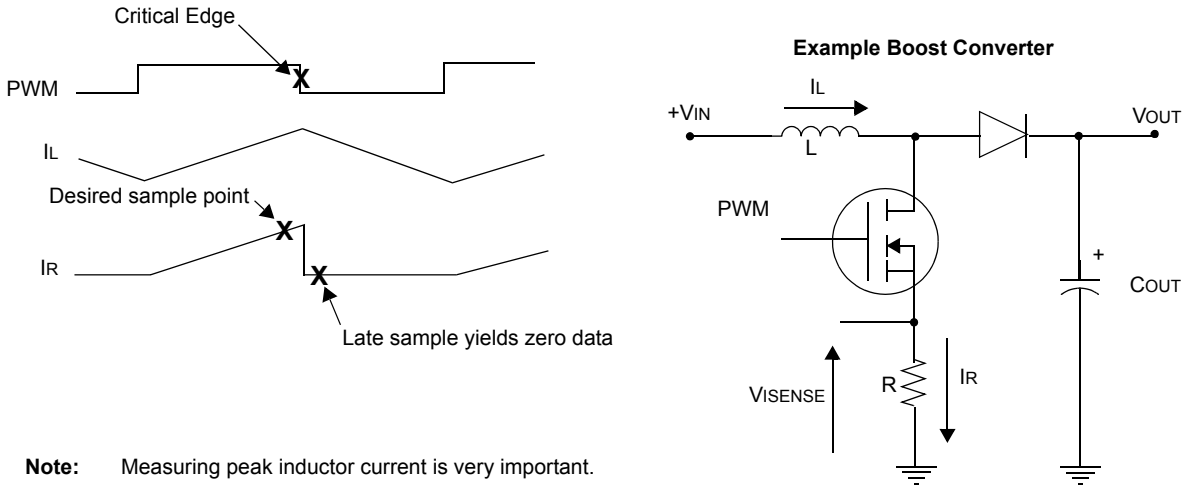
During the conversion time, the stored voltage in the selected S&H circuit is converted to equivalent digital bits. The conversion time is 14 TAD clock cycles.

44.4.2 Analog Input Pair

The High-Speed 10-Bit ADC module converts analog inputs in pairs. The 26 analog inputs available on the ADC module are grouped into seven analog input pairs. The analog input pair is a combination of an even and odd numbered analog input, such as AN0 and AN1, AN2 and AN3 and so on (see Figure 44-7). The technique of using pairs is particularly useful in power conversion applications that require voltage and current measurement for each PWM control loop.

Each of the first four analog input pairs in both single and dual SAR device has a dedicated S&H circuit to sample the even numbered analog input. For example, the dedicated S&H circuit (SH0) samples AN0, as shown in Figure 44-1. On a conversion request, the dedicated S&H circuit allows the corresponding analog input to be sampled without any latency (zero latency). For example, in the boost circuit (see Figure 44-6), the dedicated S&H circuit enables the peak inductor current measurement with zero latency. Any latency in sampling would lead to an incorrect result.

Figure 44-6: Power Conversion Application Example



Note: Measuring peak inductor current is very important.

44.4.2.1 ADC INPUT PAIR CONTROL REGISTERS

The High-Speed 10-Bit ADC module has up to four ADC Pair Control registers (ADCPC0, ADCPC1, ADCPC2, ADCPC3, ADCPC4, ADCPC5 and ADCPC6) that support all seven of the analog input pairs. These registers support each analog input pair using the following control bits:

- Trigger Source Select (TRGSRCx<4:0>) bits: These bits select a trigger source for an analog input pair.
- Software Trigger (SWTRGx) bit: This bit generates conversion request for an analog input pair in software.
- Interrupt Request Enable (IRQENx) bit: This bit enables an analog input pair to generate a common ADC interrupt.
- Conversion Pending Status (PENDx) bit: This bit indicates that a conversion is requested but has not yet finished.

44.4.2.2 ADC TRIGGER SOURCE

Each analog input pair receives a separate conversion request. The analog input pairs are triggered independently for conversion. An analog pair can be triggered by using any of the following sources:

- Individual software trigger
- Global software triggers
- PWM Special Event Trigger
- PWM generator ‘n’ primary trigger (where n = 1 through 8)
- PWM generator ‘n’ secondary trigger (where n = 1 through 9)
- PWM generator ‘n’ current limit trigger (where n = 1 through 8)
- Timer1 period match
- Timer2 period match

The trigger source is configured by the Trigger Source Selection (TRGSRCx<4:0>) bits in the ADC Pair Control (ADCPCx) registers (see Register 44-6). If multiple analog input pairs are triggered at the same time, the conversion requests are prioritized. The Analog Input Pair 0 (AN0 and AN1) has the highest priority and the Analog Input Pair 12 (AN24 and AN25) has the lowest priority.

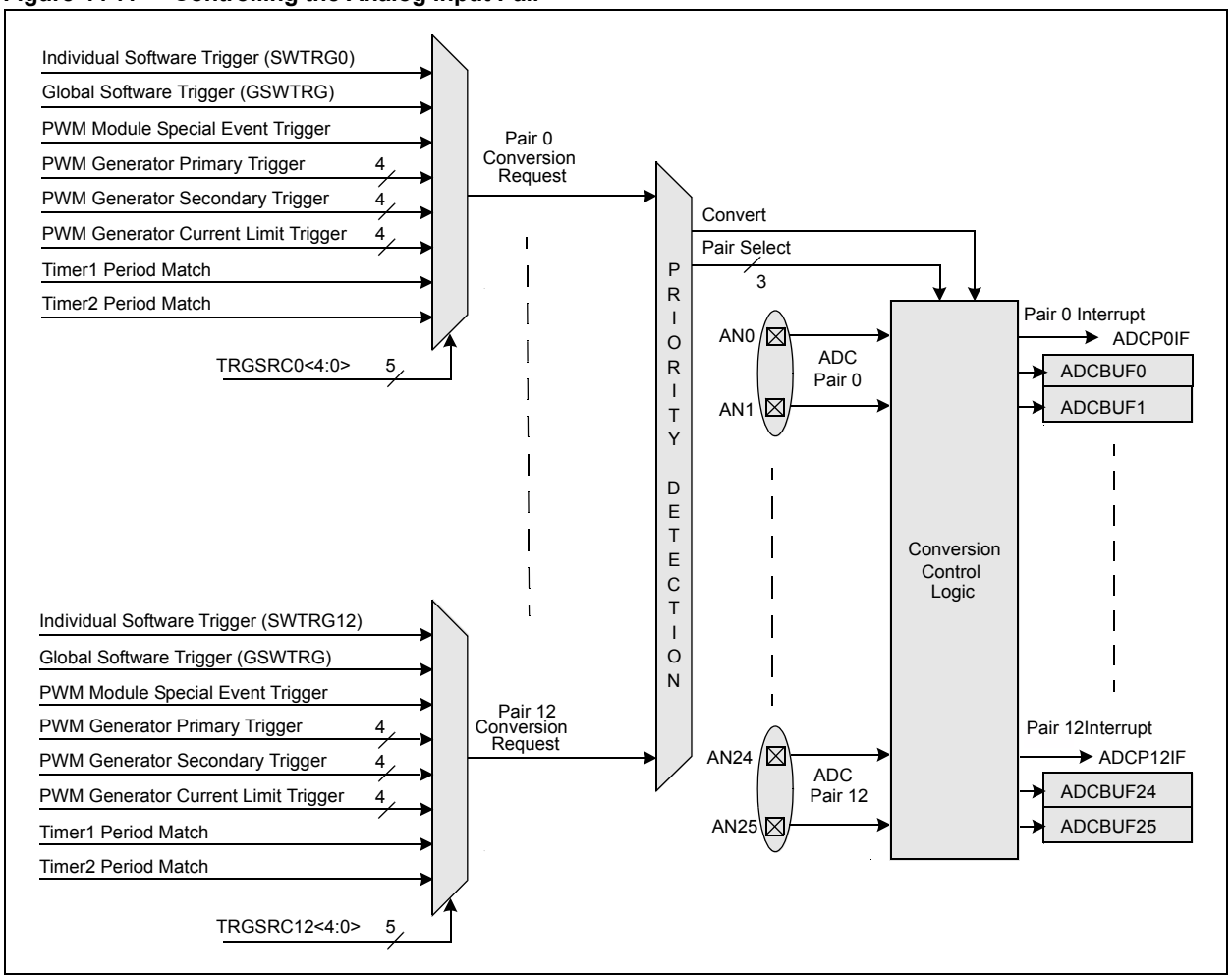
44.4.2.2.1 Software Trigger for Individual Pairs

Each ADC input pair can select an individual software trigger as a trigger source via the TRGSRCx<4:0> bits. After selecting the trigger source, the Software Trigger (SWTRGx) bit in the A/D Convert Pair Control Register (ADCPCx), when set, can generate a conversion request for the Analog Input Pair ‘x’. The SWTRGx bit is automatically cleared when the request is captured by the High-Speed 10-Bit ADC module.

44.4.2.2.2 Global Software Trigger

Each ADC input pair can select the global software trigger as a trigger source via the TRGSRCx<4:0> bits. After selecting the trigger source, the Global Software Trigger (GSWTRG) bit in the ADC control register (ADCON<10>), when set, can generate the conversion request for the selected analog input pairs. The GSWTRG bit is automatically cleared when the request is captured by the High-Speed 10-Bit ADC module.

Figure 44-7: Controlling the Analog Input Pair



44.4.2.3 RESULT REGISTER

Each analog input uses a dedicated result register to store the converted result. For example, AN0 conversion results are always stored in the ADCBUF0 register and AN1 conversion results are always stored in the ADCBUF1 register.

44.4.2.4 INDIVIDUAL ADC PAIR INTERRUPT

The High-Speed 10-Bit ADC module also provides individual interrupt outputs, one for each analog input pair. When an analog input pair is converted, the following occurs:

- The associated ADC pair interrupt flag (ADCPxIF) is set.
- If the ADC pair interrupt (ADCPxIE) is enabled, the ADC pair conversion interrupt is generated.

Refer to **Section 41. "Interrupts (Part IV)"** (DS70300) for more information on interrupt control and status bits.

The analog input pair also uses an associated Pending Conversion Status (PENDx) bit to indicate that a conversion is requested but has not yet finished. The PENDx bit is set when a trigger request for conversion is received and it is automatically cleared after the conversion is completed. Refer to **44.5 "Sample and Conversion Sequence for Single SAR ADC"** and **44.6 "Sample and Conversion Sequence for Dual SAR ADC"**, for more information on interrupt timings.

Note:	The PENDx bit is set based on the ADC clock. If the PENDx bit is to be used to determine the completion of conversion, poll the PENDx bit until it is set. This indicates that the conversion trigger has been issued. Poll the PENDx bit again until the bit gets cleared, indicating that the conversion is complete.

44.4.2.5 COMMON ADC INTERRUPT

The High-Speed 10-Bit ADC module can generate a common ADC interrupt request (ADIF) for multiple analog input pairs instead of generating an individual ADC pair interrupt (ADCPxIF). The common interrupt request can be generated by setting the Interrupt Request (IRQENx) bit in the ADC pair control register (ADCPCx). The common ADC interrupt is useful for applications that use a common software routine to process ADC interrupts for multiple analog input pairs. Refer to **44.8 "Common ADC Interrupt"**, for more information on handling common ADC interrupts.

44.5 SAMPLE AND CONVERSION SEQUENCE FOR SINGLE SAR ADC

This section illustrates the sample and conversion sequence for the single SAR ADC module in various bit configurations. The sample and conversion sequence is controlled by the following control bits:

- ASYNCSAMP (ADCON<4>): Asynchronous Sampling Select bit
- SEQSAMP(ADCON<5>): Sequential Sampling Select bit
- ORDER (ADCON<6>): Conversion Order Select bit

Note:	The SEQSAMP and ORDER bits have no effect on the dual SAR ADC operation.

44.5.1 Dedicated Sample and Hold (S&H)

The sampling techniques for the dedicated S&H circuit are selected using the ASYNCSAMP (ADCON<4>) bit.

44.5.1.1 ASYNCHRONOUS SAMPLING MODE

In this mode (ASYNCSAMP = 1), when not performing a conversion, the dedicated S&H circuit continuously samples the analog input. On a pair conversion request, the sampling process is terminated and the S&H circuit enters a hold state, thereby providing zero latency. The zero latency enables the dedicated S&H circuit to capture transitory information at a specific time instance. The user application must allow at least the minimum sampling time between each end of conversion and the new conversion request.

44.5.1.2 SYNCHRONOUS SAMPLING MODE

In this mode (ASYNCSAMP = 0), a pair conversion request is synchronized to the ADC clock domain (TAD) and it is prioritized with other requests. The sampling latency in synchronous sampling mode for various conditions are as follows:

- If a pair conversion request is generated when the High-Speed 10-Bit ADC module is idle, the corresponding dedicated S&H circuit samples the analog input in 2-3 TAD clock cycles.
- If a pair conversion request is generated when the High-Speed 10-Bit ADC module is busy, it has to wait for the High-Speed 10-Bit ADC module to become idle. When the High-Speed 10-Bit ADC module becomes idle, the dedicated S&H circuit for the selected analog input pair samples the analog input.
- If a multiple pair conversion request is generated simultaneously, the conversion requests are prioritized. Therefore, the conversion request having the highest priority is processed first and the lower priority requests will be processed in the order of their priority.

Note:	The ASYNCSAMP bit affects the dedicated S&H circuit only and has no effect on the shared S&H circuit.

44.5.2 Shared Sample and Hold (S&H)

The sampling technique for the Shared S&H circuit is selected using the SEQSAMP (ADCON<5>) bit.

44.5.2.1 SEQUENTIAL SAMPLING MODE

In this mode (SEQSAMP = 1), the shared S&H circuit samples the analog inputs just before the conversion.

44.5.2.2 SIMULTANEOUS SAMPLING MODE

In this mode (SEQSAMP = 0), the shared S&H circuit samples the analog input pair along with the dedicated S&H circuit. The even numbered analog input is sampled by the dedicated S&H circuit and the odd numbered input is sampled by the shared S&H circuit.

Note:	The SYNCSAMP bit affects the shared S&H circuit only and has no effect on the dedicated S&H circuit. Any pairs with both inputs on the shared S&H circuit will always be sampled sequentially and the SEQSAMP bit has no effect.

44.5.3 Conversion Order

If the normal order (ORDER = 0) is selected, the even numbered analog input is converted first, and then the odd numbered analog input is converted. If reversed order (ORDER = 1) is selected, the odd numbered analog input is converted first, and then the even numbered analog input is converted.

44.5.4 Sample Conversion Timing Diagrams of Single SAR ADC

In the single SAR ADC module, an analog input pair can be sampled either by both the dedicated and the shared S&H circuit or by the shared S&H circuit alone (see Figure 44-2).

Table 44-1 lists the sample conversion sequence for the analog input pairs that use the dedicated S&H circuit for even numbered analog inputs and the shared S&H circuit for odd numbered analog inputs.

Table 44-1: Sample Conversion Sequence

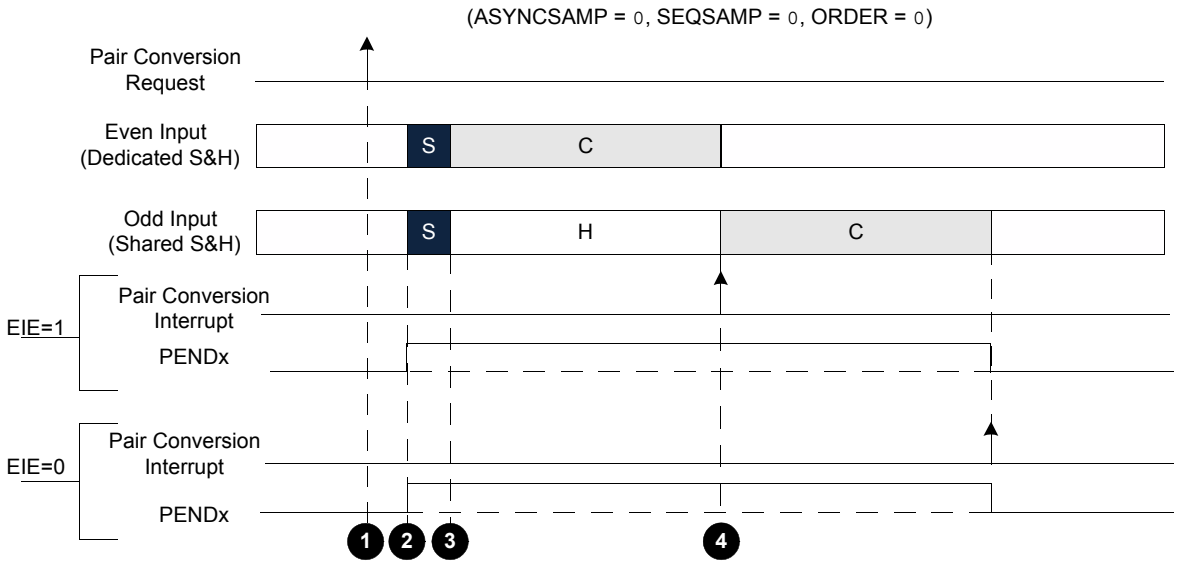
ASYNCSAMP	SEQSAMP	ORDER	Sample Conversion Sequence	See Figure
0	0	0	Synchronous and Simultaneous Sampling (normal order)	44-8
0	0	1	Synchronous and Simultaneous Sampling (reverse order)	44-9
0	1	0	Synchronous and Sequential Sampling (normal order)	44-10
0	1	1	Synchronous and Sequential Sampling (reverse order)	44-11
1	0	0	Asynchronous and Simultaneous Sampling (normal order)	44-12
1	0	1	Asynchronous and Simultaneous Sampling (reverse order)	44-13
1	1	0	Asynchronous and Sequential Sampling (normal order)	44-14
1	1	1	Asynchronous and Sequential Sampling (reverse order)	44-15

Table 44-2 lists the sample conversion sequence for different bit settings for analog input pairs that use the shared S&H circuit for both analog inputs.

Table 44-2: Sample Conversion Sequence

ASYNCSAMP	SEQSAMP	ORDER	Sample Conversion Sequence	See Figure
x	x	0	Synchronous Sampling (normal order)	44-16
x	x	1	Synchronous Sampling (reverse order)	44-17

Figure 44-8: Synchronous and Simultaneous Sampling (Normal Order)



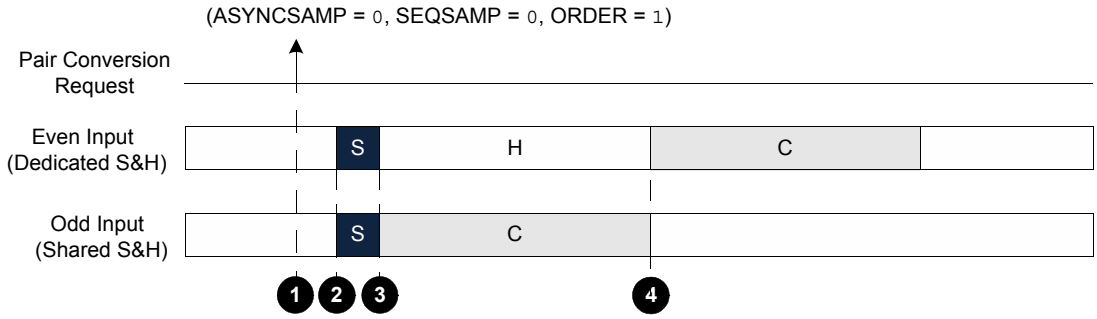
Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the even and odd numbered analog inputs are sampled simultaneously. The even numbered analog input is sampled by the dedicated S&H circuit and the odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-9: Synchronous and Simultaneous Sampling (Reverse Order)



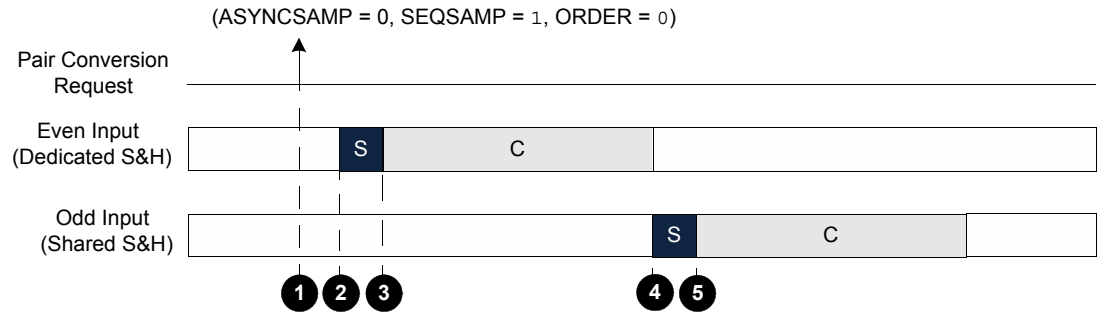
Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the even and odd numbered analog inputs are sampled simultaneously. The even numbered analog input is sampled by the dedicated S&H circuit and the odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-10: Synchronous and Sequential Sampling (Normal Order)



Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clocks.

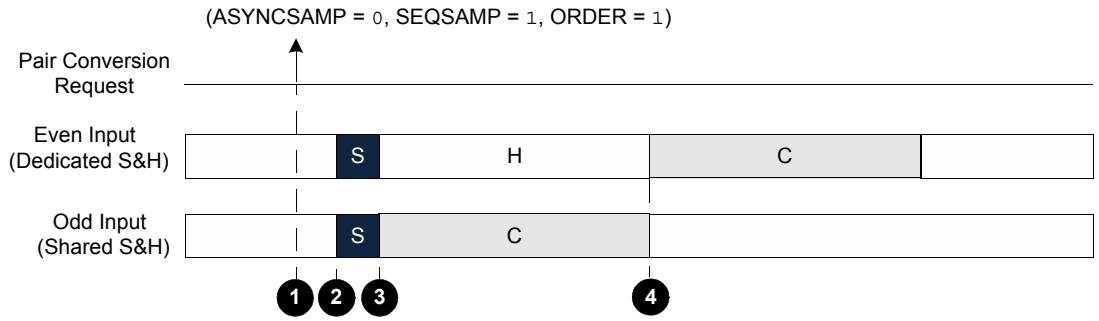
2: After the synchronization delay has elapsed, the even numbered analog input is sampled by the dedicated S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

5: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-11: Synchronous and Sequential Sampling (Reverse Order)



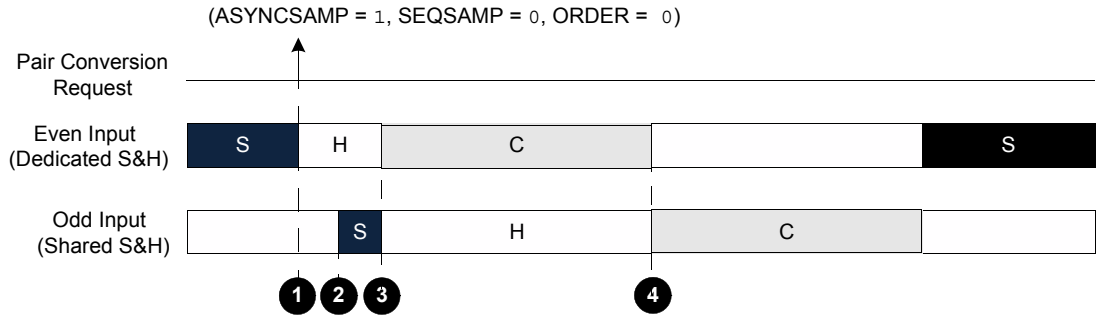
Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the even numbered analog input is sampled by the dedicated S&H and the odd numbered analog input is sampled by the shared S&H. The sampling time is 2 TAD clock cycles.

3: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-12: Asynchronous and Simultaneous Sampling (Normal Order)



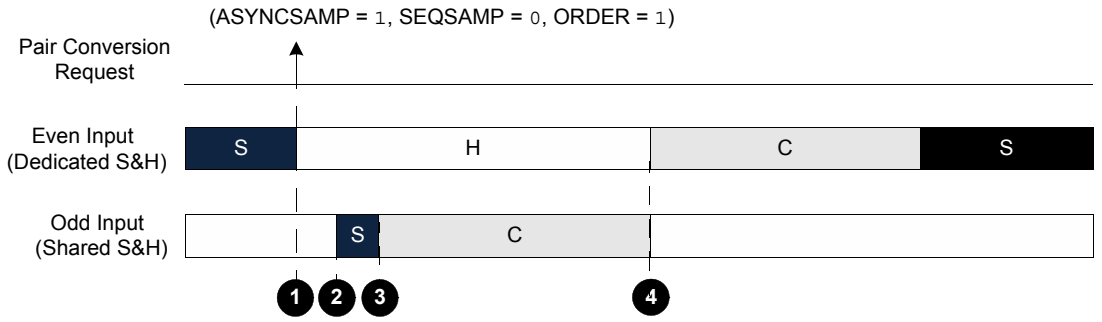
Note 1: In Asynchronous Sampling mode, the even numbered analog input is continuously sampled by the dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured by the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-13: Asynchronous and Simultaneous Sampling (Reverse Order)



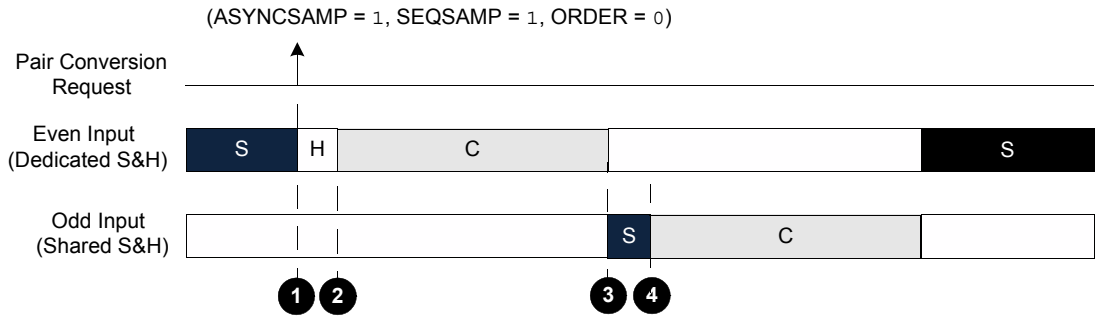
Note 1: In Asynchronous Sampling mode, the even numbered analog input is continuously sampled by dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after first conversion.

4: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-14: Asynchronous and Sequential Sampling (Normal Order)



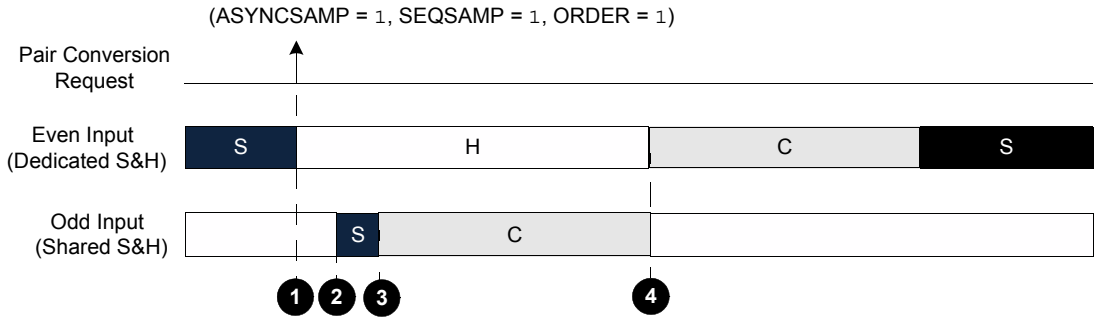
Note 1: In Asynchronous Sampling mode, the even numbered analog input is continuously sampled by the dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

3: The odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

4: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-15: Asynchronous and Sequential Sampling (Reverse Order)



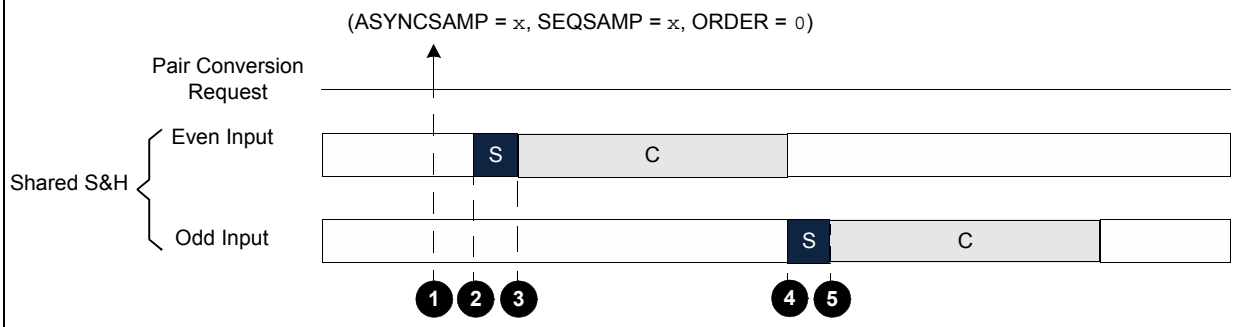
Note 1: In Asynchronous Sampling mode, the even numbered analog input is continuously sampled by the dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-16: Synchronous Sampling (Normal Order)



Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clocks.

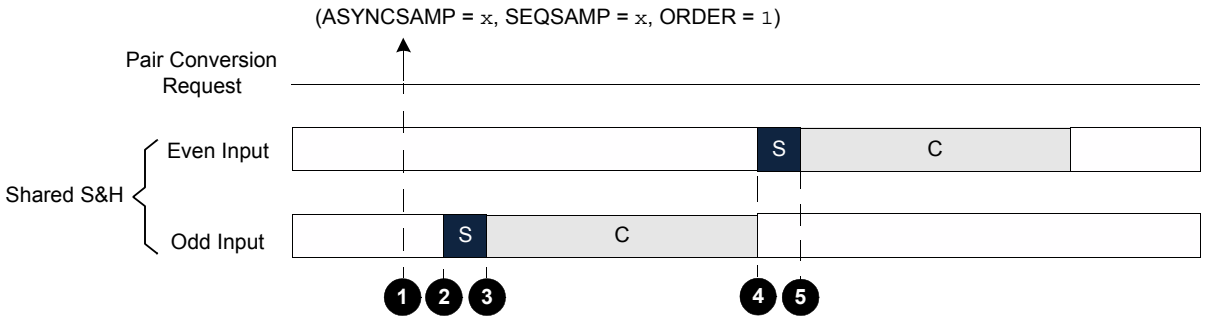
2: After the synchronization delay has elapsed, the even numbered analog input is first sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles. Since both the even numbered and the odd numbered analog input use the same shared sample and hold circuit, they are sampled and converted sequentially.

3: The even numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after the first conversion.

4: The odd numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

5: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after the second conversion.

Figure 44-17: Synchronous Sampling (Reverse Order)



Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start the sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, the odd numbered analog input is first sampled by shared S&H circuit. The sampling time is 2 TAD clock cycles. Since both the even numbered and the odd numbered analog input use the same shared sample and hold circuit, they are sampled and converted sequentially.

3: The odd numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is selected (EIE = 1), the ADC Pair conversion interrupt is generated after first conversion.

4: The even numbered analog input is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

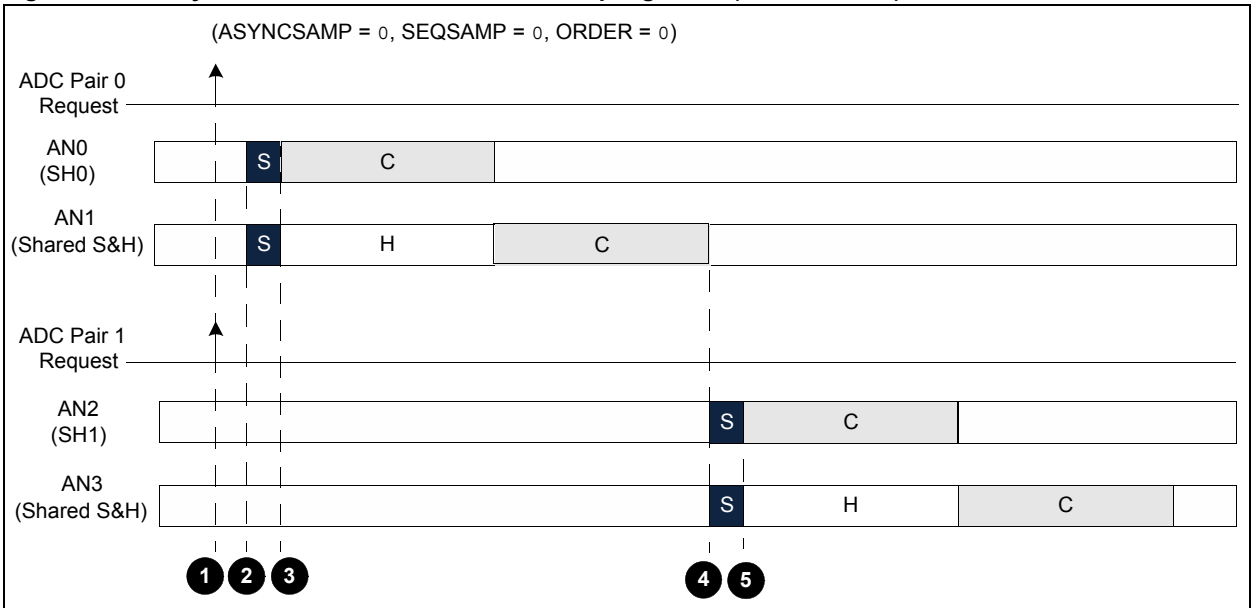
5: The even numbered analog input captured in the shared S&H circuit is converted to equivalent digital counts. If the early interrupt is not selected (EIE = 0), the ADC Pair conversion interrupt is generated after second conversion.

44.5.4.1 SIMULTANEOUS CONVERSION REQUESTS

This section describes the behavior of the High-Speed 10-Bit ADC module when multiple analog input pairs request conversion simultaneously. If multiple analog input pairs receive a conversion request at the same time, the conversion requests are prioritized. Analog Input Pair 0 has the highest priority and Analog Input Pair 12 has the lowest priority.

Figure 44-18 shows the sample conversion timing sequence when two analog input pairs, for example, Analog Input Pair 0 (AN1, AN0) and Analog Input Pair 1 (AN3, AN2) are triggered at the same time and are configured for Synchronous Sampling mode.

Figure 44-18: Synchronous and Simultaneous Sampling Mode (Normal Order)



Note 1: For Analog Input Pair 0 and Pair 1, the conversion request is generated at the same time. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, AN0 and AN1 are sampled simultaneously. AN0 is sampled by the dedicated S&H circuit (SH0) and AN1 is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

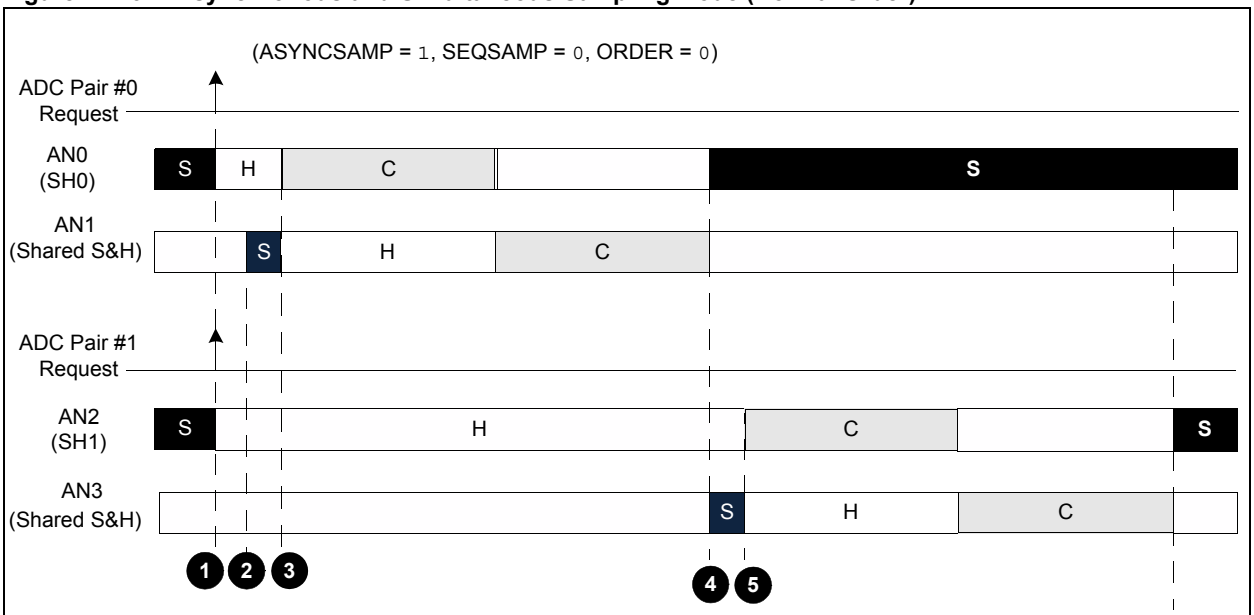
3: The analog input (AN0) captured in the dedicated S&H (SH0) circuit is converted first, and then the analog input (AN1) captured in the shared S&H circuit is converted.

4: AN2 and AN3 are sampled simultaneously. AN2 is sampled by the dedicated S&H circuit (SH1), and AN3 is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

5: The analog input (AN2) captured in the dedicated S&H circuit (SH1) is converted first, and then the analog input (AN3) captured in the shared S&H circuit is converted.

Figure 44-19 shows the sample conversion timing sequence when two Analog Input Pairs [for example, Analog Input Pair 0 (AN1, AN0) and Analog Input Pair 1 (AN3, AN2)] are triggered at the same time and are configured for Asynchronous Sampling mode. The analog inputs (AN0 and AN2) use the corresponding dedicated S&H circuit and the analog inputs (AN1 and AN3) use the shared S&H circuit for sampling.

Figure 44-19: Asynchronous and Simultaneous Sampling Mode (Normal Order)



Note 1: For Analog Input Pair 0 and Pair 1, the conversion request is generated at the same time. AN0 and AN1 are continuously sampled by the respective dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

2: After the synchronization delay has elapsed, AN1 is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

3: The analog input (AN0) captured in the dedicated S&H (SH0) circuit is converted first, and then the analog input (AN1) captured in shared S&H circuit is converted.

4: AN3 is sampled by the shared S&H circuit. The sampling time is 2 TAD clock cycles.

5: The analog input (AN2) captured in the dedicated S&H (SH1) circuit is converted first, and then the analog input (AN3) captured in shared S&H circuit is converted.

44.6 SAMPLE AND CONVERSION SEQUENCE FOR DUAL SAR ADC

In the dual SAR ADC module, an analog input pair can be sampled either by the dedicated S&H circuit and shared (odd) S&H circuit or by the shared (even) S&H circuit and shared (odd) S&H circuit (see Figure 44-1).

Note: The SEQSAMP and ORDER bits have no effect on the High-Speed 10-Bit ADC module with dual SARs.

Table 44-3 lists the sample conversion sequence for different bit settings for the analog input pairs that use the dedicated S&H circuit for even numbered analog inputs and the shared (odd) S&H circuit for odd numbered analog inputs.

Table 44-3: Sample Conversion Sequence

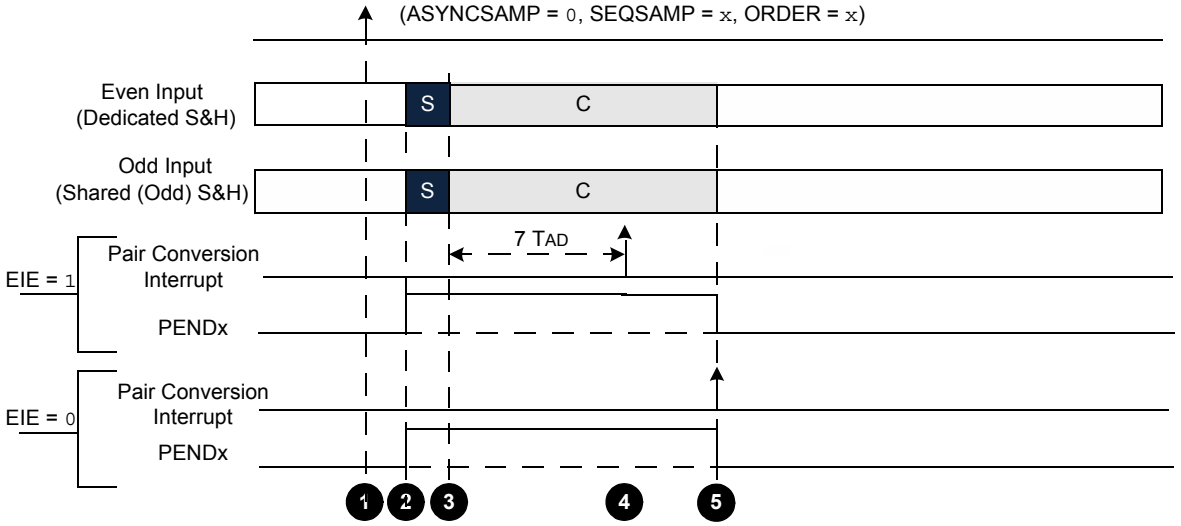
ASYNCSAMP	SEQSAMP	ORDER	Sample Conversion Sequence	See Figure
0	x	x	Synchronous Sampling and Parallel Conversion	44-20
1	x	x	Asynchronous Sampling and Parallel Conversion	44-21

Table 44-4 lists the sample conversion sequence for different bit settings for the analog input pairs that use the shared (even) S&H circuit for even numbered analog inputs and the shared (odd) S&H circuit for odd numbered analog inputs.

Table 44-4: Sample Conversion Sequence

ASYNCSAMP	SEQSAMP	ORDER	Sample Conversion Sequence	See Figure
x	x	x	Synchronous Sampling and Parallel Conversion	44-22

Figure 44-20: Synchronous Sampling and Parallel Conversion



Note 1: The ADC pair conversion request is generated by the CPU clock domain. To start sampling, it is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

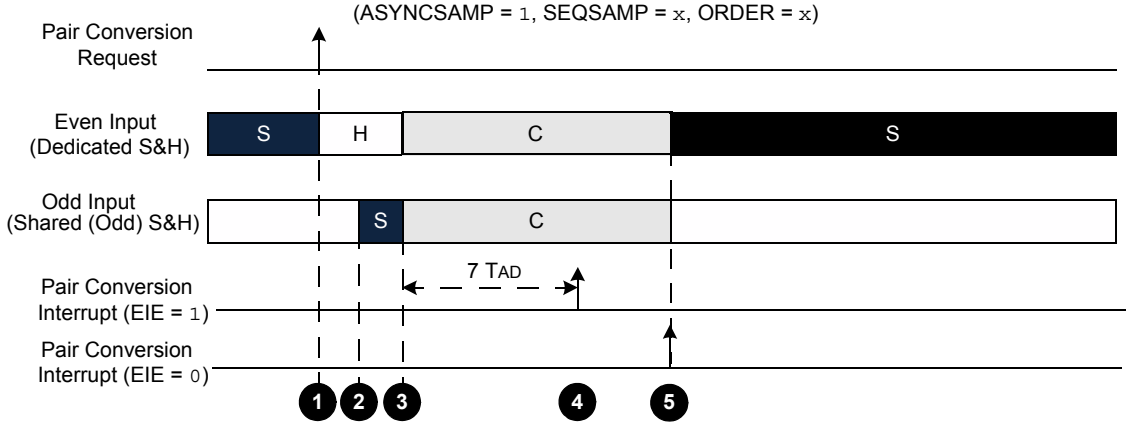
2: After the synchronization delay has elapsed, the even and odd numbered analog inputs are sampled simultaneously. The even numbered analog input is sampled by the dedicated S&H circuit, and the odd numbered analog input is sampled by the shared (odd) S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts by the even SAR. The odd numbered analog input captured in the shared (odd) S&H circuit is converted to equivalent digital counts by the odd SAR.

4: If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after 7 TAD clock cycles.

5: If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after completing the conversion.

Figure 44-21: Asynchronous Sampling and Parallel Conversion



Note 1: In Asynchronous Sampling mode, the even numbered analog input is continuously sampled in the dedicated S&H circuit. On an ADC pair conversion request, the sampling process is terminated instantaneously. The ADC pair conversion request from the CPU clock domain is synchronized with the ADC clock. The synchronization delay is about 2-3 TAD clock cycles.

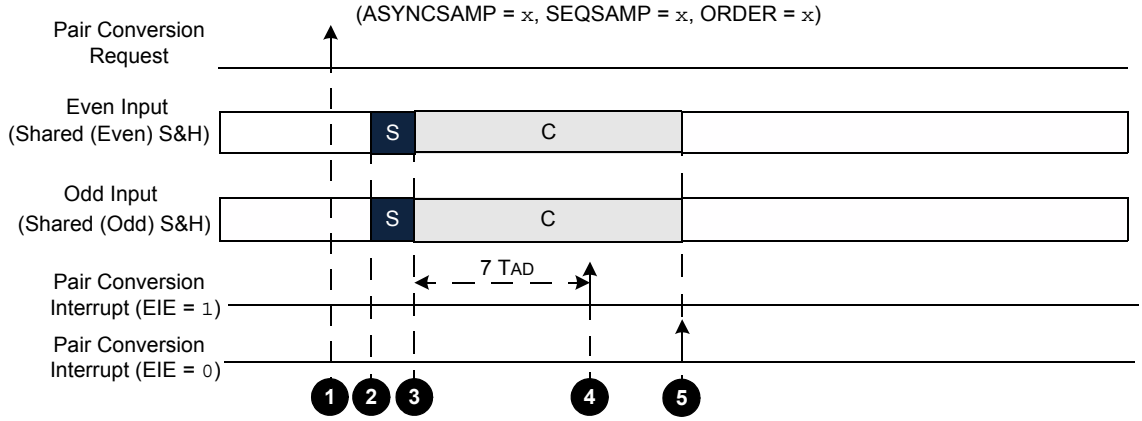
2: After the synchronization delay has elapsed, the odd numbered analog input is sampled by the shared (odd) S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured in the dedicated S&H circuit is converted to equivalent digital counts by the even SAR. The odd numbered analog input captured in the shared (odd) S&H circuit is converted to equivalent digital counts by the odd SAR.

4: If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after 7 TAD clock cycles.

5: If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after completing the conversion.

Figure 44-22: Synchronous Sampling and Parallel Conversion



Note 1: The ADC pair conversion request is generated by the CPU clock domain. It is synchronized with the ADC clock to start sampling. The synchronization delay is about 2-3 TAD clocks.

2: After the synchronization delay has elapsed, the even and odd numbered analog inputs are sampled simultaneously. The even numbered analog input is sampled by the shared (even) S&H circuit, and the odd numbered analog input is sampled by the shared (odd) S&H circuit. The sampling time is 2 TAD clock cycles.

3: The even numbered analog input captured in the shared (even) S&H circuit is converted to equivalent digital counts by the even SAR. The odd numbered analog input captured in the shared (odd) S&H circuit is converted to equivalent digital counts by the odd SAR.

4: If the early interrupt is selected (EIE = 1), the ADC pair conversion interrupt is generated after 7 TAD clock cycles.

5: If the early interrupt is not selected (EIE = 0), the ADC pair conversion interrupt is generated after completing the conversion.

44.7 ADC INTERRUPT

44.7.1 Individual ADC Interrupt

The High-Speed 10-Bit ADC module provides individual interrupt outputs, one for each analog input pair. Example 44-1 shows the code sequence that configures the High-Speed ADC module and generates an individual pair interrupt for Analog Input Pair 0 and Input Pair 1.

Example 44-1 shows the code sequence that configures the High-Speed 10-Bit ADC module and generates an individual pair interrupt.

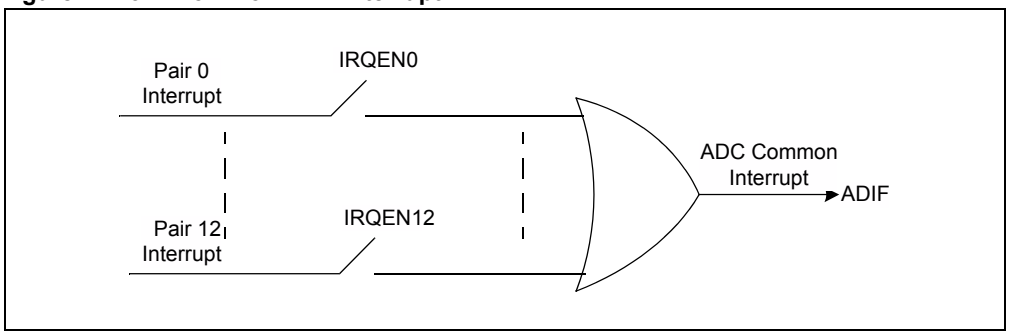
Example 44-1: Individual ADC Pair Interrupt

```
ADCONbits.FORM = 1; // Output in Integer Format
ADCONbits.EIE = 1; // Enable Early Interrupt
ADCONbits.ORDER = 0; // Normal Order of conversion
ADCONbits.SEQSAMP = 0; // Simultaneous sampling
ADCONbits.ASYNCSAMP = 1; // Asynchronous sampling
ADCONbits.SLOWCLK = 0; // High Frequency Clock input
ADCONbits.ADCS = 5; // Clock divider selection

ADCPC0bits.TRGSRC0=0b00100; // PWM Generator 1 Primary Trigger Selected
ADCPC0bits.TRGSRC1=0b00101; // PWM Generator 2 Primary Trigger Selected

ADPCFGbits.PCFG0 = 0; // AN0 is configured as analog input
ADPCFGbits.PCFG1 = 0; // AN1 is configured as analog input
ADPCFGbits.PCFG2 = 0; // AN2 is configured as analog input
ADPCFGbits.PCFG3 = 0; // AN3 is configured as analog input

IPC27bits.ADCP0IP = 0x01; // Set ADC Pair 0 Interrupt Priority (Level 1)
IFS6bits.ADCP0IF = 0; // Clear ADC Pair 0 Interrupt Flag
IEC6bits.ADCP0IE = 1; // Enable ADC Pair 0 Interrupt

IPC27bits.ADCP1IP = 0x02; // Set ADC Pair 1 Interrupt Priority (Level 2)
IFS6bits.ADCP1IF = 0; // Clear ADC Pair 1 Interrupt Flag
IEC6bits.ADCP1IE = 1; // Enable ADC Pair 1 Interrupt
ADCONbits.ADON = 1; // Enable ADC module

/* Example code for ADC Pair 0 ISR*/
void __attribute__((interrupt, no_auto_psv)) _ ADCP0Interrupt (void)
{
/* Interrupt Service Routine code goes here */

IFS6bits.ADPC0IF = 0; // Clear ADC Pair 0 Interrupt Flag
}



/* Example code for ADC Pair 1 ISR*/
void __attribute__((interrupt, no_auto_psv)) _ ADCP1Interrupt (void)
{
/* Interrupt Service Routine code goes here */

IFS6bits.ADPC1IF = 0; // Clear ADC Pair 1 Interrupt Flag
}
```

44.8 COMMON ADC INTERRUPT

The High-Speed 10-Bit ADC module can generate a common ADC interrupt (ADIF) for multiple analog input pairs instead of generating an individual ADC pair interrupt (ADCPxIF) for each pair (see Figure 44-23). An analog input pair can generate the common interrupt by setting the interrupt request (IRQENx) bit in the ADC pair control register (ADCPCx<15>). The common ADC interrupt is useful for applications that use a common software routine to process ADC interrupts for multiple analog input pairs.

Figure 44-23: Common ADC Interrupt



When the CPU receives a common ADC interrupt request, it does not know which ADC input pair has caused the request. To identify the analog pair that caused the request, software uses a unique offset that is generated in response to an active conversion pair request. In the ADC interrupt routine, the software can read the ADC Base (ADBASE) register that provides the sum of the contents of the ADBASE register and the offset based on the specific pair that causes the interrupt. Table 44-5 lists the offset values for different analog input pairs.

Table 44-5: Offset Value for Different ADC Pair Conversion Request

Analog Input Pair	Offset	ADBASE Value
Analog Input Pair 0	0	ADBASE + 0
Analog Input Pair 1	4	ADBASE + 4
Analog Input Pair 2	8	ADBASE + 8
Analog Input Pair 3	12	ADBASE + 12
Analog Input Pair 4	16	ADBASE + 16
Analog Input Pair 5	20	ADBASE + 20
Analog Input Pair 6	24	ADBASE + 24
Analog Input Pair 7	28	ADBASE + 28
Analog Input Pair 8	32	ADBASE + 32
Analog Input Pair 9	36	ADBASE + 36
Analog Input Pair 10	40	ADBASE + 40
Analog Input Pair 11	44	ADBASE + 44
Analog Input Pair 12	48	ADBASE + 48

The user application typically loads the ADBASE register with the base address of a jump table or the base address of an array of function pointers:

- A jump table in program memory contains branch instructions to branch to the appropriate pair handler. The offset value of ‘4’ reserves two instruction words per entry in the jump table.
- An array of function pointers in data memory can be initialized with the appropriate pair handler. The user application can use the ADBASE register value to call the specific pair handler. The offset value of ‘4’ allows a 24-bit function pointer.

In the common ADC Interrupt Service Routine, the value in the ADBASE register is used along with either a jump table or an array of function pointers to execute the specific pair handler. The user application must clear the ADC interrupt (ADIF) flag first, and then it should clear the specific pair data ready (PxRDY) flag that causes the ADC interrupt.

Note:	The individual ADC pair interrupt sets the associated ADC pair data ready (PxRDY) bit in the ADC Status (ADSTAT) register.

Example 44-2 shows the code sequence that configures the High-Speed 10-Bit ADC module and generates a common ADC interrupt.

Example 44-2: Common ADC Interrupt

```
#define CONVERSION_PAIRS 2

void ConvPair0Handler (void); // Declare the pair conversion handlers
void ConvPair1Handler (void);

void (*jumpTable[CONVERSION_PAIRS * 2 -1])(void);

main()
{

jumpTable[0] = &ConvPair0Handler; /* Set up the jump table*/
jumpTable[2] = &ConvPair1Handler;

ADCONbits.FORM = 1;       // Output in Integer Format
ADCONbits.EIE = 1;        // Enable Early Interrupt
ADCONbits.ORDER = 0;      // Normal Order of conversion
ADCONbits.SEQSAMP = 0;    // Simultaneous sampling
ADCONbits.ASYNCSAMP = 1;  // Asynchronous sampling
ADCONbits.SLOWCLK = 0;    // High Frequency Clock input
ADCONbits.ADCS = 5;       // Clock divider selection

ADCPC0bits.TRGSRC0=0b00100;// PWM Generator 1 Primary Trigger Selected
ADCPC0bits.IRQEN0=1;      // Enable common ADC Interrupt for Pair 0

ADCPC0bits.TRGSRC1=0b00101;// PWM Generator 2 Primary Trigger Selected
ADCPC0bits.IRQEN1=1; // Enable common ADC Interrupt for Pair 1

ADPCFGbits.PCFG0 = 0; // AN0 is configured as analog input
ADPCFGbits.PCFG1 = 0; // AN1 is configured as analog input
ADPCFGbits.PCFG2 = 0; // AN2 is configured as analog input
ADPCFGbits.PCFG3 = 0; // AN3 is configured as analog input

IPC3bits.ADIP = 0x01;  // Set Common ADC Interrupt Priority Level (Level 1)
IFS0bits.ADIF = 0;     // Clear ADC Pair 0 Interrupt Flag
IEC0bits.ADIE = 1;     // Enable ADC Pair 0 Interrupt
ADCONbits.ADON = 1; // Enable ADC module
While(1);
}

/* Example code for ADC ISR*/
void __attribute__((interrupt, no_auto_psv)) _ ADCInterrupt (void)
{

IFS0bits.ADIF = 0; // Clear ADC Pair 0 Interrupt Flag
( (void (*)()) *((int *)ADBASE))(); // Call the corresponding handler
}

void ConvPair0Handler (void)
{
    int an0, an1;

    an0 = ADCBUF0;         // Read AN0 conversion result
    an1 = ADCBUF1;         // Read AN1 conversion result
    ADSTATbits.P0RDY = 0;  // Clear the ADSTAT bits

}

void ConvPair1Handler (void)
{
    int an2, an3;

    an2 = ADCBUF2;         // Read AN2 conversion result
    an3 = ADCBUF3;         // Read AN3 conversion result
    ADSTATbits.P1RDY = 0;  // Clear the ADSTAT bits
}
```

44.9 OPERATION DURING SLEEP AND IDLE MODES

44.9.1 ADC Operation During CPU Sleep Mode

When the device enters Sleep mode, all clock sources to the High-Speed 10-Bit ADC module are shut down and stay at logic '0'. If the device enters Sleep mode in the middle of a conversion, the conversion is aborted. The converter does not resume a partially completed conversion on exiting from Sleep mode.

44.9.2 ADC Operation During CPU Idle Mode

When the device enters Idle mode, the system clock sources remain functional and the CPU stops code execution. The ADC Stop-in Idle (ADSIDL bit) in the ADC Control register (ADCON<13>) determines whether the module stops its operation in Idle mode, or continues to operate in Idle mode.

If ADSIDL = 0, the module continues to operate in Idle mode, providing full functionality. If enabled, the ADC interrupt wakes up the device from Idle mode, and the following occurs:

- If the assigned priority for the interrupt is less than or equal to the current CPU priority, the device wakes up and continues code execution from the instruction following the PWRSAV instruction that initiated Sleep mode.
- If the assigned priority level for the interrupt source is greater than the current CPU priority, the device wakes up and the CPU exception process begins. Code execution continues from the first instruction of the ADC ISR.

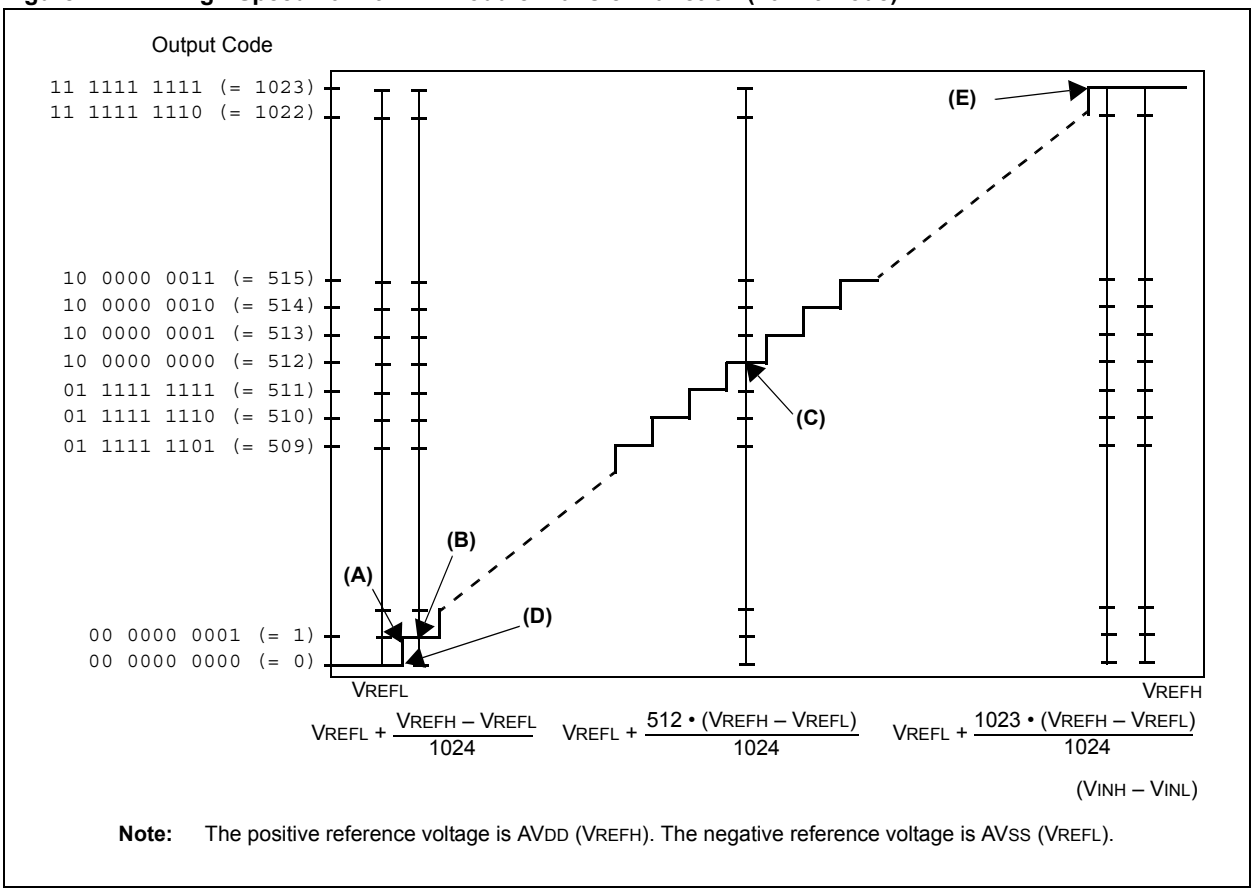
If ADSIDL = 1, the module stops its operation in Idle mode. If the device enters Idle mode in the middle of a conversion, the conversion is aborted. The converter will not resume a partially completed conversion on exiting from Idle mode.

44.10 TRANSFER FUNCTION FOR 10-BIT ADC

The ideal transfer function of the High-Speed 10-Bit ADC module is shown in Figure 44-24. The difference of the input voltages, (VINH – VINL), is compared to the reference, (VREFH – VREFL).

- The first code transition (**A**) occurs when the input voltage is (VREFH – VREFL/2048) or 0.5 LSb.
- The `00 0000 0001` code is centered at (VREFH – VREFL/1024) or 1.0 LSb (**B**).
- The `10 0000 0000` code is centered at (512 • (VREFH – VREFL)/1024) (**C**).
- An input voltage less than (1 • (VREFH – VREFL)/2048) converts as `00 0000 0000` (**D**).
- An input greater than (2045 • (VREFH – VREFL)/2048) converts as `11 1111 1111` (**E**).

Figure 44-24: High-Speed 10-Bit ADC Module Transfer Function (10-Bit Mode)



Note: The positive reference voltage is AVDD (VREFH). The negative reference voltage is AVSS (VREFL).

44.11 REGISTER MAP

The following table lists the special function registers, including their addresses and formats. All unimplemented registers and/or bits within a register are read as '0'.

Table 44-6: High-Speed 10-bit ADC Register Map

SFR Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ADCON	ADON	—	ADSIDL	SLOWCLK	—	GSWTRG	—	FORM	EIE	ORDER	SEQSAMP	ASYNCSAMP	—	ADCS<2:0>			0003
ADPCFG	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
ADPCFG2	—	—	—	—	—	—	—	—	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16	0000
ADSTAT	—	—	—	P12RDY	P11RDY	P10RDY	P9RDY	P8RDY	P7RDY	P6RDY	P5RDY	P4RDY	P3RDY	P2RDY	P1RDY	P0RDY	0000
ADBASE	ADBASE<15:1>															—	0000
ADCPC0	IRQEN1	PEND1	SWTRG1	TRGSRC1<4:0>					IRQEN0	PEND0	SWTRG0	TRGSRC0<4:0>					0000
ADCPC1	IRQEN3	PEND3	SWTRG3	TRGSRC3<4:0>					IRQEN2	PEND2	SWTRG2	TRGSRC2<4:0>					0000
ADCPC2	IRQEN5	PEND5	SWTRG5	TRGSRC5<4:0>					IRQEN4	PEND4	SWTRG4	TRGSRC4<4:0>					0000
ADCPC3	IRQEN7	PEND7	SWTRG7	TRGSRC7<4:0>					IRQEN6	PEND6	SWTRG6	TRGSRC6<4:0>					0000
ADCPC4	IRQEN9	PEND9	SWTRG9	TRGSRC9<4:0>					IRQEN8	PEND8	SWTRG8	TRGSRC8<4:0>					0000
ADCPC5	IRQEN11	PEND11	SWTRG11	TRGSRC11<4:0>					IRQEN10	PEND10	SWTRG10	TRGSRC10<4:0>					0000
ADCPC6	—	—	—	—	—	—	—	—	IRQEN12	PEND12	SWTRG12	TRGSRC12<4:0>					0000
ADCBUF0	ADC Data Buffer 0																xxxx
ADCBUF1	ADC Data Buffer 1																xxxx
ADCBUF2	ADC Data Buffer 2																xxxx
ADCBUF3	ADC Data Buffer 3																xxxx
ADCBUF4	ADC Data Buffer 4																xxxx
ADCBUF5	ADC Data Buffer 5																xxxx
ADCBUF6	ADC Data Buffer 6																xxxx
ADCBUF7	ADC Data Buffer 7																xxxx
ADCBUF8	ADC Data Buffer 8																xxxx
ADCBUF9	ADC Data Buffer 9																xxxx
ADCBUF10	ADC Data Buffer 10																xxxx
ADCBUF11	ADC Data Buffer 11																xxxx
ADCBUF12	ADC Data Buffer 12																xxxx
ADCBUF13	ADC Data Buffer 13																xxxx
ADCBUF14	ADC Data Buffer 14																xxxx
ADCBUF15	ADC Data Buffer 15																xxxx
ADCBUF16	ADC Data Buffer 16																xxxx
ADCBUF17	ADC Data Buffer 17																xxxx

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Table 44-6: High-Speed 10-bit ADC Register Map (Continued)

SFR Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ADCBUF18	ADC Data Buffer 18																xxxx
ADCBUF19	ADC Data Buffer 19																xxxx
ADCBUF20	ADC Data Buffer 20																xxxx
ADCBUF21	ADC Data Buffer 21																xxxx
ADCBUF22	ADC Data Buffer 22																xxxx
ADCBUF23	ADC Data Buffer 23																xxxx
ADCBUF24	ADC Data Buffer 24																xxxx
ADCBUF25	ADC Data Buffer 25																xxxx

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

44.12 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the dsPIC33F product family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the High-Speed 10-Bit ADC module are:

Title	Application Note #

No related application notes at this time.

Note:	Please visit the Microchip web site (www.microchip.com) for additional Application Notes and code examples for the dsPIC33F family of devices.

44.13 REVISION HISTORY

Revision A (January 2008)

This is the initial release of this document.

Revision B (July 2008)

This revision incorporates the following updates:

- Figures:
  - Updated the analog input channel AN6 in Figure 44-2, from being an input to the dedicated Sample and Hold (S&H) circuit to being an input to the shared Sample and Hold (S&H) circuit.
  - Updated the incorrect result buffer numbers as ADCBUF0 and ADCBUF1 (see Figure 44-7).
  - Updated the incorrect result buffer numbers in Figure 44-7. Replaced ADCBUF13 with ADCBUF0 and ADCBUF14 with ADCBUF1.
- Notes:
  - Added a note on the behavior of the PENDx bit (ADCPCx<14>) in **44.4.2.4 "Individual ADC Pair Interrupt"**.
  - Added a note for configuring the auxiliary clock (see **44.3.1 "ADC Clock Selection"**).
- Registers:
  - Updated the incorrect bit descriptions for all bits in ADBASE: A/D Base Register (see Register 44-3)
  - Updated the bit descriptions for bit 15 and bit 7 in the following registers: Register 44-6, Register 44-7 and Register 44-8.
  - Updated the bit description for bit 7 in Register 44-9.
- Sections:
  - Updated the conversion time as 14 TAD clock cycles in **44.4.1.2 "Conversion Time"**.
- Tables:
  - Corrected the ADCON reset value as 0003 in the High-Speed 10-bit ADC Register Map table (see Table 44-6).
- Additional minor corrections such as language and formatting updates are incorporated throughout the document.

Revision C (February 2009)

This revision has been updated in support of additional ADC and PWM channels. These updates are reflected in the following areas:

- Figures:
  - **Figure 44-1: "High-Speed 10-Bit ADC with Two SAR Converters"**
  - **Figure 44-2: "High-Speed 10-Bit ADC with One SAR Converter"**
  - **Figure 44-3: "ADC Clock Generation"**
  - **Figure 44-5: "Sample and Conversion Sequence"**
  - **Figure 44-7: "Controlling the Analog Input Pair"**
  - **Figure 44-23: "Common ADC Interrupt"**
- Registers:
  - Added the ADCPCFG2 register (see Register 44-5)
  - Added the ADCPC4 register (see Register 44-10)
  - Added the ADCPC5 register (see Register 44-11)
  - Added the ADCPC6 register (see Register 44-12)
  - Updated definitions for SLOWCLK (bit 12) and ADCS (bits 2-0) in the ADCON register (see Register 44-1)
  - Updated the ADSTAT register by adding PxRDY bit definitions (see Register 44-2)
  - Updated the ADPCFG register by adding PCFGx bit definitions (see Register 44-4)

Revision C (February 2009) (Continued)

- Registers (Continued)
  - Updated trigger source details in the ADCPC0, ADCPC1, ADCPC2 and ADCPC3 registers (see Register 44-6, Register 44-7, Register 44-8 and Register 44-9)
  - Added IRQEN7, PEND7, SWTRG7, and TRGSRC7<4:0> bit definitions to the ADCPC3 register (see Register 44-9)
- Sections:
  - Major updates were made to the text in section **44.3.1 “ADC Clock Selection”**
- Tables:
  - Added details for Analog Input Pair 7 through Pair 12 (see Table 44-5)
  - Updated the register map to reflect new bits and registers (see Table 44-6)
- Additional minor corrections such as language and formatting updates are incorporated throughout the document.