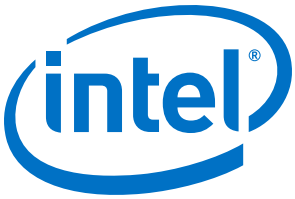


Intel® MAX® 10 Analog to Digital Converter User Guide

Updated for Intel® Quartus® Prime Design Suite: **17.1**

UG-M10ADC | 2017.12.15
Latest document on the web: **PDF** | **HTML**

Contents

1 Intel® MAX® 10 Analog to Digital Converter Overview

Intel® MAX® 10 devices feature up to two analog-to-digital converters (ADC). The ADCs provide the Intel MAX 10 devices with built-in capability for on-die temperature monitoring and external analog signal conversion.

The ADC solution consists of hard IP blocks in the Intel MAX 10 device periphery and soft logic through the Altera Modular ADC IP core.

The ADC solution provides you with built-in capability to translate analog quantities to digital data for information processing, computing, data transmission, and control systems. The basic function is to provide a 12 bit digital representation of the analog signal being observed.

The ADC solution works in two modes:

- Normal mode—monitors single-ended external inputs with a cumulative sampling rate of up to 1 million samples per second (MSPS):
  - Single ADC devices—up to 17 single-ended external inputs (one dedicated analog and 16 dual function input pins)
  - Dual ADC devices—up to 18 single-ended external inputs (one dedicated analog and eight dual function input pins in each ADC block)
- Temperature sensing mode—monitors external temperature data input with a sampling rate of up to 50 kilosamples per second. In dual ADC devices, only the first ADC block supports this mode.

Related Links

- Intel MAX 10 ADC Architecture and Features on page 11
- Intel MAX 10 ADC Design Considerations on page 34
- Intel MAX 10 ADC Implementation Guides on page 39
- Altera Modular ADC and Altera Modular Dual ADC IP Cores References on page 46
- Intel MAX 10 Getting Started
- Intel MAX 10 Online Training
- Intel MAX 10 How-to Videos
- How to Create ADC Design in Intel MAX 10 Device Using Platform Designer (Standard) Tool

  Provides video instruction that demonstrates how to create the ADC design in Intel MAX 10 devices using the Qsys system integration tool within the Intel Quartus Prime software and how to use the ADC toolkit to view the measured analog signal.

- How to Create Simultaneous Measurement with Intel MAX 10 ADC, Part 1
  Provides the first part of video instruction series that explains the differences between the Intel MAX 10 Altera Modular ADC and Altera Modular Dual ADC IP cores. The video also demonstrates how to create a simple simultaneous ADC measurement and how to place signal taps to measure the digital code output for analog signal.
- How to Create Simultaneous Measurement with Intel MAX 10 ADC, Part 2
  Provides the second part of video instruction series that explains the differences between the Intel MAX 10 Altera Modular ADC and Altera Modular Dual ADC IP cores. The video also demonstrates how to create a simple simultaneous ADC measurement and how to place signal taps to measure the digital code output for analog signal.

1.1 ADC Block Counts in Intel MAX 10 Devices

The ADC block is available in single and dual supply Intel MAX 10 devices.

Table 1. Number of ADC Blocks in Intel MAX 10 Devices and Packages

For more information about the device part numbers that feature ADC blocks, refer to the device overview.

Package	Power Supply	Device					
		10M04	10M08	10M16	10M25	10M40	10M50
M153	Single	1	1	—	—	—	—
U169	Single	1	1	1	—	—	—
U324	Single	1	1	1	—	—	—
	Dual	1	1	1	—	—	—
F256	Dual	1	1	1	2	2	2
E144	Single	1	1	1	1	1	1
F484	Dual	—	1	1	2	2	2
F672	Dual	—	—	—	—	2	2

Related Links

Intel MAX 10 FPGA Device Overview

1.2 ADC Channel Counts in Intel MAX 10 Devices

Different Intel MAX 10 devices support different number of ADC channels.

Table 2. ADC Channel Counts in Intel MAX 10 Devices

- Devices with two ADC blocks have two dedicated analog inputs and each ADC block has 8 dual function pins. You can use the dual function pins in an ADC block as general purpose I/O (GPIO) pins if you do not use the ADC.
- For more information about the device part numbers that feature ADC blocks, refer to the device overview.

Package	Pin Type	ADC Channel Counts Per Device					
		10M04	10M08	10M16	10M25	10M40	10M50
M153	Dedicated	1	1	—	—	—	—
	Dual function	8	8	—	—	—	—
U169	Dedicated	1	1	1	—	—	—
	Dual function	8	8	8	—	—	—
U324	Dedicated	1	1	1	—	—	—
	Dual function	16	16	16	—	—	—
F256	Dedicated	1	1	1	2	2	2
	Dual function	16	16	16	16	16	16
E144	Dedicated	1	1	1	1	1	1
	Dual function	8	8	8	8	8	8
F484	Dedicated	—	1	1	2	2	2
	Dual function	—	16	16	16	16	16
F672	Dedicated	—	—	—	—	2	2
	Dual function	—	—	—	—	16	16

Related Links

- Intel MAX 10 FPGA Device Overview
- Intel MAX 10 ADC Vertical Migration Support on page 7

1.3 Intel MAX 10 ADC Vertical Migration Support

Figure 1. ADC Vertical Migration Across Intel MAX 10 Devices

The arrows indicate the ADC migration paths. The devices included in each vertical migration path are shaded.

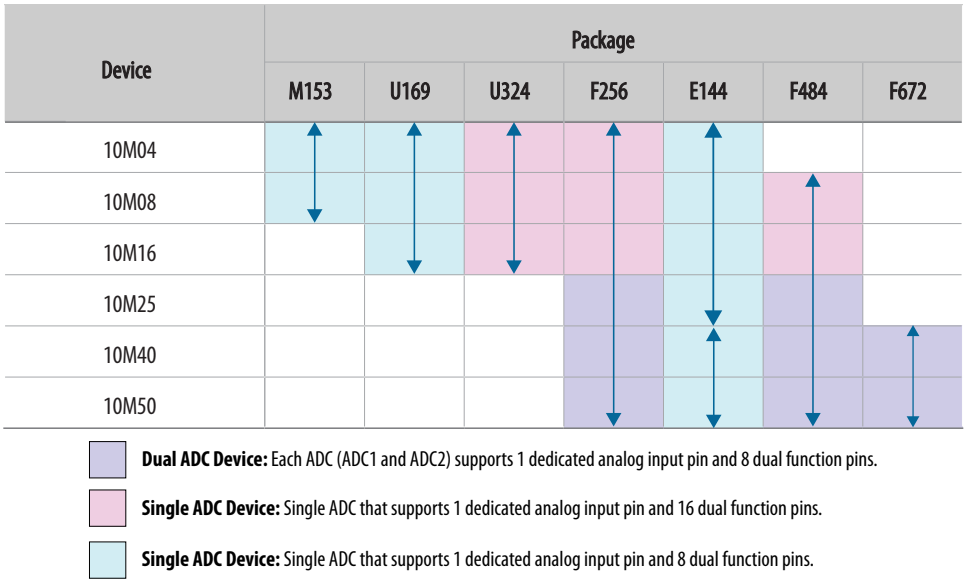


Table 3. Pin Migration Conditions for ADC Migration

Source	Target	Migratable Pins
Single ADC device	Single ADC device	You can migrate all ADC input pins
Dual ADC device	Dual ADC device	
Single ADC device	Dual ADC device	• One dedicated analog input pin. • Eight dual function pins from the ADC1 block of the source device to the ADC1 block of the target device.
Dual ADC device	Single ADC device	

Related Links

ADC Channel Counts in Intel MAX 10 Devices on page 6

1.4 Intel MAX 10 Single or Dual Supply Devices

Intel MAX 10 devices are available in single or dual supply packages.

- For devices with single power supply:
  - Use on chip regulator to power up the digital supply.
  - Use V_{CCA} to power up the ADC analog.
- For dual power supply devices, you must provide external power supplies of 1.2 V and 2.5 V to power up the ADC.

To choose the correct device, refer to the Intel MAX 10 device overview.

For more information about the ADC parameter, refer to the device datasheet.

Related Links

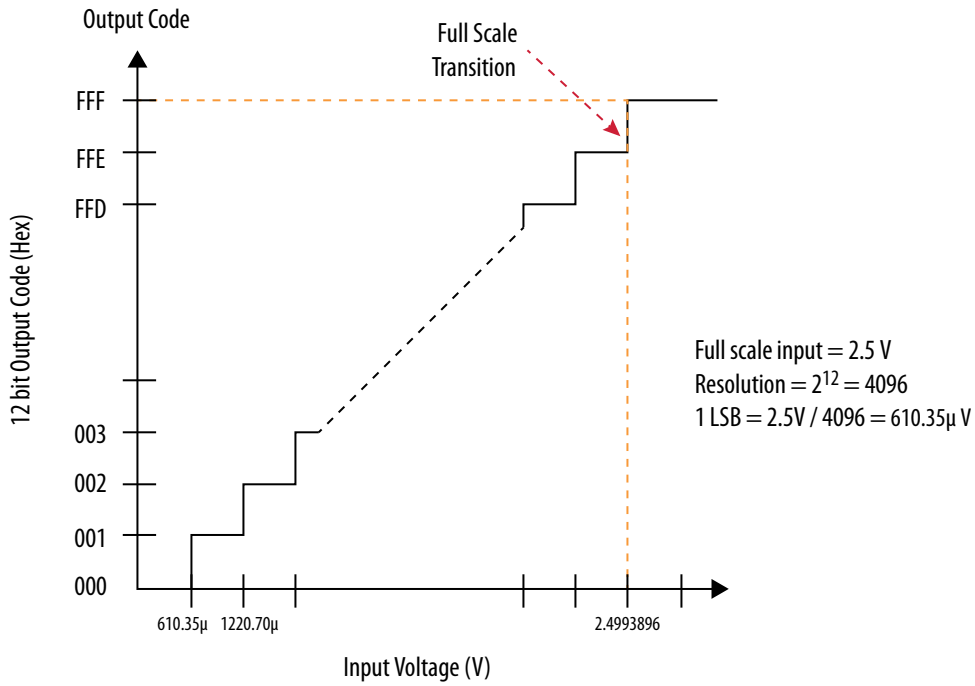
- Intel MAX 10 Device Datasheet
- Intel MAX 10 FPGA Device Overview

1.5 Intel MAX 10 ADC Conversion

The ADC in dual supply Intel MAX 10 devices can measure from 0 V to 2.5 V. In single supply Intel MAX 10 devices, it can measure up to 3.0 V or 3.3 V, depending on your power supply voltage.

- In prescaler mode, the analog input can measure up to 3.0 V in dual supply Intel MAX 10 devices and up to 3.6 V in single supply Intel MAX 10 devices.
- The analog input scale has full scale code from 000h to FFFh. However, the measurement can only display up to *full scale – 1 LSB*.
- For the 12 bits corresponding value calculation, use unipolar straight binary coding scheme.

Figure 2. ADC Measurement Display for 2.5 V



The Intel MAX 10 ADC is a 1 MHz successive approximation register (SAR) ADC. If you set up the PLL and Altera Modular ADC IP core correctly, the ADC operates at up to 1 MHz during normal sampling and 50 kHz during temperature sensing.

Note: The analog value represented by the all-ones code is not full scale but *full scale – 1 LSB*. This is a common convention in data conversion notation and applies to ADCs.

Related Links

- Creating Intel MAX 10 ADC Design on page 40
- Altera Modular ADC Parameters Settings on page 47
- Altera Modular Dual ADC Parameters Settings on page 51

1.5.1 Voltage Representation Conversion

Use the following equations to convert the voltage between analog value and digital representation.

Equation 1. Conversion from Analog Value to Digital Code

$$\text{Digital Code} = \left(\frac{V_{IN}}{V_{REF}}\right) \times 2^{12}$$

Equation 2. Conversion from Digital Code to Analog Value

$$\text{Analog Value} = \text{Digital Code} \times \left(\frac{V_{REF}}{2^{12}}\right)$$

Example 1. Calculation Example for V_{REF} of 2.5 V

Analog voltage value to digital code (in decimal), where signal in is 2 V:

$$\text{Digital Code} = \left(\frac{2}{2.5}\right) \times 4096 = 3277$$

Digital code to analog voltage value, approximation to 4 decimal points:

$$\text{Analog Value} = 3277 \times \left(\frac{2.5}{4096}\right) = 2.0000$$

2 Intel MAX 10 ADC Architecture and Features

In Intel MAX 10 devices, the ADC is a 12-bit SAR ADC that provides the following features:

- Sampling rate of up to 1 MSPS
- Up to 18 channels for analog measurement: 16 dual function channels and two dedicated analog input channels in dual ADC devices
- Single-ended measurement capability
- Simultaneous measurement capability at the dedicated analog input pins for dual ADC devices
- Soft logic sequencer
- On-chip temperature sensor with sampling rate of 50 kilosamples per second
- Internal or external voltage references usage. The source of the internal voltage reference is the ADC analog supply; the ADC conversion result is ratiometric.

Related Links

- Intel MAX 10 Analog to Digital Converter Overview on page 4
- Intel MAX 10 Analog to Digital Converter User Guide Archives on page 62
  Provides a list of user guides for previous versions of the Altera Modular ADC and Altera Modular Dual ADC IP cores.

2.1 Intel MAX 10 ADC Hard IP Block

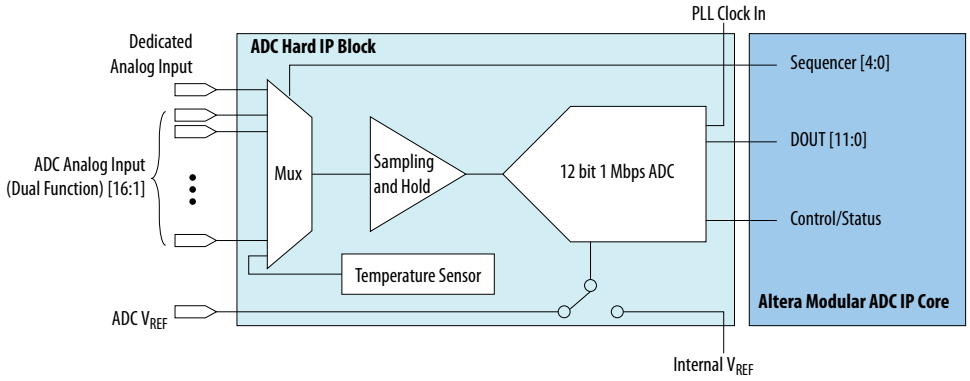
The Intel MAX 10 ADC is a successive approximation register (SAR) ADC that converts one analog sample in one clock cycle.

Each ADC block supports one dedicated analog input pin and up to 16 channels of dual function pins.

You can use the built-in temperature sensing diode (TSD) to perform on-chip temperature measurement.

Figure 3. ADC Hard IP Block in Intel MAX 10 Devices

Note: In dual ADC devices, the temperature sensor is available only in ADC1.



Related Links

Sequencer Core on page 27
Provides mode information about the sequencer conversion modes.

2.1.1 ADC Block Locations

The ADC blocks are located at the top left corner of the Intel MAX 10 device periphery.

Figure 4. ADC Block Location in Intel MAX 10 04 and 08 Devices

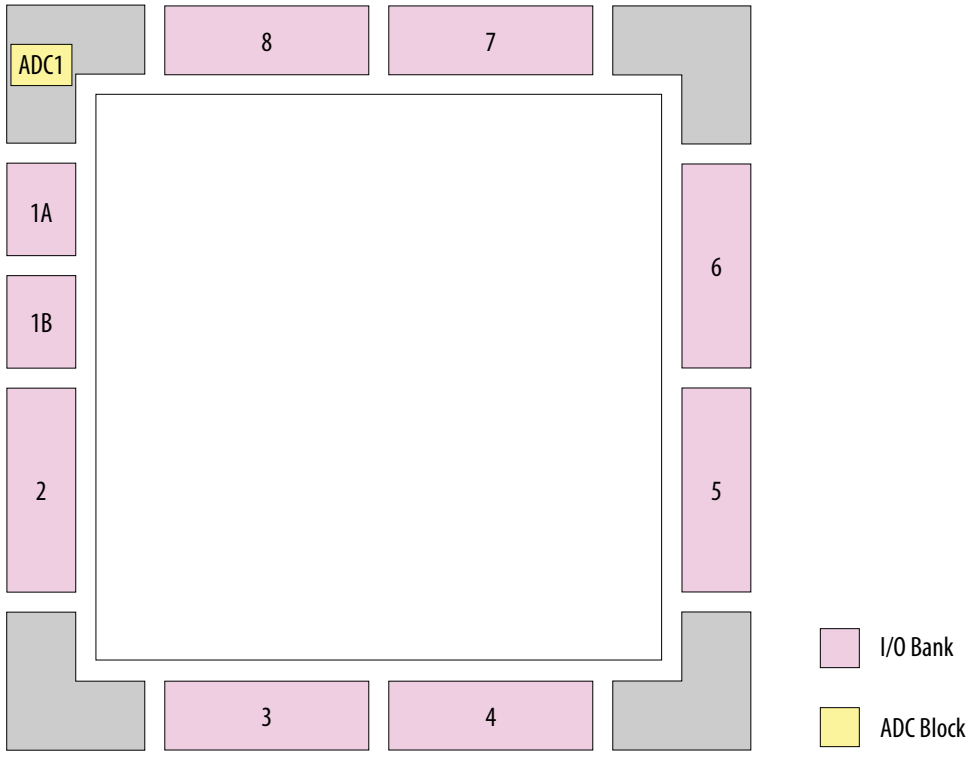


Figure 5. ADC Block Location in Intel MAX 10 16 Devices

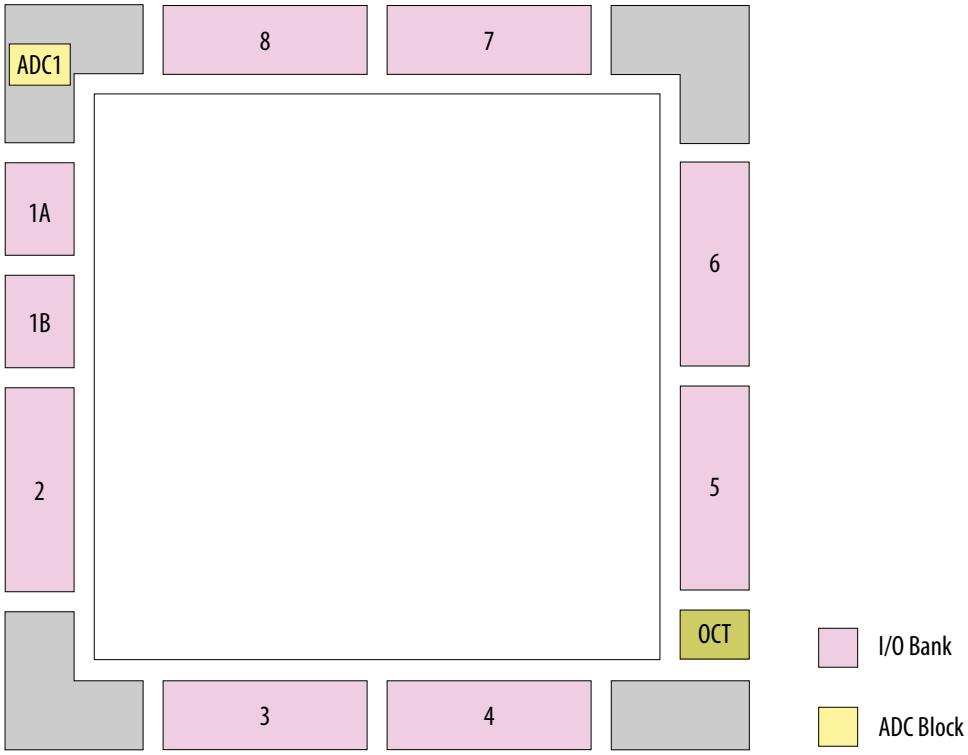
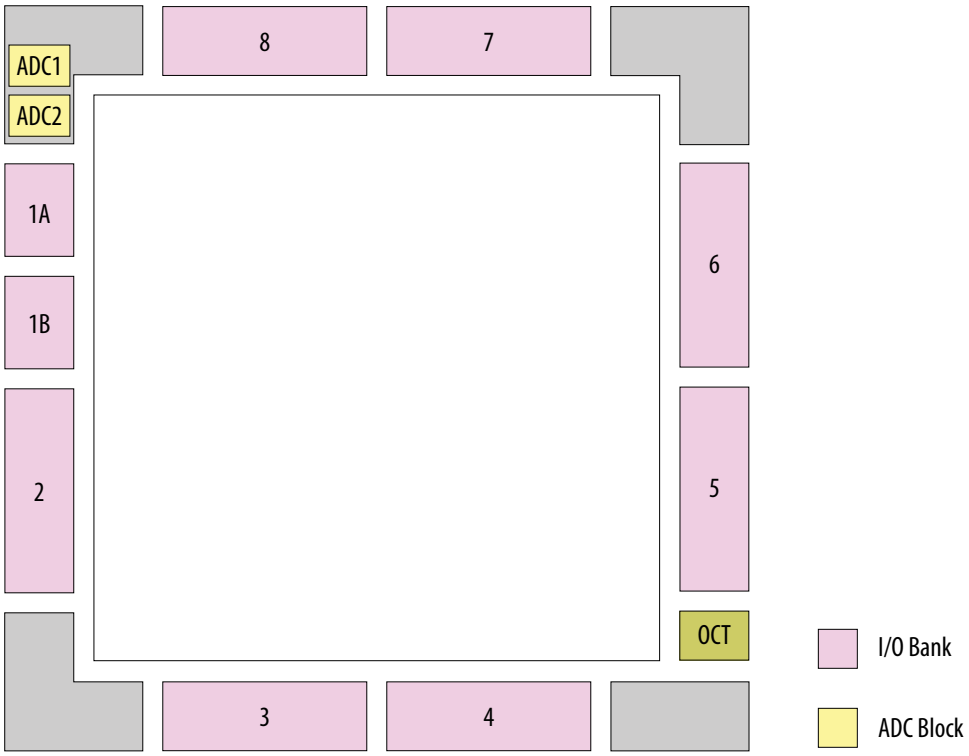


Figure 6. ADC Block Location in Intel MAX 10 25, 40, and 50 Devices

Package E144 of these devices have only one ADC block.



2.1.2 Single or Dual ADC Devices

Intel MAX 10 devices are available with single or dual ADC blocks.

For devices with one ADC block, you can use up to 17 ADC channels:

- These channels include one dedicated analog input and up to 16 dual function pins.
- You can use the dual function pins as GPIO pins when you do not use the ADC.

Note: Intel MAX 10 devices in the E144 package have only 8 dual function ADC pins.

For devices with two ADC blocks, you can use up to 18 ADC channels:

- For dual ADC devices, each ADC block can support one dedicated analog input pin and up to 8 dual function pins.
- If you use both ADC blocks in dual ADC devices, you can use up to two dedicated analog input pins and 16 dual function pins.
- For simultaneous measurement, you can use only dedicated analog input pins in both ADC blocks because the package routing of both dedicated analog pins are matched. For dual function pins, the routing latency between two ADC blocks may cause data mismatch in simultaneous measurement.
- For simultaneous measurement, use the Altera Modular Dual ADC IP core.

To choose the correct device, refer to the Intel MAX 10 device overview.

Related Links

- Intel MAX 10 FPGA Device Overview
- ADC Channel Counts in Intel MAX 10 Devices on page 6

2.1.3 ADC Analog Input Pins

The analog input pins support single-ended and unipolar measurements.

The ADC block in Intel MAX 10 devices contains two types of ADC analog input pins:

- Dedicated ADC analog input pin—pins with dedicated routing that ensures both dedicated analog input pins in a dual ADC device has the same trace length.
- Dual function ADC analog input pin—pins that share the pad with GPIO pins.

If you use bank 1A for ADC, you cannot use the bank for GPIO.

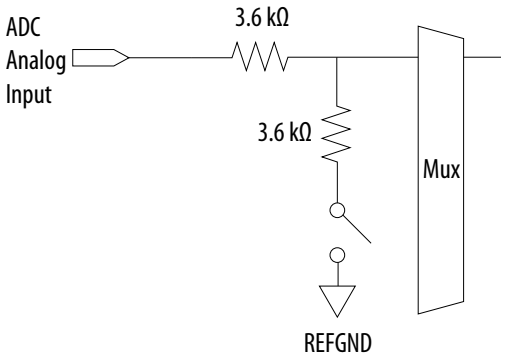
Each analog input pin in the ADC block is protected by electrostatic discharge (ESD) cell.

2.1.4 ADC Prescaler

The ADC block in Intel MAX 10 devices contains a prescaler function.

The prescaler function divides the analog input voltage by half. Using this function, you can measure analog input greater than 2.5 V. In prescaler mode, the analog input can handle up to 3 V input for the dual supply Intel MAX 10 devices and 3.6 V for the single supply Intel MAX 10 devices.

Figure 7. ADC Prescaler Block Diagram



The prescaler feature is available on these channels in each ADC block:

- Single ADC device—channels 8 and 16 (if available)
- Dual ADC device:
  - Using Altera Modular ADC IP core—channel 8 of first or second ADC
  - Using Altera Modular Dual ADC IP core—channel 8 of ADC1 and channel 17 of ADC2

2.1.5 ADC Clock Sources

The ADC block uses the device PLL as the clock source. The ADC clock path is a dedicated clock path. You cannot change this clock path.

Depending on the device package, the Intel MAX 10 devices support one or two PLLs—PLL1 only, or PLL1 and PLL3.

For devices that support two PLLs, you can select which PLL to connect to the ADC. You can configure the ADC blocks with one of the following schemes:

- Both ADC blocks share the same clock source for synchronization.
- Both ADC blocks use different PLLs for redundancy.

If each ADC block in your design uses its own PLL, the Intel Quartus Prime Fitter automatically selects the clock source scheme based on the PLL clock input source:

- If each PLL that clocks its respective ADC block uses different PLL input clock source, the Intel Quartus Prime Fitter follows your design (two PLLs).
- If both PLLs that clock their respective ADC block uses the same PLL input clock source, the Intel Quartus Prime Fitter merges both PLLs as one.

In dual ADC mode, both ADC instance must share the same ADC clock setting.

Related Links

PLL Locations, Intel MAX 10 Clocking and PLL User Guide
Provides more information about the availability of PLL3 in different Intel MAX 10 devices and packages.

2.1.6 ADC Voltage Reference

Each ADC block in Intel MAX 10 devices can independently use an internal or external voltage reference. In dual ADC devices, you can assign an internal voltage reference to one ADC block and an external voltage reference to the other ADC block.

There is only one external VREF pin in each Intel MAX 10 device. Therefore, if you want to assign external voltage reference for both ADC blocks in dual ADC devices, share the same external voltage reference for both ADC blocks.

Intel recommends that you use a clean external voltage reference with a maximum resistance of 100 Ω for the ADC blocks. If the ADC block uses an internal voltage reference, the ADC block is tied to its analog voltage and the conversion result is ratiometric.

2.1.7 ADC Temperature Sensing Diode

The ADC block in Intel MAX 10 devices has built-in TSD. You can use the built-in TSD to monitor the internal temperature of the Intel MAX 10 device.

- While using the temperature sensing mode, the ADC sampling rate is 50 kilosamples per second during temperature measurement.
- After the temperature measurement completes, if the next conversion in the sequence is normal sampling mode, the Altera Modular ADC IP core automatically switches the ADC back to normal sampling mode. The maximum cumulative sampling rate in normal sampling mode is 1 MSPS.
- When the ADC switches from normal sensing mode to temperature sensing mode, and vice versa, calibration is run automatically for the changed clock frequency. The calibration incurs at least six clock calibration cycles from the new sampling rate.
- The ADC TSD measurement uses a 64-samples running average method. For example:
  - The first measured temperature value is the average of samples 1 to 64.
  - The second measured temperature value is the average of samples 2 to 65.
  - The third measured temperature value is the average of samples 3 to 66.
  - The subsequent temperature measurements follow the same method.

For dual ADC devices, the temperature sensor is available in ADC1 only.

2.1.7.1 Temperature Measurement Code Conversion

Use the temperature measurement code conversion table to convert the values measured by the ADC TSD to actual temperature.

Table 4. Temperature Code Conversion Table

Temp (C)	Code	Temp (C)	Code	Temp (C)	Code	Temp (C)	Code	Temp (C)	Code
-40	3798	-6	3738	28	3670	62	3593	96	3510
-39	3796	-5	3736	29	3667	63	3592	97	3507
-38	3795	-4	3733	30	3666	64	3591	98	3504
-37	3793	-3	3732	31	3664	65	3590	99	3501
-36	3792	-2	3731	32	3662	66	3589	100	3500
-35	3790	-1	3730	33	3660	67	3585	101	3498
-34	3788	0	3727	34	3658	68	3582	102	3496
-33	3786	1	3725	35	3656	69	3579	103	3494
-32	3785	2	3721	36	3654	70	3576	104	3492
-31	3782	3	3720	37	3651	71	3573	105	3490
-30	3781	4	3719	38	3648	72	3570	106	3489
-29	3780	5	3717	39	3645	73	3567	107	3486
-28	3779	6	3715	40	3643	74	3564	108	3483
-27	3777	7	3713	41	3642	75	3561	109	3480
-26	3775	8	3711	42	3641	76	3558	110	3477
-25	3773	9	3709	43	3640	77	3555	111	3474
-24	3771	10	3707	44	3638	78	3552	112	3471
-23	3770	11	3704	45	3636	79	3551	113	3468
-22	3768	12	3703	46	3634	80	3550	114	3465
-21	3766	13	3702	47	3632	81	3549	115	3461
-20	3765	14	3700	48	3630	82	3548	116	3460
-19	3764	15	3699	49	3628	83	3547	117	3459
-18	3762	16	3698	50	3625	84	3546	118	3456
-17	3759	17	3697	51	3622	85	3542	119	3451
-16	3756	18	3696	52	3619	86	3538	120	3450
-15	3754	19	3695	53	3616	87	3534	121	3449
-14	3752	20	3688	54	3613	88	3530	122	3445
-13	3751	21	3684	55	3610	89	3526	123	3440
-12	3750	22	3682	56	3607	90	3525	124	3432
-11	3748	23	3680	57	3604	91	3524	125	3431
-10	3746	24	3678	58	3601	92	3522	—	—

continued...

Temp (C)	Code	Temp (C)	Code	Temp (C)	Code	Temp (C)	Code	Temp (C)	Code
-9	3744	25	3677	59	3598	93	3519	—	—
-8	3742	26	3676	60	3595	94	3516	—	—
-7	3740	27	3673	61	3594	95	3513	—	—

2.1.7.2 Temperature Measurement Sampling Rate

In temperature sensing mode, the maximum ADC sampling rate is 50 kilosamples per second (50 KHz frequency). The sampling rate of the TSD depends on the **ADC Sample Rate** parameter you selected in the Altera Modular ADC or Altera Modular Dual ADC IP core.

Table 5. Intel MAX 10 TSD Sampling Rate Based on Selected ADC Sample Rate Parameter

ADC Sample Rate Selected	Actual TSD Sampling Rate
1 MHz	50 KHz
500 KHz	50 KHz
250 KHz	25 KHz
200 KHz	20 KHz
125 KHz	12.5 KHz
100 KHz	10 KHz
50 KHz	5 KHz
25 KHz	2.5 KHz

2.1.8 ADC Sequencer

The Altera Modular ADC and Altera Modular Dual ADC IP cores implement the sequencer. Use the Altera Modular ADC or Altera Modular Dual ADC parameter editor to define the ADC channel acquisition sequence and generate the HDL code.

The sequencer can support sequences of up to 64 ADC measurement slots. While configuring the Altera Modular ADC or Altera Modular Dual ADC IP core, you can select which channel, including the TSD channel, to sample in each sequencer slot. During runtime, you cannot change the channel sequence but you can configure the sequencer conversion mode using the Nios® II HAL driver API.

You can specify up to 64 slots and assign the channel for each slot. You can repeat the same channel number several times if required.

Related Links

Guidelines: ADC Sequencer in Altera Modular Dual ADC IP Core on page 19

2.1.8.1 Guidelines: ADC Sequencer in Altera Modular Dual ADC IP Core

Follow these sequencer guidelines if you use dual ADC blocks with the Altera Modular Dual ADC IP core.

- The conversion sequence length of both ADC blocks must be the same.
- You can configure independent patterns for the conversion sequence of each ADC blocks.
- You can set a sequencer slot in ADC2 to NULL. If you set the slot to NULL, ADC2 will perform a dummy conversion for the slot with output of "0". The NULL option is available only for ADC2.
- The temperature sensor is available only in ADC1. If you configure a sequencer slot in ADC1 for temperature sensing, you must set the same sequencer slot number in ADC2 to NULL.

Related Links

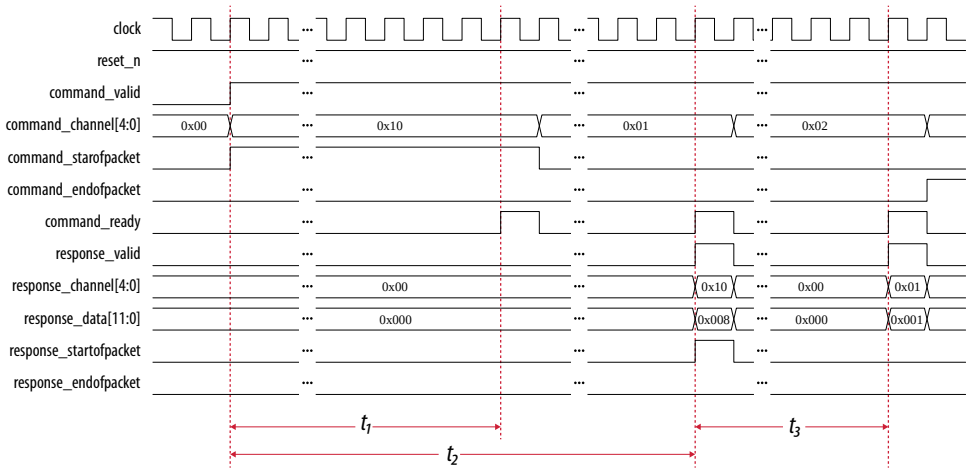
ADC Sequencer on page 19

2.1.9 ADC Timing

Figure 8. Intel MAX 10 ADC Control Core Timing Diagram

The following figure shows the timing diagram for the command and response interface of the Altera Modular ADC control core. The timing diagram shows the latency of the first valid response data, and the latency between the first acknowledgment of the first command request and the back-to-back response data. The diagram shows an example where:

- The conversion sequence is from channel 16 to channel 1 to channel 2
- The response data for channel 16 is 8
- The response data for channel 1 is 1



Mark	Time Period
t_1	$3 \times \text{ADC soft IP clock} + \frac{2}{\text{Sampling rate}}$
t_2	$3 \times \text{ADC soft IP clock} + \frac{2}{\text{Sampling rate}} + \frac{1}{\text{Sampling rate}}$
t_3	$\frac{1}{\text{Sampling rate}}$

Example Calculation			
Sampling Rate Setting	**t_1**	**t_2**	**t_3**
1 Msa/s	3 ×ADC soft IP clock + 2 µs	3 ×ADC soft IP clock + 3 µs	1 µs
500 ksa/s	3 ×ADC soft IP clock + 4 µs	3 ×ADC soft IP clock + 6 µs	2 µs

2.2 Altera Modular ADC and Altera Modular Dual ADC IP Cores

You can use the Altera Modular ADC and Altera Modular Dual ADC IP cores to generate soft IP controllers for the ADC hard IP blocks in Intel MAX 10 devices.

There are two ADC IP cores:

- Altera Modular ADC IP core—each instance can control one ADC hard IP block. In a dual ADC device, you can instantiate one Altera Modular ADC IP core instance for each ADC block. However, both instances will be asynchronous to each other.
- Altera Modular Dual ADC IP core—you can control both ADC hard IP block with a single IP instance.
  - For the analog input pins (ANAIN1 and ANAIN2) in both ADC hard IP blocks, the measurement is synchronous.
  - For the dual function input pins, there are some measurement timing differences caused by the routing latency.

You can perform the following functions with the Altera Modular ADC or Altera Modular Dual ADC IP core parameter editor:

- Configure the ADC clock, sampling rate, and reference voltage.
- Select which analog input channels that the ADC block samples.
- Configure the threshold value to trigger a threshold violation notification.
- Set up a conversion sequence to determine which channel requires more frequent attention.

Related Links

- Altera Modular ADC and Altera Modular Dual ADC IP Cores References on page 46
- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- Creating Version-Independent IP and Qsys Simulation Scripts
  Create simulation scripts that do not require manual updates for software or IP version upgrades.
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

2.2.1 Altera Modular ADC IP Core Configuration Variants

The Altera Modular ADC IP core provides four configuration variants that target different ADC use cases. These configuration variants support usages from basic system monitoring to high performance ADC data streaming.

Configuration 1: Standard Sequencer with Avalon-MM Sample Storage on page 22
Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection on page 23
Configuration 3: Standard Sequencer with External Sample Storage on page 24
Configuration 4: ADC Control Core Only on page 24

Related Links

Altera Modular ADC and Altera Modular Dual ADC IP Cores References on page 46

2.2.1.1 Configuration 1: Standard Sequencer with Avalon-MM Sample Storage

In this configuration variant, you can use the standard sequencer micro core with internal on-chip RAM for storing ADC samples. This configuration is useful for basic system monitoring application.

In a system monitoring application, the ADC captures a block of samples data and stores them in the on-chip RAM. The host processor retrieves the data before triggering another block of ADC data sample request. The speed of the host processor in servicing the interrupt determines the interval between each block sample request.

Figure 9. Standard Sequencer with Avalon-MM Sample Storage (Altera Modular ADC IP Core)

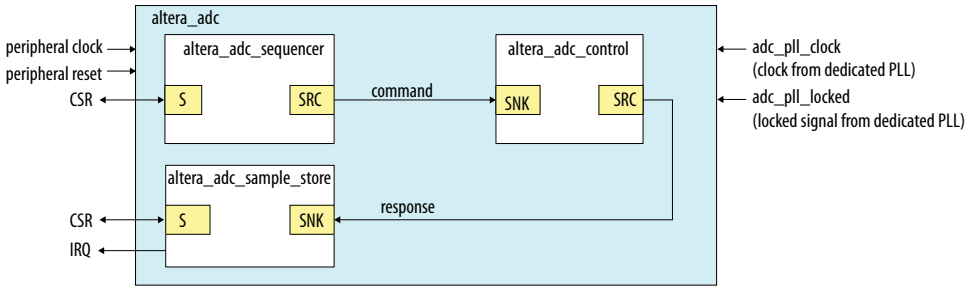
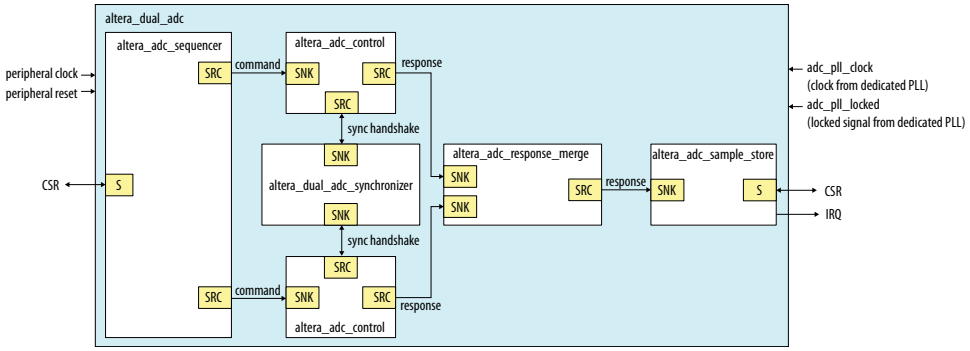


Figure 10. Standard Sequencer with Avalon-MM Sample Storage (Altera Modular Dual ADC IP Core)



Related Links

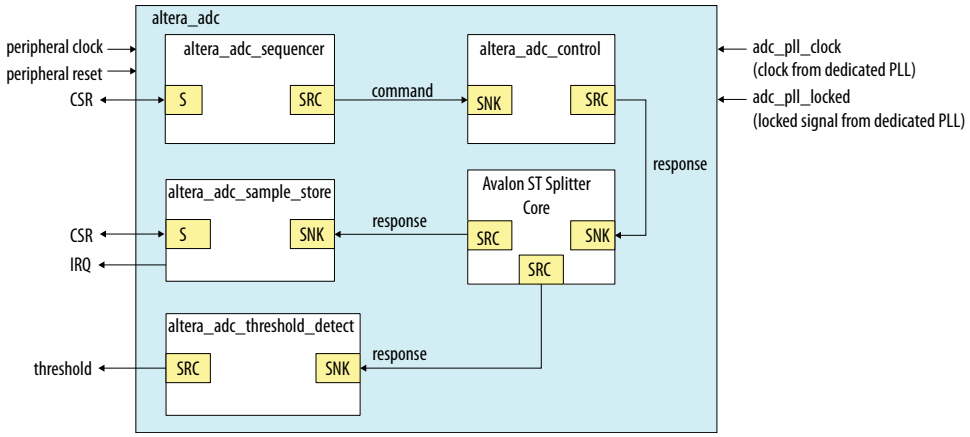
- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45

2.2.1.2 Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection

In this configuration variant, you can use the standard sequencer micro core with internal on-chip RAM for storing ADC samples with the additional capability of detecting threshold violation. This configuration is useful for system monitoring application where you want to know whether the ADC samples value fall outside the maximum or minimum threshold value.

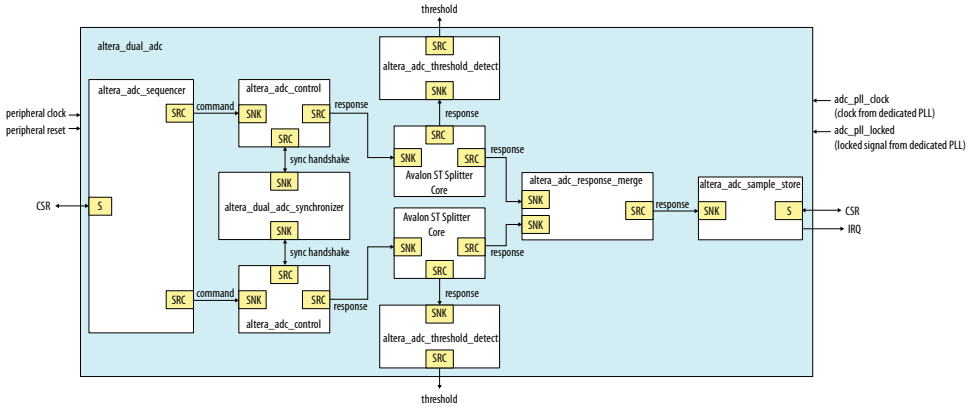
When the threshold value is violated, the Altera Modular ADC or Altera Modular Dual ADC IP core notifies the discrete logic component. The discrete component then triggers system recovery action. For example, the system can increase the fan speed in a temperature control system.

Figure 11. Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection (Altera Modular ADC IP Core)



In dual ADC mode, you can configure the threshold detection of each ADC instance independently of each other. This capability is available because each ADC instance measures different analog metrics.

Figure 12. Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection (Altera Modular Dual ADC IP Core)



Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45

2.2.1.3 Configuration 3: Standard Sequencer with External Sample Storage

In this configuration variant, you can use the standard sequencer micro core and store the ADC samples in external storage.

You need to design your own logic to interface with the external storage.

Figure 13. Standard Sequencer with External Sample Storage (Altera Modular ADC IP Core)

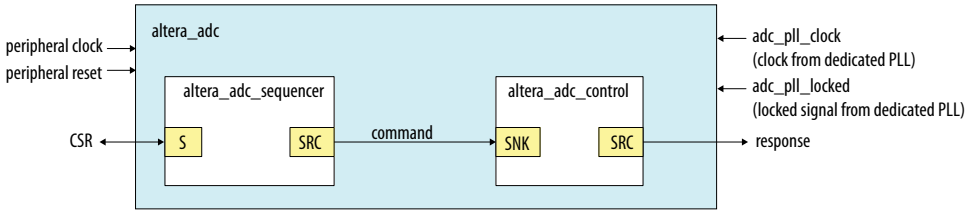
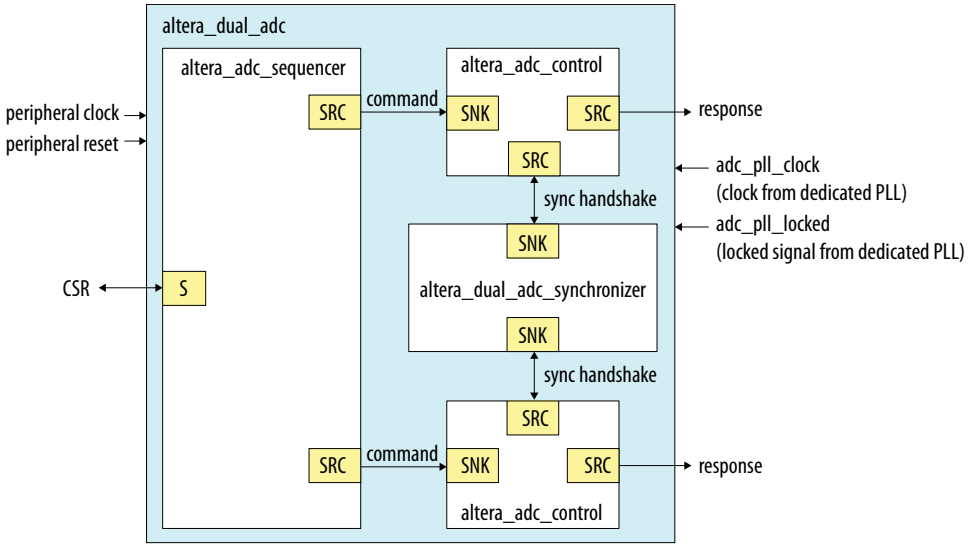


Figure 14. Standard Sequencer with External Sample Storage (Altera Modular Dual ADC IP Core)



Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45

2.2.1.4 Configuration 4: ADC Control Core Only

In this configuration variant, the Altera Modular ADC generates only the ADC control core. You have full flexibility to design your own application-specific sequencer and use your own way to manage the ADC samples.

Figure 15. ADC Control Core Only (Altera Modular ADC IP Core)

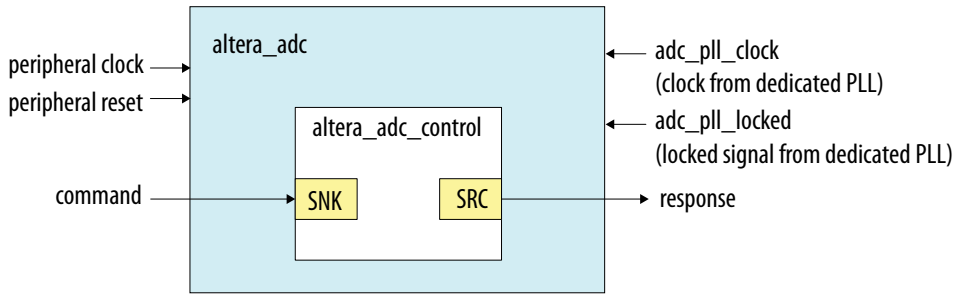
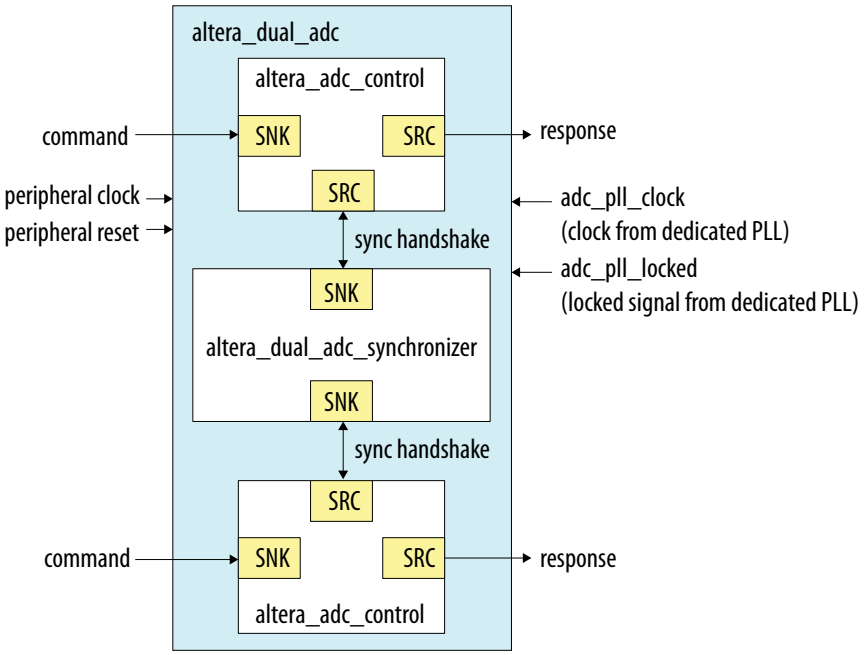


Figure 16. ADC Control Core Only (Altera Modular Dual ADC IP Core)



Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45

2.2.2 Altera Modular ADC and Altera Modular Dual ADC IP Cores Architecture

The Altera Modular ADC IP core consists of six micro cores.

Table 6. Altera Modular ADC Micro Cores

Micro Core	Description
ADC control	This core interacts with the ADC hard IP block. The ADC control core uses Avalon ST interface to receive commands from upstream cores, decodes, and drives the ADC hard IP block accordingly.
Sequencer	This core contains command register and static conversion sequence data. The sequencer core issues commands for downstream cores to execute. • You can use the command register to configure the intended conversion mode. • You can configure the length and content of the conversion sequence data only when generating the IP core. • You can access the register of the sequencer core through the Avalon-MM slave interface. • The command information to the downstream core goes through the Avalon ST interface.
Sample storage	This core stores the ADC samples that are received through the Avalon ST interface. • The samples are stored in the on-chip RAM. You can retrieve the samples through the Avalon-MM slave interface. • With this core, you have the option to generate interrupt when the ADC receives a block of ADC samples (one full round of conversion sequence).
Response merge	This core merges simultaneous responses from two ADC control cores into a single response packet to send to the sample storage core. This core is available only if you use the Altera Modular Dual ADC IP core in the following configurations: • Standard Sequencer with Avalon-MM Sample Storage • Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection
Dual ADC synchronizer core	This core performs synchronization handshakes between two ADC control cores. This core is available only if you use the Altera Modular Dual ADC IP core.
Threshold detection	• This core supports fault detection. The threshold detection core receives ADC samples through the Avalon ST interface and checks whether the samples value exceeds the maximum or falls below the minimum threshold value. • The threshold detection core conveys threshold value violation information through the Avalon ST interface. • You can configure which channel to enable for maximum and minimum threshold detection and the threshold values only during IP core generation.

2.2.2.1 ADC Control Core

The ADC control core drives the ADC hard IP according to the command it receives. The control core also maps the channels from the Altera Modular ADC IP core to the channels in the ADC hard IP block.

The ADC control core of the Altera Modular ADC IP core implements only the functions that are related to ADC hard IP block operations. For example:

- Power up
- Power down
- Analog to digital conversion on analog pins
- Analog to digital conversion on on-chip temperature sensor

The ADC control core has two clock domains:

- One clock domain for clocking the ADC control core soft logic
- Another clock domain for the ADC hard IP block

The ADC control core does not have run-time configurable options.

Figure 17. ADC Control Core High-Level Block Diagram

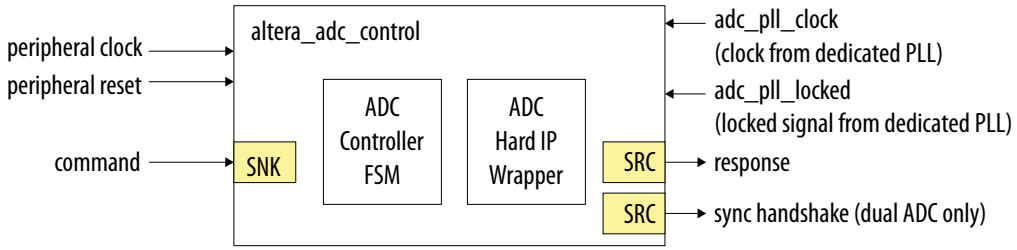


Table 7. ADC Control Core Backpressure Behavior

Interface	Backpressure Behavior
Command	The ADC control core asserts ready when it is ready to perform a sample conversion. The ADC control core only accepts one command at a time. The control core releases ready when it completes processing current command and prepares to perform the next command. Once the ADC control core asserts "cmd_ready=1" to acknowledge the current command, the Sequencer core provides the next valid request within two clock cycles. If the next valid request comes after two clock cycles, the ADC control core perform non-continuous sampling.
Response	The ADC control core does not support backpressure in the response interface. The fastest back-to-back assertion of valid request is 1 µs.

2.2.2.2 Sequencer Core

The sequencer core controls the type of conversion sequence performed by the ADC hard IP. You can configure the conversion mode during run time using the sequencer core registers.

During Altera Modular ADC or Altera Modular Dual ADC IP core configuration, the sequencer core provides up to 64 configurable slots. You can define the sequence that the ADC channels are sampled by selecting the ADC channel for each sequencer slot.

The sequencer core has a single clock domain.

Figure 18. Sequencer Core High-Level Block Diagram

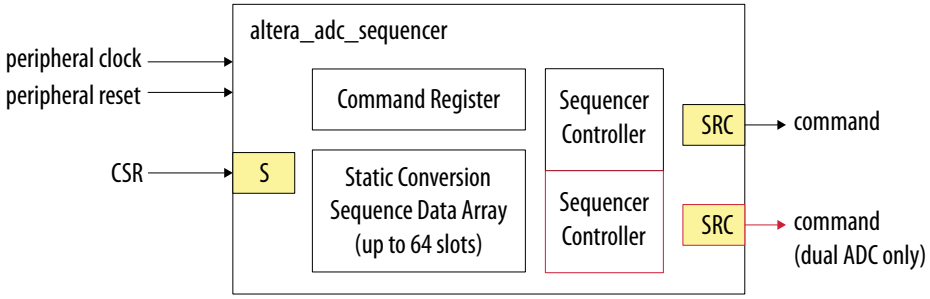


Table 8. Sequencer Core Conversion Modes

Conversion Mode	Description
Single cycle ADC conversion	• In this mode, when the run bit is set, ADC conversion starts from the channel that you specify in the first slot. • The conversion continues onwards with the channel that you specify in each sequencer slot. • Once the conversion finishes with the last sequencer slot, the conversion cycle stops and the ADC hard IP block clears the run bit.
Continuous ADC conversion	• In this mode, when the run bit is set, ADC conversion starts from the channel that you specify in the first slot. • The conversion continues onwards with the channel that you specify in each sequencer slot. • Once the conversion finishes with the last sequencer slot, the conversion begins again from the first slot of the sequence. • To stop the continuous conversion, clear the run bit. The sequencer core continues the conversion sequence until it reaches the last slot and then stops the conversion cycle.

Related Links

- Altera Modular ADC Parameters Settings on page 47
  Lists the parameters available during Altera Modular ADC IP core configuration.
- Altera Modular Dual ADC Parameters Settings on page 51
  Lists the parameters available during Altera Modular Dual ADC IP core configuration.
- Sequencer Core Registers on page 59
  Lists the registers for run-time control of the sequencer core.

2.2.2.3 Sample Storage Core

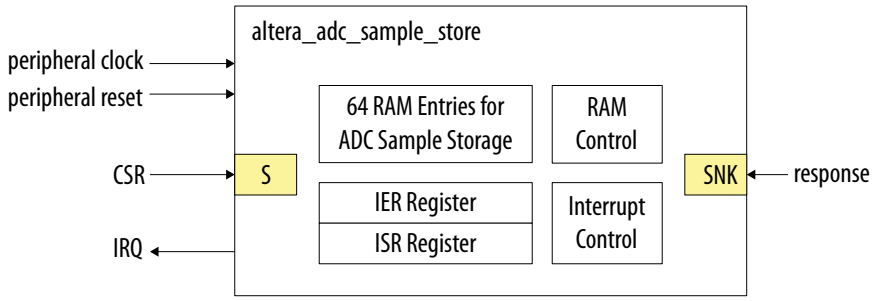
The sample storage core stores the ADC sampling data in the on-chip RAM. The sample storage core stores the ADC samples data based on conversion sequence slots instead of ADC channels.

For example, if you sample a sequence of CH1, CH2, CH1, CH3, CH1, and then CH4, the ADC sample storage core stores the channel sample data in the same RAM entry sequence. This means that CH1 sample data will be in the first, third, and fifth RAM entries; one for each sequence slot.

The sample storage core asserts IRQ when it completes receipt of a sample block. You can disable the IRQ assertion during run time using the interrupt enable register (IER) of the sample storage core. If you disable IRQ assertion, you must create polling methods in your design to determine the complete receipt of a sample block.

The sample storage core has a single clock domain.

Figure 19. Sample Storage Core High-Level Block Diagram



Related Links

Sample Storage Core Registers on page 60

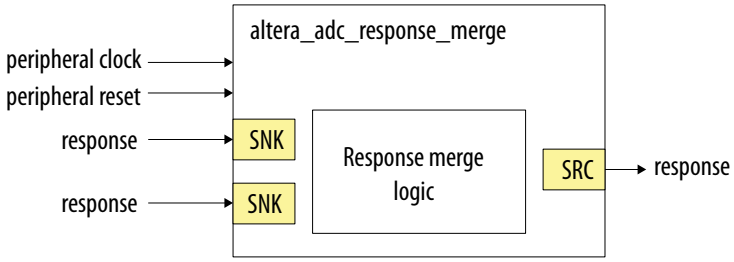
2.2.2.4 Response Merge Core

The response merge core merges simultaneous responses from two ADC control cores in the Altera Modular Dual ADC IP core.

The Altera Modular Dual ADC IP core uses the response merge core if you use the following configurations:

- Standard Sequencer with Avalon-MM Sample Storage
- Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection

Figure 20. Response Merge Core High-Level Block Diagram



2.2.2.5 Dual ADC Synchronizer Core

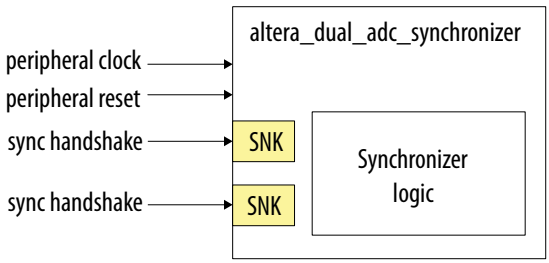
The dual ADC synchronizer core performs synchronization handshakes between two ADC control cores in the Altera Modular Dual ADC IP core.

The peripheral clock domain is asynchronous to the ADC PLL clock domain in the ADC control core. Control event from the ADC hard IP block can appear at the peripheral clock domain at the same time, or by a difference of one peripheral clock between ADC1 and ADC2 control cores. Both ADC hard IP cores communicate with the dual ADC synchronizer core through the Avalon-ST interface.

For example, although a new command valid event from the sequencer arrives at both ADC control cores at the same peripheral clock cycle, the end of conversion signals arrive at one peripheral clock cycle difference between ADC1 and ADC2. To avoid the condition where ADC1 begins conversion earlier or later than ADC2, the ADC control core performs synchronization handshake using the dual ADC synchronizer core.

An ADC control core asserts a `sync_valid` signal when it detects an ADC PLL clock domain event. The dual ADC synchronizer core asserts the `sync_ready` signal after it receives `sync_valid` signals from both ADC control cores. After the `sync_ready` signal is asserted, both ADC control cores proceed to their next internal state.

Figure 21. Dual ADC Synchronizer Core High-Level Block Diagram



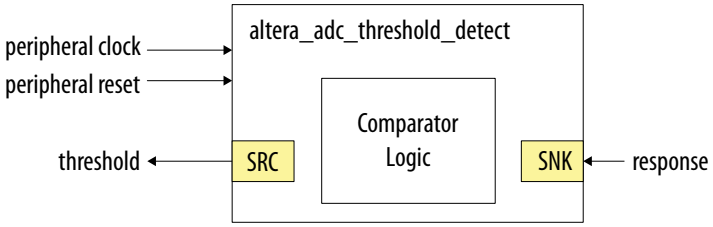
2.2.2.6 Threshold Detection Core

The threshold detection core compares the sample value that the ADC block receives to the threshold value that you define during Altera Modular ADC IP core configuration. This core does not have run-time configurable options.

If the ADC sample value is beyond the maximum or minimum threshold limit, the threshold detection core issues a violation notification through the Avalon-ST interface.

The threshold detection core has a single clock domain.

Figure 22. Threshold Detection Core High-Level Block Diagram



2.3 Intel FPGA ADC HAL Driver

The Intel FPGA ADC HAL driver supports the following features:

- Read ADC channel data.
- Enable maximum or minimum threshold and return a user callback when the interrupt is triggered.
- Command the control of the ADC (run, stop, and recalibrate).

Related Links

- HAL API Reference, Nios II Gen 2 Software Developer's Handbook
  Provides more information about the HAL API.
- ADC HAL Device Driver for Nios II Gen 2 on page 61

2.4 ADC Toolkit for Testing ADC Performance

You can use the ADC Toolkit provided with the Intel Quartus Prime software to understand the performance of the analog signal chain as seen by the Intel MAX 10 ADC blocks.

The ADC Toolkit supports monitoring the ADC whether you use the Altera Modular ADC or Altera Modular Dual ADC IP core. However, the ADC Toolkit can only monitor one ADC block at a time. If you are using the Altera Modular Dual ADC IP core, configure the **Debug Path** parameter in the IP core to select which ADC block you want to hook up to the ADC Toolkit.

Related Links

ADC Toolkit
Provides more information about the ADC Toolkit.

2.5 ADC Logic Simulation Output

By default, the ADC logic simulation outputs a fixed unique value for each ADC channel. However, you can enable an option to specify your own output values for each ADC channel other than the TSD.

The ADC simulation model for Intel MAX 10 devices supports the standard digital logic simulators that the Intel Quartus Prime software supports.

Related Links

Intel Quartus Prime Simulator Support

2.5.1 Fixed ADC Logic Simulation Output

By default, the **Enable user created expected output file** option in the Altera Modular ADC or Altera Modular Dual ADC IP core is disabled. The ADC simulation always output a fixed value for each ADC channel, including the analog and TSD channels. The values are different for single and dual ADC devices.

Table 9. Fixed Expected Output Data for Single ADC Device Simulation

Channel	Expected Output Data (Decimal Value)
CH0	0
CH1	1
CH2	2
CH3	3
CH4	4
CH5	5
CH6	6
CH7	7
CH8	8
CH9	9
CH10	10
CH11	11
CH12	12
CH13	13

continued...

Channel	Expected Output Data (Decimal Value)
CH14	14
CH15	15
CH16	16
TSD	3615

Table 10. Fixed Expected Output Data for Dual ADC Device Simulation

Channel	Expected Output Data (Decimal Value)	
	ADC1	ADC2
CH0	10	20
CH1	11	21
CH2	12	22
CH3	13	23
CH4	14	24
CH5	15	25
CH6	16	26
CH7	17	27
CH8	18	28
TSD	3615	— (No TSD in ADC2)

2.5.2 User-Specified ADC Logic Simulation Output

You can configure the Altera Modular ADC or Altera Modular Dual ADC IP core to output user-specified values in the logic simulation for each ADC channel except the TSD channel.

If you enable this feature, you must provide a simulation stimulus input file for each ADC channel that you enable. The logic simulation reads the input file for each channel and outputs the value of the current sequence. Once the simulation reaches the end of the file, it repeats from the beginning of the sequence.

The stimulus input file is a plain text file that contains two columns of numbers:

- The first column of numbers is ignored by the simulation model. You can use any values that you want such as time or sequence. The actual data sequencing is based on the text rows.
- The second column contains the voltage values.

The ADC IP core automatically converts each voltage value to a 12-bit digital value based on the reference voltage you specify in the IP core parameter settings.

Figure 23. Simulation Output Example, One Channel Enabled

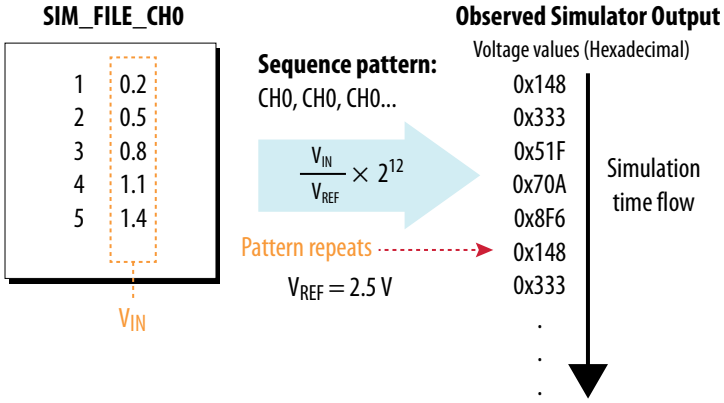
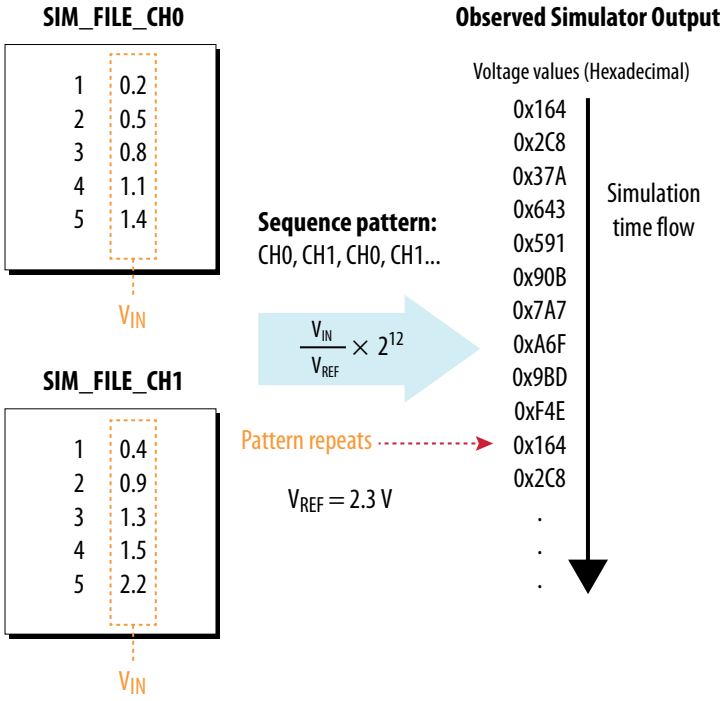


Figure 24. Simulation Output Example, Two Channels Enabled



3 Intel MAX 10 ADC Design Considerations

There are several considerations that require your attention to ensure the success of your designs. Unless noted otherwise, these design guidelines apply to all variants of this device family.

Related Links

Intel MAX 10 Analog to Digital Converter Overview on page 4

3.1 Guidelines: ADC Ground Plane Connection

For the ADC and V_{REF} pins, use the REFGND pin as the analog ground plane connection.

Related Links

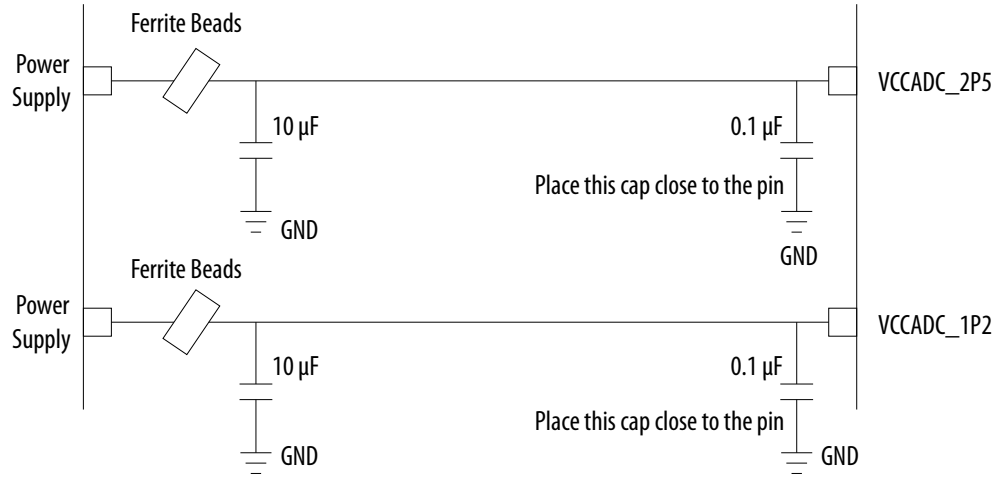
Intel MAX 10 FPGA Device Family Pin Connection Guidelines
Provides more information about pin connections including pin names and connection guidelines.

3.2 Guidelines: Board Design for Power Supply Pin and ADC Ground (REFGND)

The crosstalk requirement for analog to digital signal is -100 dB up to 2 GHz. There must be no parallel routing between power, ground, and surrounding general purpose I/O traces. If a power plane is not possible, route the power and ground traces as wide as possible.

- To reduce IR drop and switching noise, keep the impedance as low as possible for the ADC power and ground. The maximum DC resistance for power is 1.5 Ω.
- The power supplies connected to the ADC should have ferrite beads in series followed by a 10 µF capacitor to the ground. This setup ensures that no external noise goes into the device power supply pins.
- Decouple each of the device power supply pin with a 0.1 µF capacitor. Place the capacitor as close as possible to the device pin.

Figure 25. Recommended RC Filter for Power Traces



There is no impedance requirement for the REFGND. Intel recommends that you use the lowest impedance with the most minimum DC resistance possible. Typical resistance is less than 1 Ω.

Intel recommends that you set a REFGND plane that extends as close as possible to the corresponding decoupling capacitor and FPGA:

- If possible, define a complete REFGND plane in the layout.
- Otherwise, route the REFGND using a trace that is as wide as possible from the island to the FPGA pins and decoupling capacitor.
- The REFGND ground is the analog ground plane for the ADC V_{REF} and analog input.
- Connect REFGND ground to the system digital ground through ferrite beads. You can also evaluate the ferrite bead option by comparing the impedance with the frequency specifications.

3.3 Guidelines: Board Design for Analog Input

The crosstalk requirement for analog to digital signal is -100 dB up to 2 GHz. There must be no parallel routing between analog input signals and I/O traces, and between analog input signals and FPGA I/O signal traces.

- The ADC presents a switch capacitor load to the driving circuit. Therefore, the total RC constant, including package, trace, and parasitic driver must be less than 42.4 ns. This consideration is to ensure that the input signal is fully settled during the sampling phase.
- If you reduce the total sampling rate, you can calculate the required settling time as $0.45 \div F_S > 10.62 \times RC\ constant$.
- To gain more total RC margin, Intel recommends that you make the driver source impedance as low as possible:
  - For non-prescaler channel—less than 1 kΩ
  - For prescaler channel—less than 11 Ω

  Note: Not adhering to the source impedance recommendation may impact parameters such as total harmonic distortion (THD), signal-to-noise and distortion ratio (SINAD), differential non-linearity (DNL), and integral non-linearity (INL).

Trace Routing

- If possible, route the switching I/O traces on different layers.
- There is no specific requirement for input signal trace impedance. However, the DC resistance for the input trace must be as low as possible.
- Route the analog input signal traces as adjacent as possible to REFGND if there is no REFGND plane.
- Use REFGND as ground reference for the ADC input signal.
- For prescaler-enabled input signal, set the ground reference to REFGND. Performance degrades if the ground reference of prescaler-enabled input signal is set to common ground (GND).

Input Low Pass Filter Selection

- Intel recommends that you place a low pass filter to filter out high frequency noise being aliased back onto the input signal.
- Place the low pass filter as close as possible to the analog input signals.
- The cut off frequency depends on the analog input frequency. Intel recommends that the $F_{cutoff\ @\ -3dB}$ is at least two times the input frequency.
- You can download the ADC input SPICE model for ADC front end board design simulation from the Intel website.

Table 11. RC Constant and Filter Value

This table is an example of the method to quantify the RC constant and identify the RC filter value.

Total RC Constant = $(R_{DRIVER} + R_{BOARD} + R_{PACKAGE} + R_{FILTER}) \times (C_{DRIVER} + C_{BOARD} + C_{PACKAGE} + C_{FILTER} + C_{PIN})$

Driver		Board		Package		Pin Capacitance (pF)	RC Filter		$F_{cutoff\ @\ -3dB}$ (MHz)	Total RC Constant (ns)	Settling Time (ns)
R (Ω)	C (pF)	R (Ω)	C (pF)	R (Ω)	C (pF)		R (Ω)	C (pF)			
5	2	5	17	3	5	6	60	550	4.82	42.34	42.4
10	2	5	17	3	5	6	50	580	5.49	41.48	42.4

Figure 26. Passive Low Pass Filter Example

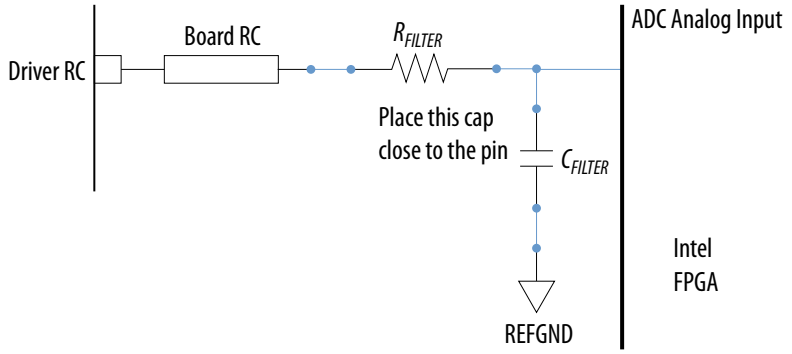
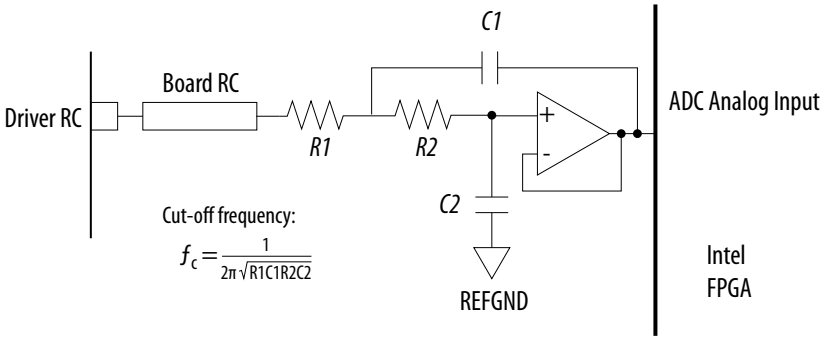


Figure 27. First Order Active Low Pass Filter Example

This figure is an example. You can design *n*th order active low pass filter.



Related Links

- Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core on page 42
- Altera Modular ADC Parameters Settings on page 47
- Altera Modular Dual ADC Parameters Settings on page 51
- SPICE Models for Intel FPGAs
  Provides the MAX 10 ADC spice model download.

3.4 Guidelines: Board Design for ADC Reference Voltage Pin

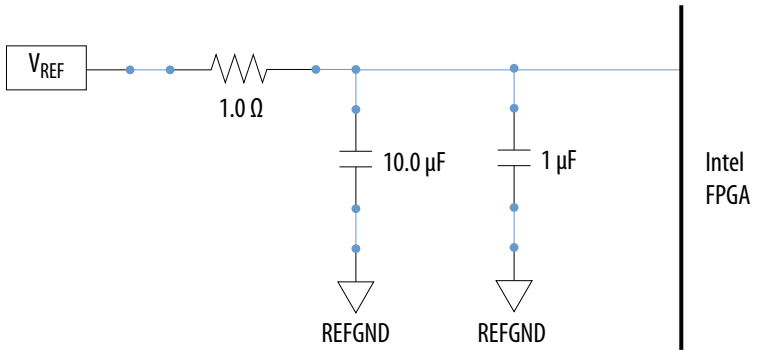
The crosstalk requirement for analog to digital signal is -100 dB up to 2 GHz. There is no parallel routing between analog input signals and I/O traces. Route the V_{REF} traces as adjacent as possible to REFGND.

If a REFGND plane is not possible, route the analog input signal as adjacent as possible to REFGND.

There is one ADC reference voltage pin in each Intel MAX 10 device. This pin uses REFGND as ground reference. Keep the trace resistance less than 0.8 Ω.

Figure 28. RC Filter Design Example for Reference Voltage Pin

Place the RC filter as close as possible to the analog input pin.

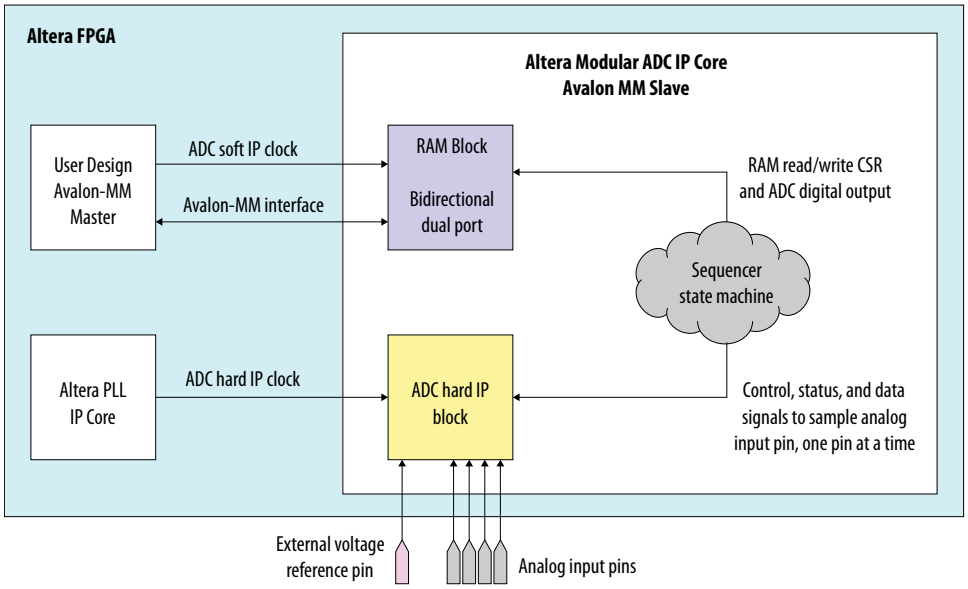


4 Intel MAX 10 ADC Implementation Guides

You can implement your ADC design in the Intel Quartus Prime software. The software contains tools for you to create and compile your design, and configure your device.

The Intel Quartus Prime software allows you to set up the parameters and generate your Altera Modular ADC IP core. To understand the ADC signal performance, you can use the Intel Quartus Prime ADC Toolkit. For more information about using the Intel Quartus Prime software and the ADC toolkit, refer to the related information.

Figure 29. High Level Block Diagram of the Intel MAX 10 ADC Solution



Related Links

- Intel MAX 10 Analog to Digital Converter Overview on page 4
- Intel Quartus Prime Standard Edition Handbook, Volume 1: Design and Synthesis
  Provides more information about using IP cores in the Quartus II software.
- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- Creating Version-Independent IP and Qsys Simulation Scripts
  Create simulation scripts that do not require manual updates for software or IP version upgrades.
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

- ADC Toolkit
  Provides more information about the ADC Toolkit.
- ADC Toolkit for Testing ADC Performance on page 30

4.1 Creating Intel MAX 10 ADC Design

To create your ADC design, you must customize and generate the ALTPLL and Altera Modular ADC IP cores.

The ALTPLL IP core provides the clock for the Altera Modular ADC IP core.

1. Customize and generate the ALTPLL IP core.
2. Customize and generate the Altera Modular ADC IP core.
3. Connect the ALTPLL IP core to the Altera Modular ADC IP core.
4. Create ADC Avalon slave interface to start the ADC.

Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core on page 42
- Parameters Settings for Generating ALTPLL IP Core on page 41
- Completing ADC Design on page 45
- Intel MAX 10 Getting Started
- Intel MAX 10 Online Training
- Intel MAX 10 How-to Videos
- How to Create ADC Design in Intel MAX 10 Device Using Platform Designer (Standard) Tool
  Provides video instruction that demonstrates how to create the ADC design in Intel MAX 10 devices using the Qsys system integration tool within the Intel Quartus Prime software and how to use the ADC toolkit to view the measured analog signal.
- How to Create Simultaneous Measurement with Intel MAX 10 ADC, Part 1
  Provides the first part of video instruction series that explains the differences between the Intel MAX 10 Altera Modular ADC and Altera Modular Dual ADC IP cores. The video also demonstrates how to create a simple simultaneous ADC measurement and how to place signal taps to measure the digital code output for analog signal.
- How to Create Simultaneous Measurement with Intel MAX 10 ADC, Part 2
  Provides the second part of video instruction series that explains the differences between the Intel MAX 10 Altera Modular ADC and Altera Modular Dual ADC IP cores. The video also demonstrates how to create a simple simultaneous ADC measurement and how to place signal taps to measure the digital code output for analog signal.

4.2 Customizing and Generating Altera Modular ADC IP Core

Intel recommends that you use the Altera Modular ADC IP core with a Nios II processor, which supports the ADC HAL driver.

1. Create a new project in the Intel Quartus Prime software.

   While creating the project, select a device that has one or two ADC blocks.
2. In the Intel Quartus Prime software, select **Tools ➤ Qsys**.
3. In the **Qsys** window, select **File ➤ New System**.
   A clock source block is automatically added under the **System Contents** tab.
4. In the **System Contents** tab, double click the clock name.
5. In the **Parameters** tab for the clock source, set the **Clock frequency**.
6. In the **IP Catalog** tab in the **Qsys** window, double click **Processors and Peripherals ➤ Peripherals ➤ Altera Modular ADC**.
   The Altera Modular ADC appears in the **System Contents** tab and the Altera Modular ADC parameter editor opens.
7. In the Altera Modular ADC parameter editor, specify the parameter settings and channel sampling sequence for your application.
8. In the **System Contents** tab in the **Qsys** window, double click the **Export** column of the `adc_pll_clock` and `adc_pll_locked` interfaces to export them.
9. Connect the `clock`, `reset_sink`, `sample_store_csr`, and `sample_store_irq` signals. Optionally, you can use the Nios II Processor, On-Chip Memory, and JTAG UART IP cores to form a working ADC system that uses the Intel FPGA ADC HAL drivers.
10. In the **Qsys** window, select **File ➤ Save**.

You can copy an example HDL code to declare an instance of your ADC system. In the **Qsys** window, select **Generate ➤ HDL Example**.

Related Links

- Creating Intel MAX 10 ADC Design on page 40
- Parameters Settings for Generating ALTPLL IP Core on page 41
- Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core on page 42
- Configuration 1: Standard Sequencer with Avalon-MM Sample Storage on page 22
- Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection on page 23
- Configuration 3: Standard Sequencer with External Sample Storage on page 24
- Configuration 4: ADC Control Core Only on page 24
- ADC PLL Clock Interface of Altera Modular ADC and Altera Modular Dual ADC on page 58
- ADC PLL Locked Interface of Altera Modular ADC and Altera Modular Dual ADC on page 59

4.3 Parameters Settings for Generating ALTPLL IP Core

Navigate through the ALTPLL IP core parameter editor and specify the settings required for your design. After you have specified all options as listed in the following table, you can generate the HDL files and the optional simulation files.

For more information about all ALTPLL parameters, refer to the related information.

Table 12. ALTPLL Parameters Settings

To generate the PLL for the ADC, use the following settings.

Tab	Parameter	Setting
Parameter Settings ➤ General/Modes	What is the frequency of the inclk0 input?	Specify the input frequency to the PLL.
Parameter Settings ➤ Inputs/Lock	Create an 'areset' input to asynchronously reset the PLL	Turn off this option.
	Create 'locked' output	Turn on this option. You need to connect this signal to the `adc_pll_locked` port of the Altera Modular ADC or Altera Modular Dual ADC IP core.
Output Clocks ➤ clk c0	Use this clock	Turn on this option.
	Enter output clock frequency	Specify an output frequency of 2, 10, 20, 40, or 80 MHz. You can specify any of these frequencies. The ADC block runs at 1 MHz internally but it contains a clock divider that can further divide the clock by a factor of 2, 10, 20, 40, and 80. Use this same frequency value in your Altera Modular ADC or Altera Modular Dual ADC IP core. You need to connect this signal to the `adc_pll_clock` port of the Altera Modular ADC or Altera Modular Dual ADC IP core. Different ADC sampling rates support different clock frequencies. For a valid sampling rate and clock frequency combination, refer to the related information.

Related Links

- Creating Intel MAX 10 ADC Design on page 40
- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45
- Intel MAX 10 Clock Networks and PLLs User Guide
- ADC PLL Clock Interface of Altera Modular ADC and Altera Modular Dual ADC on page 58
- ADC PLL Locked Interface of Altera Modular ADC and Altera Modular Dual ADC on page 59
- Valid ADC Sample Rate and Input Clock Combination on page 54

4.4 Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core

Navigate through the Altera Modular ADC IP core parameter editor and specify the settings required for your design. After you have specified all options as listed in the following tables, you can generate the HDL files and the optional simulation files.

Note: The Altera Modular ADC and Altera Modular Dual ADC IP cores support generating only Verilog* simulation scripts.

Intel recommends that you save the generated files in the design file directory (default setting).

For more information about each Altera Modular ADC or Altera Modular Dual ADC parameter, refer to the related information section.

Table 13. Parameter Settings in General Group

Parameter	Setting
Core Variant	There are four configuration variants of the Altera Modular ADC IP core. Select the core variant that meets your requirement. For more information, refer to the related information.
Debug Path	Turn this on to enable the debug path for the selected core variant. You can use the ADC Toolkit to monitor the ADC performance.
Generate IP for which ADCs of this device?	For devices with two ADC blocks, select the ADC block for which you are generating the IP core. There are feature differences between the two ADC blocks. The temperature sensor is available only in the first ADC block. There are also different channel counts in both ADC blocks.
ADC Sample Rate	Select the predefined sampling rate for the ADC from 25 kHz to 1 MHz. A lower sampling rate allows you greater flexibility in designing your ADC front end driver circuit. For example, using a lower sampling rate gives you a wider settling time margin for your filter design. The sampling rate you select affects which ADC input clock frequencies are available. Refer to the related information for more details about the sampling rate and the required settling time.
ADC Input Clock	Select the same frequency that you set for the ALTPLL IP core that clocks the Altera Modular ADC IP core. When configuring the ALTPLL IP core, specify a clock frequency that is supported by the ADC sampling rate. For more details, refer to the related information.
Reference Voltage Source	Select whether you want to use external or internal reference voltage. There is only one VREF pin. For dual ADC blocks, you can use one external V_{REF} source for both ADC blocks, or external V_{REF} for one ADC block and internal V_{REF} for the other ADC block.
External Reference Voltage	If you use external V_{REF} source in your design, specify the V_{REF} level.
Enable user created expected output file	If you want to use your own stimulus input file to simulate the ADC output data, enable this function and specify the file for the specific ADC channel. For more information about user-specified ADC logic simulation output, refer to the related information.

Table 14. Parameters Settings in Channels Group

You can navigate through the tabs for all the available channels and turn on the channel you want to use. In each channel (and **TSD**) tab, you can specify the settings in this table.

Parameter	Setting
Use Channel 0 (Dedicated analog input pin - ANAIN)	This option is available in the **CH0** tab. CH0 is the dedicated analog input channel. If you want to use the dedicated analog input, turn on this option.
User created expected output file	If you enabled the option to use your own stimulus input file to simulate the output data, click **Browse** and select the file for each enabled channel. This option is available in all channel tabs except the TSD tab.
Use Channel *N*	You can select which dual-function ADC channels to turn on or off. There are 16 channels (CH1 to CH16) for single ADC devices and 8 channels (CH1 to CH8) for each ADC block in dual ADC devices.
Use on-chip TSD	This option is available in the **TSD** tab. The TSD channel is the temperature sensing channel. Turn on this option if you want the IP core to read the built-in temperature sensor in the ADC block. The sampling rate of the ADC block reduces to 50 kHz when it reads the temperature measurement. After it completes the temperature reading, the ADC sampling rate returns to 1 MHz.

continued...

Parameter	Setting
	For the Altera Modular Dual ADC IP core, if you specify the TSD in a sequencer slot for ADC1, specify NULL in the same sequencer slot number for ADC2.
Enable Maximum threshold for Channel *N*	Turn on this option if you want to set a maximum threshold value for the channel.
Enter Maximum Threshold for Channel *N*	Enter the maximum threshold voltage for the channel. The IP core will generate a threshold violation notification signal to indicate that the sampled data is over the threshold value that you specify.
Enable Maximum threshold for on-chip TSD (TSD tab)	Enter the maximum threshold temperature for the temperature sensor in Celsius. The IP core will generate a threshold violation notification signal to indicate that the sampled temperature is over the temperature that you specify.
Enable Minimum threshold for Channel *N*	Turn on this option if you want to set a minimum threshold value for the channel.
Enter Minimum Threshold for Channel *N*	Enter the minimum threshold voltage for the channel. The IP core will generate a threshold violation notification signal to indicate that the sampled data is below the threshold value that you specify.
Enter Minimum Threshold for on-chip TSD (TSD tab)	Enter the maximum threshold temperature for the temperature sensor in Celsius. The IP core will generate a threshold violation notification signal to indicate that the sampled temperature is below the temperature that you specify.

Table 15. Parameters Settings in Sequencer Group

Parameter	Setting
Number of slot used	Select the number of channels to use for conversion. The parameter editor displays the number of slots available in the **Conversion Sequence Channels** based on your selection.
Slot *N*	For each available slot, select the channel to sample in the sequence. The available channels depend on the channels that you turned on in the **Channels** parameters group. If you turned on a channel but do not select the channel in any of the sequencer slots, the unselected channel is not measured during the ADC sampling sequence. The ADC block samples the measurements in the sequence you specify. After it reaches the last slot in the sequence, it repeats the sampling from the first slot.

Related Links

- Creating Intel MAX 10 ADC Design on page 40
- Customizing and Generating Altera Modular ADC IP Core on page 40
- Completing ADC Design on page 45
- Altera Modular ADC Parameters Settings on page 47
- Altera Modular Dual ADC Parameters Settings on page 51
- Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 49
- Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 53
- Valid ADC Sample Rate and Input Clock Combination on page 54
- User-Specified ADC Logic Simulation Output on page 32
  Provides more information about using your own stimulus input file to simulate the ADC output data.

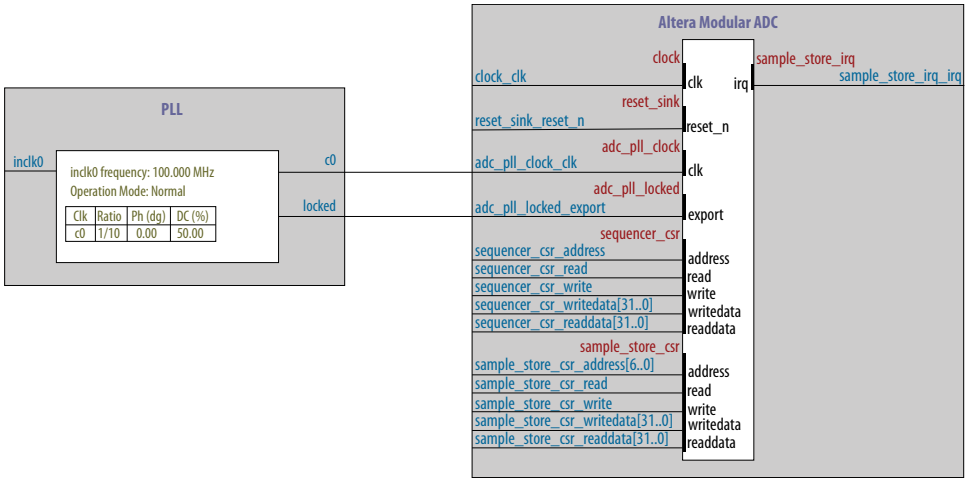
- Guidelines: Board Design for Analog Input on page 35
  Provides more information about the sampling rate and settling time.

4.5 Completing ADC Design

The ADC design requires that the ALTPLL IP core clocks the Altera Modular ADC IP core.

Generate the ALTPLL and Altera Modular ADC IP cores with the settings in the related information.

Figure 30. Basic Intel MAX 10 ADC Design



1. Create the design as shown in the preceding figure.
2. Connect the `c0` signal from the ALTPLL IP core to the `adc_pll_clock_clk` port of the Altera Modular ADC IP core.
3. Connect the `locked` signal from the ALTPLL IP core to the `adc_pll_locked_export` port of the Altera Modular ADC IP core.
4. Create the ADC Avalon slave interface to start the ADC.

Related Links

- Creating Intel MAX 10 ADC Design on page 40
- Parameters Settings for Generating ALTPLL IP Core on page 41
- Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core on page 42
- Configuration 1: Standard Sequencer with Avalon-MM Sample Storage on page 22
- Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection on page 23
- Configuration 3: Standard Sequencer with External Sample Storage on page 24
- Configuration 4: ADC Control Core Only on page 24

5 Altera Modular ADC and Altera Modular Dual ADC IP Cores References

The Altera Modular ADC or Altera Modular Dual ADC IP core is a soft controller for the ADC hard IP blocks. You can generate soft IPs to instantiate the on-chip ADC blocks. With this IP core, you can configure the ADCs and abstract the low level handshake with the ADC hard IP blocks.

The Intel Quartus Prime software generates your customized Altera Modular ADC or Altera Modular Dual ADC IP core according to the parameter options that you set in the parameter editor.

Related Links

- Intel MAX 10 Analog to Digital Converter Overview on page 4
- Altera Modular ADC and Altera Modular Dual ADC IP Cores on page 21
- Altera Modular ADC IP Core Configuration Variants on page 21

5.1 Altera Modular ADC Parameters Settings

There are three groups of options: **General**, **Channels**, and **Sequencer**.

Table 16. Altera Modular ADC Parameters - General

Parameter	Allowed Values	Description
Core Variant	• Standard sequencer with Avalon-MM sample storage • Standard sequencer with Avalon-MM sample storage and threshold violation detection • Standard sequencer with external sample storage • ADC control core only	Selects the core configuration for the Altera Modular ADC IP core.
Debug Path	• Disabled • Enabled	Enables the debug path.
Generate IP for which ADCs of this device?	• 1st ADC • 2nd ADC	For devices that have two ADC blocks, specifies which ADC block you want to instantiate using the IP core.
ADC Sample Rate	25 kHz, 50 kHz, 100 kHz, 200 kHz, 250 kHz, 500 kHz, and 1 MHz	Specifies the ADC sampling rate. The sampling rate you select affects which ADC input clock frequencies are available. Refer to the related information for more details about the sampling rate and the required settling time.
ADC Input Clock	2 MHz, 10 MHz, 20 MHz, 40 MHz, and 80 MHz	Specifies the frequency of the PLL clock counter zero (c0) clock supply for the ADC core clock. • You must configure the c0 of the first ALTPLL IP core that you instantiate to output one of the frequencies in the allowed values list. • Connect the ALTPLL c0 output signal to the Altera Modular ADC clk_in_pll_c0 input signal. For valid ADC sampling rate and input clock frequencies combinations, refer to the related information.
Reference Voltage Source	• External • Internal	Specifies the source of voltage reference for the ADC: • External—uses ADC_VREF pin as the voltage reference source. • Internal—uses the on-chip 2.5 V (3.0/3.3V on voltage-regulated devices) as the voltage reference source.
External Reference Voltage	• Dual supply devices: up to 2.5 V • Single supply devices: up to 3.63 V	Specifies the voltage of ADC_VREF pin if you use it as reference voltage to the ADC.
Enable user created expected output file	• Enabled • Disabled	Specifies the source of output data for ADC logic simulation: • Enabled—uses the stimulus input file you provide for each ADC channel, except the TSD channel, to simulate the output data. • Disabled—uses fixed expected output data for all ADC channels. This is the default setting.

continued...

Parameter	Allowed Values	Description
		For more information about user-specified ADC logic simulation output, refer to the related information.

Table 17. Altera Modular ADC Parameters - Channels

This group of parameters is divided into several tabs—one for each channel, and one tab for the TSD.

Parameter	Allowed Values	Description
Use Channel 0 (Dedicated analog input pin - ANAIN) (CH0 tab)	• On • Off	Enables the dedicated analog input pin.
User created expected output file	—	Specifies user-created stimulus input file to simulate the output data for the channel. This option is available for each enabled channel except the TSD if you select **Enable user created expected output file**.
Use Channel *N* (Each channel in its own tab)	• On • Off	Enables the dual-function analog input, where *N* is: • 1 to 16 channels for single ADC devices • 1 to 8 channels for dual ADC devices
Use on-chip TSD (TSD tab)	• On • Off	Specifies that the IP core reads the built-in temperature sensor in the ADC. If you turn on this option, the ADC sampling rate is up to 50 kHz when it reads the temperature measurement. After it completes the temperature reading, the ADC sampling rate is up to 1 MHz.
Enable Maximum threshold for Channel *N* (Each channel in its own tab)	• On • Off	Enables the maximum threshold feature for the channel. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Maximum threshold for on-chip TSD (TSD tab)	• On • Off	Enables the maximum threshold feature for the TSD. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Maximum Threshold for Channel *N* (Each channel in its own tab, including channel 0)	Depends on reference voltage	Specifies the maximum threshold value in Volts. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Maximum Threshold for on-chip TSD (TSD tab)	—	Specifies the maximum threshold value in Celsius. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Minimum threshold for Channel *N* (Each channel in its own tab, including channel 0)	• On • Off	Enables the minimum threshold feature for the channel. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Minimum threshold for on-chip TSD (TSD tab)	• On • Off	Enables the minimum threshold feature for the TSD. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Minimum Threshold for Channel *N* (Each channel in its own tab, including channel 0)	Depends on reference voltage	Specifies the minimum threshold value in Volts. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Minimum Threshold for on-chip TSD	—	Specifies the minimum threshold value in Celsius.

continued...

Parameter	Allowed Values	Description
(TSD tab)		This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Prescaler for Channel *N*	• On • Off	Enables the prescaler function, where *N* is: • Channels 8 and 16 (if available) for single ADC devices. • Channel 8 of ADC1 or ADC2 for dual ADC devices.

Table 18. Altera Modular ADC Parameters - Sequencer

Parameter	Allowed Values	Description
Number of slot used	1 to 64	Specifies the number of conversion sequence slots to use. The **Conversion Sequence Channels** section displays the slots available according to the number of slots you select here.
Slot *N*	Enabled channel number (CH *N*)	Specifies which enabled ADC channel to use for the slot in the sequence. The selection option lists the ADC channels that you turned on in the **Channels** parameter group.

Related Links

- Sequencer Core on page 27
- Configuration 1: Standard Sequencer with Avalon-MM Sample Storage on page 22
- Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection on page 23
- Configuration 3: Standard Sequencer with External Sample Storage on page 24
- Configuration 4: ADC Control Core Only on page 24
- Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 49
- Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 53
- Valid ADC Sample Rate and Input Clock Combination on page 54
- User-Specified ADC Logic Simulation Output on page 32
  Provides more information about using your own stimulus input file to simulate the ADC output data.
- Guidelines: Board Design for Analog Input on page 35
  Provides more information about the sampling rate and settling time.

5.1.1 Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping

Each ADC channel in the Altera Modular ADC IP core corresponds to different device pin name for single and dual ADC devices.

Table 19. Altera Modular ADC IP Core Channel to Pin Mapping for Single ADC Devices

Channel Name	Pin Name
CH0	ANAIN1
CH1	ADC1IN1

continued...

Channel Name	Pin Name
CH2	ADC1IN2
CH3	ADC1IN3
CH4	ADC1IN4
CH5	ADC1IN5
CH6	ADC1IN6
CH7	ADC1IN7
CH8	ADC1IN8
CH9	ADC1IN9
CH10	ADC1IN10
CH11	ADC1IN11
CH12	ADC1IN12
CH13	ADC1IN13
CH14	ADC1IN14
CH15	ADC1IN15
CH16	ADC1IN16

Table 20. Altera Modular ADC IP Core Channel to Pin Mapping for Dual ADC Devices

ADC Block	Channel Name	Pin Name
First ADC	CH0	ANAIN1
	CH1	ADC1IN1
	CH2	ADC1IN2
	CH3	ADC1IN3
	CH4	ADC1IN4
	CH5	ADC1IN5
	CH6	ADC1IN6
	CH7	ADC1IN7
	CH8	ADC1IN8
Second ADC	CH0	ANAIN2
	CH1	ADC2IN1
	CH2	ADC2IN2
	CH3	ADC2IN3
	CH4	ADC2IN4
	CH5	ADC2IN5
	CH6	ADC2IN6
	CH7	ADC2IN7
	CH8	ADC2IN8

5.2 Altera Modular Dual ADC Parameters Settings

There are three groups of options: **General**, **Channels**, and **Sequencer**.

Table 21. Altera Modular Dual ADC Parameters - General

Parameter	Allowed Values	Description
Core Variant	• Standard sequencer with Avalon-MM sample storage • Standard sequencer with Avalon-MM sample storage and threshold violation detection • Standard sequencer with external sample storage • ADC control core only	Selects the core configuration for the Altera Modular Dual ADC IP core.
ADC Sample Rate	25 kHz, 50 kHz, 100 kHz, 200 kHz, 250 kHz, 500 kHz, and 1 MHz	Specifies the ADC sampling rate. The sampling rate you select affects which ADC input clock frequencies are available. Refer to the related information for more details about the sampling rate and the required settling time.
ADC Input Clock	2 MHz, 10 MHz, 20 MHz, 40 MHz, and 80 MHz	Specifies the frequency of the PLL clock counter zero (c0) clock supply for the ADC core clock. • You must configure the c0 of the first ALTPLL IP core that you instantiate to output one of the frequencies in the allowed values list. • Connect the ALTPLL c0 output signal to the Altera Modular Dual ADC clk_in_pll_c0 input signal. For valid ADC sampling rate and input clock frequencies combinations, refer to the related information.
Reference Voltage (ADC1 or ADC2)	• External • Internal	Specifies the source of voltage reference for the ADC: • External—uses ADC_VREF pin as the voltage reference source. • Internal—uses the on-chip 2.5 V (3.0/3.3V on voltage-regulated devices) as the voltage reference source.
External Reference Voltage	• Dual supply devices: up to 2.5 V • Single supply devices: up to 3.63 V	Specifies the voltage of ADC_VREF pin if you use it as reference voltage to the ADC.
Enable user created expected output file	• Enabled • Disabled	Specifies the source of output data for ADC logic simulation: • Enabled—uses the stimulus input file you provide for each ADC channel, except the TSD channel, to simulate the output data. • Disabled—uses fixed expected output data for all ADC channels. This is the default setting. For more information about user-specified ADC logic simulation output, refer to the related information.

Table 22. Altera Modular Dual ADC Parameters - Channels

This group of parameters is divided into two main tabs for ADC1 and ADC2. For each tab, there are several channel tabs—one for each channel, and one tab for the TSD in ADC1.

Parameter	Allowed Values	Description
Use Channel 0 or 9 (Dedicated analog input pin - ANAIN) (CH0 tab for ADC1 or CH9 tab for ADC2)	• On • Off	Enables the dedicated analog input pin for ADC1 or ADC2.
User created expected output file	—	Specifies user-created stimulus input file to simulate the output data for the channel. This option is available for each enabled channel except the TSD if you select **Enable user created expected output file**.
Use Channel *N* (Each channel in its own tab)	• On • Off	Enables the dual-function analog input, where *N* is: • Channels 1 to 8 for ADC1 • Channels 10 to 17 for ADC2
Use on-chip TSD (TSD tab in ADC1 only)	• On • Off	Specifies that the IP core reads the built-in temperature sensor in ADC1. If you turn on this option, the ADC sampling rate is up to 50 kHz when it reads the temperature measurement. After it completes the temperature reading, the ADC sampling rate is up to 1 MHz. *Note:* If you select the TSD for a sequencer slot in ADC1, select NULL for the same sequencer slot number in ADC2.
Enable Maximum threshold for Channel *N* (Each channel in its own tab)	• On • Off	Enables the maximum threshold feature for the channel. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Maximum threshold for on-chip TSD (TSD tab)	• On • Off	Enables the maximum threshold feature for the TSD. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Maximum Threshold for Channel *N* (Each channel in its own tab, including channel 0)	Depends on reference voltage	Specifies the maximum threshold value in Volts. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Maximum Threshold for on-chip TSD (TSD tab)	—	Specifies the maximum threshold value in Celsius. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Minimum threshold for Channel *N* (Each channel in its own tab, including channel 0)	• On • Off	Enables the minimum threshold feature for the channel. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Minimum threshold for on-chip TSD (TSD tab)	• On • Off	Enables the minimum threshold feature for the TSD. This option is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Minimum Threshold for Channel *N* (Each channel in its own tab, including channel 0)	Depends on reference voltage	Specifies the minimum threshold value in Volts. This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enter Minimum Threshold for on-chip TSD (TSD tab)	—	Specifies the minimum threshold value in Celsius.

continued...

Parameter	Allowed Values	Description
		This setting is available only if you select the **Standard sequencer with Avalon-MM sample storage and threshold violation detection** core variant.
Enable Prescaler for Channel *N*	• On • Off	Enables the prescaler function, where *N* is: • Channel 8 in ADC1 • Channel 17 in ADC2

Table 23. Altera Modular Dual ADC Parameters - Sequencer

Parameter	Allowed Values	Description
Number of slot used	1 to 64	Specifies the number of conversion sequence slots to use for both ADC1 and ADC2. The **Conversion Sequence Channels** section displays the slots available for ADC1 and ADC2 according to the number of slots you select here.
Slot *N*	Enabled channel number (CH *N*)	Specifies which enabled ADC channel to use for the slot in the sequence. The selection option lists the ADC channels that you turned on in the **Channels** parameter group for ADC1 and ADC2. *Note:* If you select the TSD for a sequencer slot in ADC1, select NULL for the same sequencer slot number in ADC2.

Related Links

- Sequencer Core on page 27
- Configuration 1: Standard Sequencer with Avalon-MM Sample Storage on page 22
- Configuration 2: Standard Sequencer with Avalon-MM Sample Storage and Threshold Violation Detection on page 23
- Configuration 3: Standard Sequencer with External Sample Storage on page 24
- Configuration 4: ADC Control Core Only on page 24
- Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 49
- Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 53
- Valid ADC Sample Rate and Input Clock Combination on page 54
- User-Specified ADC Logic Simulation Output on page 32
  Provides more information about using your own stimulus input file to simulate the ADC output data.
- Guidelines: Board Design for Analog Input on page 35
  Provides more information about the sampling rate and settling time.

5.2.1 Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping

Each ADC channel in the Altera Modular Dual ADC IP core corresponds to different device pin name.

Table 24. Altera Modular Dual ADC IP Core Channel to Pin Mapping

ADC Block	Channel Name	Pin Name
ADC1	CH0	ANAIN1
	CH1	ADC1IN1
	CH2	ADC1IN2
	CH3	ADC1IN3
	CH4	ADC1IN4
	CH5	ADC1IN5
	CH6	ADC1IN6
	CH7	ADC1IN7
	CH8	ADC1IN8
ADC2	CH9	ANAIN2
	CH10	ADC2IN1
	CH11	ADC2IN2
	CH12	ADC2IN3
	CH13	ADC2IN4
	CH14	ADC2IN5
	CH15	ADC2IN6
	CH16	ADC2IN7
	CH17	ADC2IN8

5.3 Valid ADC Sample Rate and Input Clock Combination

Each predefined ADC sampling rate supports a list of input clock frequencies. When you configure the ALTPLL IP core to clock the ADC, use an ADC input clock frequency supported by your ADC sampling rate.

The ability to specify the ADC sampling rate allows you more design flexibility. If you are not using the maximum Intel MAX 10 ADC sampling rate, you get a wider settling time margin.

Table 25. Valid Combination of ADC Sampling Rate and Input Clock

Total ADC Sampling Rate (kHz)	ADC Input Clock Frequency (MHz)				
	2	10	20	40	80
1000	Yes	Yes	Yes	Yes	Yes
500	—	Yes	Yes	Yes	—
250	—	Yes	Yes	—	—
200	Yes	—	—	—	—
125	—	Yes	—	—	—

continued...

Total ADC Sampling Rate (kHz)	ADC Input Clock Frequency (MHz)				
	2	10	20	40	80
100	Yes	—	—	—	—
50	Yes	—	—	—	—
25	Yes	—	—	—	—

Related Links

- Parameters Settings for Generating ALTPLL IP Core on page 41
- Parameters Settings for Generating Altera Modular ADC or Altera Modular Dual ADC IP Core on page 42
- Altera Modular ADC Parameters Settings on page 47
- Altera Modular Dual ADC Parameters Settings on page 51

5.4 Altera Modular ADC and Altera Modular Dual ADC Interface Signals

Depending on parameter settings you specify, different signals are available for the Altera Modular ADC or Altera Modular Dual ADC IP core.

5.4.1 Command Interface of Altera Modular ADC and Altera Modular Dual ADC

The command interface is an Avalon-ST type interface that supports a ready latency of 0.

Table 26. Command Interface Signals

Signal	Width (Bit)	Description
valid	1	Indication from the source port that current transfer is valid.
ready	1	Indication from the sink port that it is ready for current transfer.
channel	5	Indicates the channel that the ADC hard block samples from for current command. • 31—recalibration request • 30:18—not used • 17—temperature sensor • 16:0—channels 16 to 0; where channel 0 is the dedicated analog input pin and channels 1 to 16 are the dual purpose analog input pins
startofpacket	1	Indication from the source port that current transfer is the start of packet. • For altera_adc_sequencer core implementation, the IP core asserts this signal during the first slot of conversion sequence data array. • For altera_adc_control core implementation, this signal is ignored. The IP core just passes the received information back to the corresponding response interface.
endofpacket	1	Indication from the source port that current transfer is the end of packet. • For altera_adc_sequencer core implementation, IP core asserts this signal during the final slot of conversion sequence data array. • For altera_adc_control core implementation, this signal is ignored. The IP core just passes the received information back to the corresponding response interface.

Related Links

- Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 49
- Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 53

5.4.2 Response Interface of Altera Modular ADC and Altera Modular Dual ADC

The response interface is an Avalon-ST type interface that does not support backpressure. To avoid overflow condition at the source port, implement sink ports with response data process time that is fast enough, or with enough buffers storage.

Table 27. Response Interface Signals

Signal	Width (Bit)	Description
`valid`	1	Indication from the source port that current transfer is valid.
`channel`	5	Indicates the ADC channel to which the ADC sampling data corresponds for the current response. • 31:18—not used • 17—temperature sensor • 16:0—channels 16 to 0; where channel 0 is the dedicated analog input pin and channels 1 to 16 are the dual purpose analog input pins
`data`	12 or 24	ADC sampling data: • 12 bit width for Altera Modular ADC • 24 bit width for Altera Modular Dual ADC
`startofpacket`	1	Indication from the source port that current transfer is the start of packet. For altera_adc_control core implementation, the source of this signal is from the corresponding command interface.
`endofpacket`	1	Indication from the source port that current transfer is the end of packet. For altera_adc_control core implementation, the source of this signal is from the corresponding command interface.

5.4.3 Threshold Interface of Altera Modular ADC and Altera Modular Dual ADC

The threshold interface is an Avalon-ST type interface that does not support backpressure.

Table 28. Threshold Interface Signals

Signal	Width (Bit)	Description
valid	1	Indication from the source port that current transfer is valid.
channel	5	Indicates the ADC channel for which the threshold value has been violated. • 31:18—not used • 17—temperature sensor • 16:0—channels 16 to 0; where channel 0 is the dedicated analog input pin and channels 1 to 16 are the dual purpose analog input pins
data	1	Indicates the type of threshold violation: • 1—Exceeds maximum threshold value • 0—Below minimum threshold value

Related Links

- Altera Modular ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 49
- Altera Modular Dual ADC IP Core Channel Name to Intel MAX 10 Device Pin Name Mapping on page 53

5.4.4 CSR Interface of Altera Modular ADC and Altera Modular Dual ADC

The CSR interface is an Avalon-MM slave interface.

Table 29. CSR Interface Signals

Signal	Width (Bit)	Description
address	1 or 7	Avalon-MM address bus. The address bus width is in the unit of word addressing: • altera_adc_sample_store core—address width is seven • altera_adc_sequencer core—address width is one
read	1	Avalon-MM read request.
write	1	Avalon-MM write request.
writedata	32	Avalon-MM write data bus.
readdata	32	Avalon-MM read data bus.

5.4.5 IRQ Interface of Altera Modular ADC and Altera Modular Dual ADC

The IRQ interface is an interrupt interface type.

Table 30. IRQ Interface Signals

Signal	Width (Bit)	Description
irq	1	Interrupt request.

5.4.6 Peripheral Clock Interface of Altera Modular ADC and Altera Modular Dual ADC

The peripheral clock interface is a clock sink interface type.

Table 31. Peripheral Clock Interface Signals

Signal	Width (Bit)	Description
clock	1	Single clock that clocks all Altera Modular ADC or Altera Modular Dual ADC micro cores. *Note:* To avoid functional failure, the required minimum peripheral clock frequency is 25 MHz.

5.4.7 Peripheral Reset Interface of Altera Modular ADC and Altera Modular Dual ADC

The peripheral reset interface is a reset sink interface type.

Table 32. Peripheral Reset Interface Signals

Signal	Width (Bit)	Description
reset_n	1	Single reset source that that resets all Altera Modular ADC or Altera Modular Dual ADC micro cores.

5.4.8 ADC PLL Clock Interface of Altera Modular ADC and Altera Modular Dual ADC

The ADC PLL clock interface is a clock sink interface type.

Table 33. ADC PLL Clock Interface Signals

Signal	Width (Bit)	Description
clock	1	ADC hard IP clock source from C0 output of dedicated PLL1 or PLL3. Export this interface from the Qsys system.

Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Parameters Settings for Generating ALTPLL IP Core on page 41
- ADC Clock Sources on page 16
- PLL Locations, Intel MAX 10 Clocking and PLL User Guide
  Provides more information about the availability of PLL3 in different Intel MAX 10 devices and packages.

5.4.9 ADC PLL Locked Interface of Altera Modular ADC and Altera Modular Dual ADC

The ADC PLL locked interface is a conduit end interface type.

Table 34. ADC PLL Locked Interface Signals

Signal	Width (Bit)	Description
conduit	1	ADC hard IP locked signal output of dedicated PLL1 or PLL3. Export this interface from the Qsys system.

Related Links

- Customizing and Generating Altera Modular ADC IP Core on page 40
- Parameters Settings for Generating ALTPLL IP Core on page 41
- ADC Clock Sources on page 16
- PLL Locations, Intel MAX 10 Clocking and PLL User Guide
  Provides more information about the availability of PLL3 in different Intel MAX 10 devices and packages.

5.5 Altera Modular ADC Register Definitions

The registers in the generated Altera Modular ADC IP core provide the IP core with the control and settings during operation.

5.5.1 Sequencer Core Registers

Table 35. Command Register (CMD)

Address Offset: 0x0

Bit	Name	Attribute	Description	Value	Default
31:4	Reserved	Read	Reserved.	—	0
3:1	Mode	Read-Write	Indicates the operation mode of the sequencer core. These bits are ignored when the run bit (bit 0) is set. In continuous conversion, the data will be overwritten after a complete sampling sequence.	• 7—Recalibrate the ADC • 6 to 2—Reserved • 1—Single cycle ADC conversion • 0—Continuous ADC conversion	0
0	Run	Read-Write	Use this control bit to trigger the sequencer core operation. The Altera Modular ADC IP core waits until the sequencer core completes its current operation before writing to this register bit.	• 1—Run • 0—Stop	0

Related Links

Sequencer Core on page 27

5.5.2 Sample Storage Core Registers

Table 36. ADC Sample Register (ADC_SAMPLE) of Altera Modular ADC

Address Offset: 0x3F (slot 64)—0x0 (slot 1)

Bit	Name	Attribute	Description	Value	Default
31:12	Reserved	Read	Reserved.	—	0
11:0	Sample	Read	The data sampled by the ADC for the corresponding slot.	Sampled data	0

Table 37. ADC Sample Register (ADC_SAMPLE) of Altera Modular Dual ADC

Address Offset: 0x3F (slot 64)—0x0 (slot 1)

Bit	Name	Attribute	Description	Value	Default
31:28	Reserved	Read	Reserved.	—	0
27:16	Sample	Read	The data sampled by ADC2 for the corresponding slot.	Sampled data	0
15:12	Reserved	Read	Reserved.	—	0
11:0	Sample	Read	The data sampled by ADC1 for the corresponding slot.	Sampled data	0

Table 38. Interrupt Enable Register (IER)

Address Offset: 0x40

Clear the enable bit to prevent the corresponding interrupt status bit from causing interrupt output assertion (IRQ). The enable bit does not stop the interrupt status bit value from showing in the interrupt status register (ISR).

Bit	Name	Attribute	Description	Value	Default
31:1	Reserved	Read	Reserved.	—	0
0	M_EOP	Read-Write	The enable bit for the end of packet (EOP) interrupt.	• 1—Enables the corresponding interrupt • 0—Disables the corresponding interrupt	1

Table 39. Interrupt Status Register (ISR)

Address Offset: 0x41

Bit	Name	Attribute	Description	Value	Default
31:1	Reserved	Read	Reserved.	—	0
0	EOP	Read-Write (one cycle)	EOP interrupt.	This bit is automatically set by the hardware. When "1", it indicates that a packet of samples is stored and ready to be read. You can retrieve the sample value from the	0

continued...

Bit	Name	Attribute	Description	Value	Default
				ADC_SAMPLE register. To clear this bit to "0" for the next interrupt, write "1".	

Related Links

Sample Storage Core on page 28

5.6 ADC HAL Device Driver for Nios II Gen 2

The Altera Modular ADC IP core provides a HAL device driver. You can integrate the device driver into the HAL system library for Nios II Gen 2 systems.

The Altera Modular ADC IP core provides software files that define low-level access to the hardware. You can use the macros definition and functions in the software files to initialize the Altera Modular ADC core.

- `altera_modular_adc_sequencer_regs.h`—this file defines the register map for the sequencer core. It provides symbolic constants to access the low-level hardware.
- `altera_modular_adc_sample_store_regs.h`—this file defines the register for sample storage core. It provides symbolic constants to access the low-level hardware.
- `altera_modular_adc.h`—include this file into your application. It automatically includes the other header files and defines additional functions.
- `altera_modular_adc.c`—this file implements helper functions that are defined in the header file.

Related Links

HAL API Reference, Nios II Gen 2 Software Developer's Handbook
Provides more information about the HAL API.

6 Intel MAX 10 Analog to Digital Converter User Guide Archives

If an IP core version is not listed, the user guide for the previous IP core version applies.

IP Core Version	User Guide
17.0	MAX 10 Analog to Digital Converter User Guide
16.1	MAX 10 Analog to Digital Converter User Guide
16.0	MAX 10 Analog to Digital Converter User Guide
15.1	MAX 10 Analog to Digital Converter User Guide
15.0	MAX 10 Analog to Digital Converter User Guide

7 Document Revision History for Intel MAX 10 Analog to Digital Converter User Guide

Date	Version	Changes
December 2017	2017.12.15	• Added single power supply U324 package. • Added a topic that shows the equations and calculation example to convert between analog voltage values and digital representation. • Updated the topic about ADC timing to add time calculation examples for the ADC control core based on the sampling rate. • Added a note to specify that the Altera Modular ADC and Altera Modular Dual ADC IP cores support generating only Verilog simulation scripts. VHDL simulation is not supported. • Added note to the topic about the peripheral clock interface signal to specify that the peripheral clock frequency must be at least 25 MHz. • Updated the slot numbers of the ADC sample register address offset to correspond to the sequence slot numbers in Intel Quartus Prime.
July 2017	2017.07.06	Updated the description of the EOP bit "0" of the Interrupt Status Register (ISR) to improve clarity.
February 2017	2017.02.21	Rebranded as Intel.
January 2017	2017.01.25	Added a topic that lists the actual TSD sampling rate based on the ADC sampling rate selected in the IP core.
October 2016	2016.10.31	• Updated the topic about the ADC voltage reference to specify that you must use clean external voltage reference with a maximum resistance of 100 Ω. • Updated the topic about the ADC sequencer to clarify that "conversion mode" refers to the sequencer conversion mode, namely the single-cycle and continuous ADC conversion modes. • Added a related information link to a topic in the *Intel MAX 10 Clocking and PLL User Guide* that lists the availability of PLL1 and PLL3 in different Intel MAX 10 devices and packages. • Updated various topics throughout the user guide to improve the clarity of descriptions related to the user-specified ADC logic simulation output feature. • Updated the VCCVREF pin name to ADC_VREF. • Edited the board design guidelines for analog input: — Updated text to improve clarity. — Updated the $F_{cutoff\ @\ -3dB}$ recommendation from "five times" to "at least two times" the input frequency. — Updated the figure showing the first order active low pass filter example.
May 2016	2016.05.02	• Removed all preliminary marks. • Added new function to specify predefined ADC sampling rate up to 1 MSPS. Previously, the ADC always operate at the maximum sampling rate. • Removed link to a workaround to reduce the sampling rate. Now you can set the sampling rate in the IP core parameter editor. • Added the ADC Toolkit that supports the Altera Modular ADC and Altera Modular Dual ADC IP cores.

continued...

Date	Version	Changes
		• Added feature to simulate ADC output using your own expected output files for each ADC channel except the TSD channel. • Corrected the description for bits 11:0 and bits 27:16 of the ADC sample register for Altera Modular ADC and Altera Modular Dual ADC IP cores. Bits 11:0 and bits 27:16 hold the actual 12 bit sampled data for the storage slot instead of the slot number. • Corrected the default value for bit 0 of the interrupt enable register (IER) and interrupt status register (ISR). The default value for M_EOP is 1 and for EOP is 0.
November 2015	2015.11.02	• Added related information link to *Introduction to Altera IP Cores*. • Added links to instructional videos that demonstrate how to create ADC designs in Intel MAX 10 devices. • Changed instances of *Quartus II* to *Quartus Prime*.
June 2015	2015.06.11	Updated the board design guidelines for analog input.
May 2015	2015.05.04	• Added the Altera Modular Dual ADC IP core. • Removed F672 from the 10M25 device and added ADC information for the E144 package of the 10M04 device: — Updated the ADC block counts. — Updated the ADC vertical migration support. — Updated the ADC channel counts. • Updated the table that lists the ADC channel count to list only 8 dual function pins (instead of 16) for the M153 and U169 packages. • Updated the ADC vertical migration diagram to clarify that there are single ADC devices with eight and 16 dual function pins. • Updated the topic about ADC conversion to specify that in prescaler mode, the analog input in dual and single supply devices can measure up to 3.0 V and 3.6 V, respectively. • Updated the ADC IP core architecture figures to include features for the dual ADC IP core. • Added information and topics about the response merge and dual ADC synchronizer micro cores. • Removed notes about contacting Altera for the ADC pin RLC filter design. • Updated the ADC prescaler topic to change the ADC2 channel that supports prescaler from channel 16 to channel 17. • Updated the diagram that shows the ADC timing: — To clarify that the numbers are hexadecimal numbers. — Relabeled the signals to match the command and response interface signal names. • Updated the RC constant and filter value and the filter design example figure to clarify the source of the example values. • Added guidelines for setting up the sequencer in dual ADC mode. • Added topics that list the mapping of Altera Modular ADC and Altera Modular Dual ADC IP cores channel names to Intel MAX 10 device pin names.

continued...

Date	Version	Changes
		• Corrected the address offset of the interrupt enable register (from 0x41 to 0x40) and interrupt status register (from 0x40 to 0x41) for the sample storage core. • Updated the sample storage core registers table to include registers for Altera Modular Dual ADC. • Removed statements about availability of the threshold trigger feature in a future version of the Intel Quartus Prime software. The feature is now available from version 15.0 of the software.
December 2014	2014.12.15	• Added ADC prescaler block diagram. • Replaced the ADC continuous conversion timing diagram with the ADC timing diagram. • Corrected a minor error in the example in the topic about the sample storage core. • Added information that the ADC TSD measures the temperature using a 64-samples running average method. • Updated majority of the temperature codes in the table that lists the temperature code conversion. • Added chapter that provides the ADC design considerations. • Removed mention of value "0" for values allowed for the number of sequencer slots used in Altera Modular ADC IP core parameter editor. Only values 1 to 64 are allowed. • Removed the statement about enabling and disabling additional ADC response interface or debugging in the topic about the Altera Modular ADC IP core configuration variants. You can enable or disable the debug path in the parameter editor. • Removed the debug paths diagrams for each ADC core configuration. • Removed the statement about using the sequencer core to trigger recalibration. The ADC is automatically recalibrated when it switches from normal sensing mode to temperature sensing mode. • Edited text to clarify about routing power or ground traces if power or ground plane is not possible. • Updated the total RC constant values in the table that shows the RC constant and filter values calculation. • Corrected spelling for "prescaler".
September 2014	2014.09.22	Initial release.