ADC User Guide for Nexus Platform

Technical Note

FPGA-TN-02129-2.0

July 2025

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
ADC	Analog to Digital Converter
CAL	Calibration
CDC	Clock Domain Crossing
COG	Conversion On-Going
DCS	Dynamic Clock Select
DNL	Differential Non-Linearity
DTR	Digital Temperature Readout
EOC	End of Conversion
FIFO	First In, First Out
GUI	Graphical User Interface
I/O	Input/Output
IP	Intellectual Property
INL	Integral Non-Linearity
KSPS	Kilo Samples Per Second
LRC	Lower Right Corner
LSB	Least Significant Bit
NTAT	Inversely Proportional to Absolute Temperature
PLL	Phase Locked Loop
RTL	Register Transfer Level
SAR	Successive Approximation Register
SOC	Start of Conversion

1. Introduction

1.1. Overview

The Lattice Nexus™ product line includes the CrossLink™-NX, Certus™-NX, CertusPro™-NX, and MachXO5™-NX families of devices. All families feature an integrated analog block consisting of two Analog to Digital Converters (ADC) and three analog continuous-time comparators.

The ADC is implemented with Successive Approximation Register (SAR) architecture and provides 12-bit resolution with up to 623 ksps (kilo samples per second) conversion speed. Each ADC has a differential analog MUX to select one of the following: eight external dual-mode differential pin pairs, four internal signals, or one dedicated external differential analog pin pair. The dual-mode pins can be either digital I/O or analog channel input.

The continuous-time comparator can be used to monitor either separate dual-mode pin-pairs or the ADC channel inputs in parallel. The output of the comparator is provided as continuous and latched outputs.

1.2. Features

The key features of the Analog IP include:

- Two ADCs
  - 12-bit resolution
  - 623 ksps conversion rate for each ADC. Both ADCs sample at the same time.
  - Selectable input signal
    - Dedicated external input pair
    - 8 dual-mode external input pairs for differential sensing. These pins can be used for other purposes on the FPGA device if they are not needed for the ADC.
    - Ability to sense internal Voltage Rails
    - Internal Junction Temperature Sensing Diode (DTR)
    - External 1.0 V to 1.8 V Reference Voltage
  - Input signal range 0 V to VREF (1.0 V to 1.8 V based on external reference)
  - Unipolar or Bipolar input conversion
  - Three continuous-time comparators
  - Straight binary or 2's compliment output

2. ADC Module Description

2.1. Overview

The ADC analog IP generated from IP express consists of 2 ADCs. Only 1 ADC analog IP can be generated in an FPGA device. Each ADC consists of a hard macro and is a 12-bit, 623 ksps SAR architecture converter that requires an external voltage reference to meet the specifications listed in the data sheet. Each ADC can convert up to a maximum 1.8 V input signal with 1.8 V reference voltage and the input signal can be converted in unipolar mode or bipolar mode. The two ADCs can sample the input sequentially and the sampling periods are programmable by changing the adc_soc_i (start of conversion) signal period. Before either ADC can be used for accurate results, they need to be calibrated. Both ADCs are calibrated when the adc_cal_i signal is active and calibration is complete when the adc_calrdy_o signal is active.

The Nexus ADC hard IP block consists of the following three main blocks: ADC 0, ADC 1, and Control/Status, as shown in Figure 2.1.

- Control status block does a clock domain conversion from fab_clk_i to adc_clk_i domain.
- ADC 0 Block (see Figure 2.2):
  - Three continuous-time comparators with synchronous and latched outputs
  - 12 Input Analog MUX to sense external voltages and internal supply voltages
  - 12-bit SAR ADC
  - Internal Reference Voltage Generation Block[1]
- ADC 1 Block (see Figure 2.3):
  - 12 Input Analog MUX to sense external voltages and internal supply voltages
  - 12-bit SAR ADC
  - DTR (on die temperature sensor)
  - Internal Reference Voltage Generation Block[1]

Note:

1. It is recommended to use external reference for higher ADC accuracy. The internal reference is mainly intended for internal testing, has approximately 10% accuracy (worst case nearly 20%), and may not be fully tested in production.

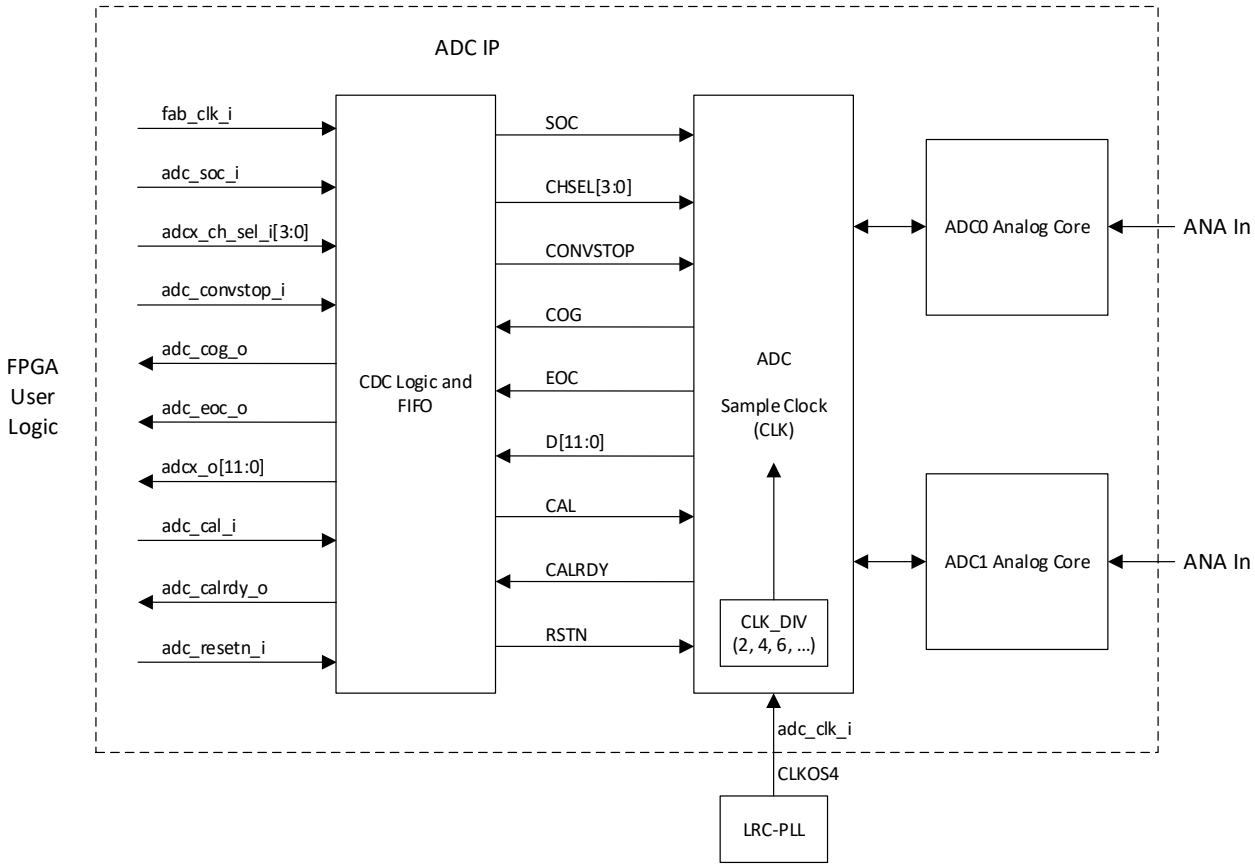

Figure 2.1. ADC Core Module Control and Status Block

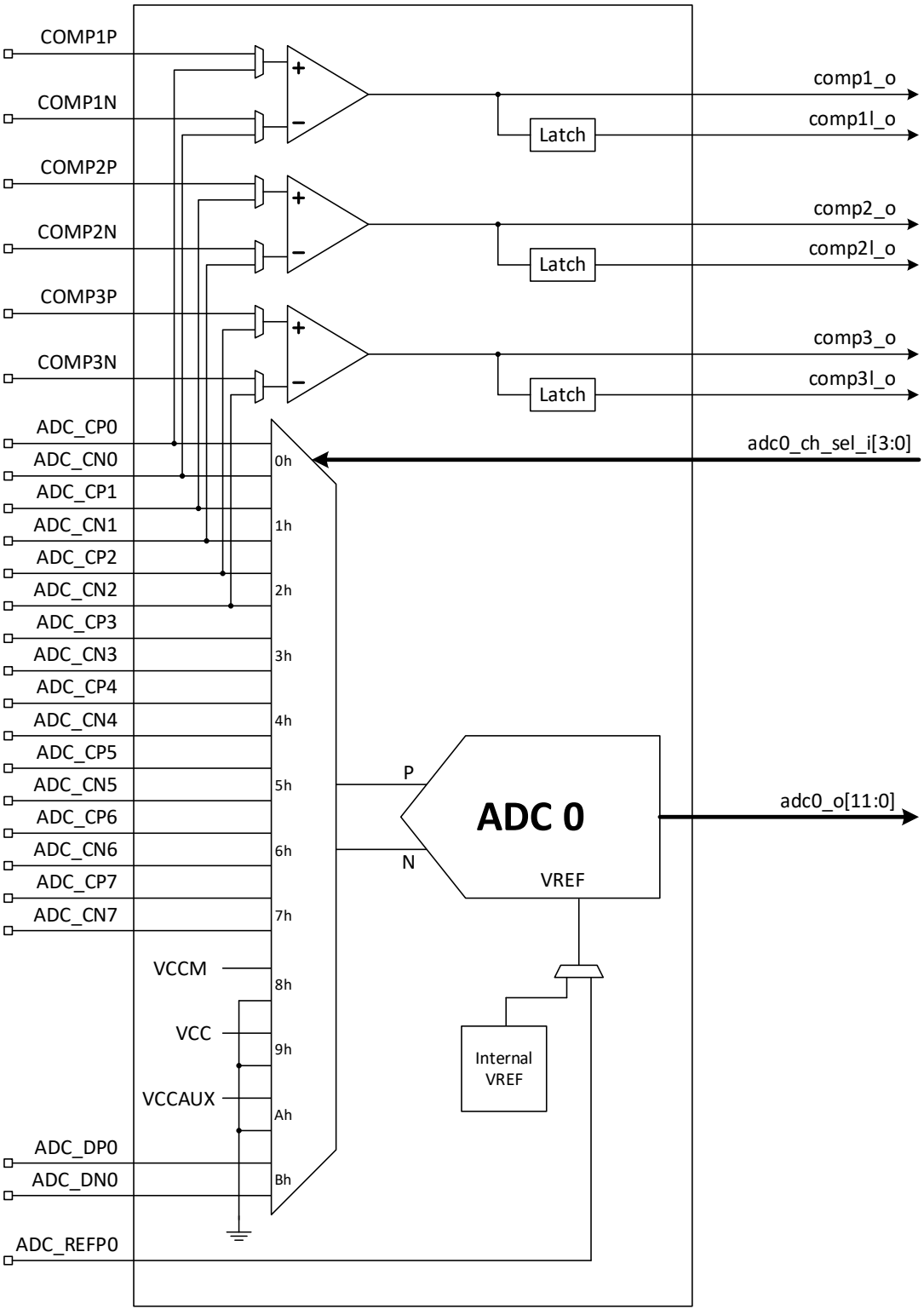

Figure 2.2. ADC Core Module ADC 0 Block

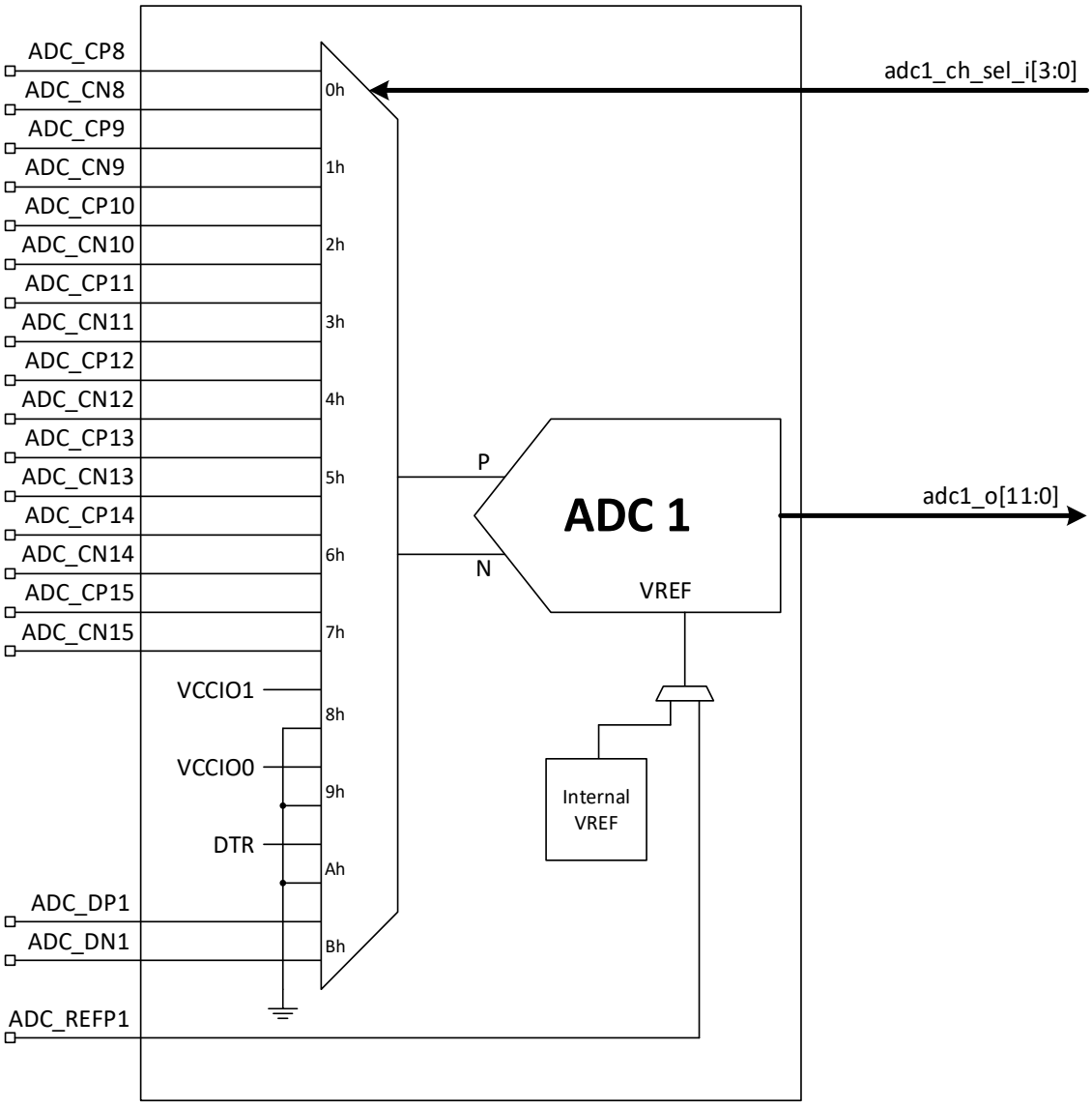

Figure 2.3. ADC Core Module ADC 1 Block

2.2. Modules Description

2.2.1. ADC Cores

The ADC Cores are implemented with an SAR (Successive Approximation Register) architecture. The SAR architecture utilizes a binary search algorithm which results in a fast and consistent conversion time. Each of the ADC Cores convert unipolar or bipolar input signal into 12-bit resolution data with maximum 623 ksps conversion speed sequentially or simultaneously.

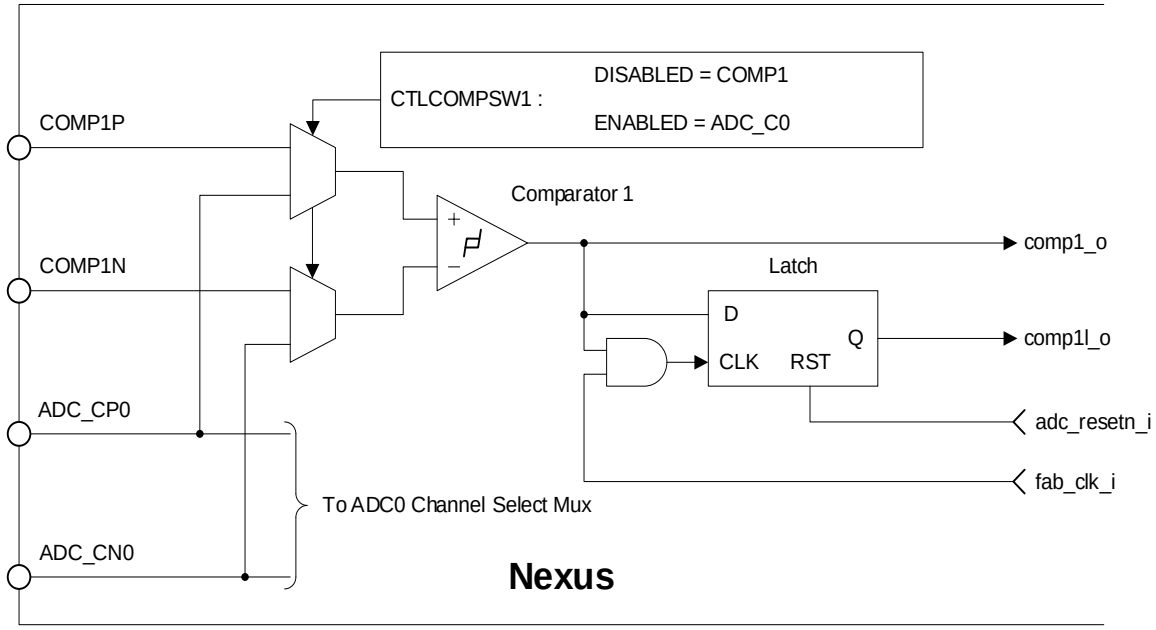

Figure 2.4. Comparator 1 Block Diagram

2.2.2. Comparators

Three analog comparators are available. Both the negative (-) and the positive (+) inputs must be connected to external signals. The output of each comparator is provided both as a continuous-time signal and a latched signal, as shown in Figure 2.4.

The inputs can be provided from either dedicated pins or shared with the ADC channel:

- Comparator 1 can use either COMP1IP and COMP1IN or ADC_CP0 and ADC_CN0.
- Comparator 2 can use either COMP2IP and COMP2IN or ADC_CP1 and ADC_CN1.
- Comparator 3 can use either COMP3IP and COMP3IN or ADC_CP2 and ADC_CN2.

The choice of dedicated pins or reuse of ADC pins can be decided in the *comparator* tab of the IP catalog GUI for the ADC at synthesis time. Some packages do not have dedicated comparator pin. See the Package Options section for more information.

2.2.3. Calibration Controller

A calibration must be run after the ADC block is enabled and before the first conversation is started. The ADC auto-calibration function calibrates gain and offset errors. The ADC calibration also corrects for process variations and devices mismatches, optimizing its performance. Calibration should be initiated by setting the adc_cal_i signal active at least eight clock cycles after reset (adc_resetn_i) is deasserted. Calibration needs to be performed any time after the ADC block reset has been activated by adc_resetn_i going from 0 to 1.

2.2.4. Voltage Reference

The ADC reference voltage must be provided by an external reference to meet the data sheet specifications. The external reference voltage should have at least ± 0.2% accuracy. The ADC can convert a maximum 1.8 V input signal with 1.8 V reference voltage.

The ADC reference voltage must be higher than or equal to any voltage applied to any of the channel inputs; this includes the power supply sensor channels after they are divided by 2.5. If the reference voltage is as little as 0.1 V lower than an applied input, then the ADC result from all channels are unreliable, as the internal circuitry inside the ADC does not work.

2.3. Signal Description

Table 2.1 lists the top-level input and output signals for ADC Core Module. The port names in upper-case represent signals that should be connected to physical pins of the device and port names in lower-case represent internal signals. All digital signals are on fab_clk_i rising edge unless otherwise specified.

Table 2.1. ADC Module Ports

Port Name	Direction	Level	Description
Clocks and Reset			
adc_clk_i	In	Digital	ADC clock drives the converter core (from lower right PLL secondary output number 4). The clock frequency is divided by the internal clock divider, and the frequency after division should not exceed 25 MHz.
fab_clk_i	In	Digital	Fabric clock synchronizes the ADC control and status signals and Comparator latches. All digital I/Os are on this clock. The maximum frequency is 40 MHz.
adc_resetn_i	In	Digital	Active Low reset to ADC, Calibration, and Comparator latches. If 0, the ADC is internally in a low power state. All logic inside the ADC is reset. Calibration information is lost and all conversions are stopped.
Control and Status			
adc_en_i	In	Digital	ADC enable
adc0_o[11:0]	Out	Digital	12-bit ADC 0 output
adc1_o[11:0]	Out	Digital	12-bit ADC 1 output
adc0_ch_sel_i[3:0]	In	Digital	ADC 0 channel input selection
adc1_ch_sel_i[3:0]	In	Digital	ADC 1 channel input selection
adc_soc_i	In	Digital	Start of conversion for both ADC 0 and ADC 1
adc_eoc_o	Out	Digital	End of conversion for both ADC 0 and ADC 1
adc_cog_o	Out	Digital	Conversion on-going signal for both ADC 0 and ADC 1
adc_convstop_i	In	Digital	Stop on-going conversion control for both ADC 0 and ADC 1
adc_cal_i	In	Digital	Calibration start for both ADC 0 and ADC 1
adc_calrdy_o	Out	Digital	End of calibration for both ADC 0 and ADC 1
comp1_o	Out	Digital	Comparator 1 output (asynchronous)
comp1l_o	Out	Digital	Comparator 1 output flopped on fab_clk_i
comp2_o	Out	Digital	Comparator 2 output (asynchronous)
comp2l_o	Out	Digital	Comparator 2 output flopped on fab_clk_i
comp3_o	Out	Digital	Comparator 3 output (asynchronous)
comp3l_o	Out	Digital	Comparator 3 output flopped on fab_clk_i
Analog Inputs			
ADC_DP0	In	Analog	ADC 0 Dedicated positive input
ADC_DN0	In	Analog	ADC 0 Dedicated negative input
ADC_DP1	In	Analog	ADC 1 Dedicated positive input
ADC_DN1	In	Analog	ADC 1 Dedicated negative input
COMP1IP	In	Analog	Comparator 1 positive external input
COMP1IN	In	Analog	Comparator 1 negative external input
COMP2IP	In	Analog	Comparator 2 positive external input
COMP2IN	In	Analog	Comparator 2 negative external input
COMP3IP	In	Analog	Comparator 3 positive external input
COMP3IN	In	Analog	Comparator 3 negative external input
ADC_CP0	In	Analog	ADC 0 Channel 0 positive input (shared with comparator 1)
ADC_CN0	In	Analog	ADC 0 Channel 0 negative input (shared with comparator 1)
ADC_CP1	In	Analog	ADC 0 Channel 1 positive input (shared with comparator 2)

Port Name	Direction	Level	Description
ADC_CN1	In	Analog	ADC 0 Channel 1 negative input (shared with comparator 2)
ADC_CP2	In	Analog	ADC 0 Channel 2 positive input (shared with comparator 3)
ADC_CN2	In	Analog	ADC 0 Channel 2 negative input (shared with comparator 3)
ADC_CP3	In	Analog	ADC 0 Channel 3 positive input
ADC_CN3	In	Analog	ADC 0 Channel 3 negative input
ADC_CP4	In	Analog	ADC 0 Channel 4 positive input
ADC_CN4	In	Analog	ADC 0 Channel 4 negative input
ADC_CP5	In	Analog	ADC 0 Channel 5 positive input
ADC_CN5	In	Analog	ADC 0 Channel 5 negative input
ADC_CP6	In	Analog	ADC 0 Channel 6 positive input
ADC_CN6	In	Analog	ADC 0 Channel 6 negative input
ADC_CP7	In	Analog	ADC 0 Channel 7 positive input
ADC_CN7	In	Analog	ADC 0 Channel 7 negative input
ADC_CP8	In	Analog	ADC 1 Channel 8 positive input
ADC_CN8	In	Analog	ADC 1 Channel 8 negative input
ADC_CP9	In	Analog	ADC 1 Channel 9 positive input
ADC_CN9	In	Analog	ADC 1 Channel 9 negative input
ADC_CP10	In	Analog	ADC 1 Channel 10 positive input
ADC_CN10	In	Analog	ADC 1 Channel 10 negative input
ADC_CP11	In	Analog	ADC 1 Channel 11 positive input
ADC_CN11	In	Analog	ADC 1 Channel 11 negative input
ADC_CP12	In	Analog	ADC 1 Channel 12 positive input
ADC_CN12	In	Analog	ADC 1 Channel 12 negative input
ADC_CP13	In	Analog	ADC 1 Channel 13 positive input
ADC_CN13	In	Analog	ADC 1 Channel 13 negative input
ADC_CP14	In	Analog	ADC 1 Channel 14 positive input
ADC_CN14	In	Analog	ADC 1 Channel 14 negative input
ADC_CP15	In	Analog	ADC 1 Channel 15 positive input
ADC_CN15	In	Analog	ADC 1 Channel 15 negative input

3. Functional Overview

3.1. Introduction

All waveforms given in this section, unless otherwise stated, are captured using Reveal Logic Analyzer on a CertusPro-NX board. Reveal Logic Analyzer is a logic analyzer that can be synthesized inside the FPGA, and this ensures the accuracy of the waveforms as they come from production silicon.

The ADC pin name mapping is given in Table 3.1.

Table 3.1. ADC Pin Name Mapping

Port Name	Description
adc_clk_i	ADC clock drives the converter core (from lower right PLL secondary output number 4).
fab_clk_i	Fabric clock synchronizes the ADC control and status signals, and Comparator latches.
adc_resetn_i	Active Low reset to ADC, Calibration, and Comparator latches.
adc_en_i	ADC enable
adc0_o[11:0]	12-bit ADC 0 output
adc1_o[11:0]	12-bit ADC 1 output
adc0_ch_sel_i[3:0]	ADC 0 channel input selection
adc1_ch_sel_i[3:0]	ADC 1 channel input selection
adc_soc_i	Start of conversion
adc_eoc_o	End of conversion
adc_cog_o	Conversion on-going signal
adc_convstop_i	Stop on-going conversion control
adc_cal_i	Start of calibration
adc_calrdy_o	End of calibration

3.2. ADC Enable and Reset

If the adc_resetn_i signal is 0, all internal logic, the calibration values and the comparator latches are reset. At power on, adc_resetn_i should be 0, and then made 1 when the ADC needs to operate. If adc_resetn_i is 0, then the ADC is in its lowest power state. After ADC comes out of reset, it needs to rerun calibration.

The adc_en_i signal enables the internal clock gating logic inside the ADC. Making it 0 does not reset the internal calibration state. Hence if the ADC needs to enter a low power state, but needs to quickly exit the low power state and reenter is operating state, make adc_en = 0.

3.3. Calibration

The ADC calibration starts with adc_cal_i going high, which should be after a delay after adc_resetn_i and adc-en_i become 1. The minimum delay is 12 fab_clk_i cycles or 8 adc_clk_i cycles, whichever is longer. When the ADC calibration is complete, adc_calrdy_o goes high to indicate that the ADC is calibrated. The timing diagram shown in Figure 3.1 includes a first conversion starting just after the calibration conclusion. Note that if adc_resetn_i is 0, the ADC returns to an uncalibrated state and needs a subsequent calibration cycle for accurate results. The calibration cycle takes approximately 5500 adc_clk_i cycles to complete, and this is shown in Figure 3.2.

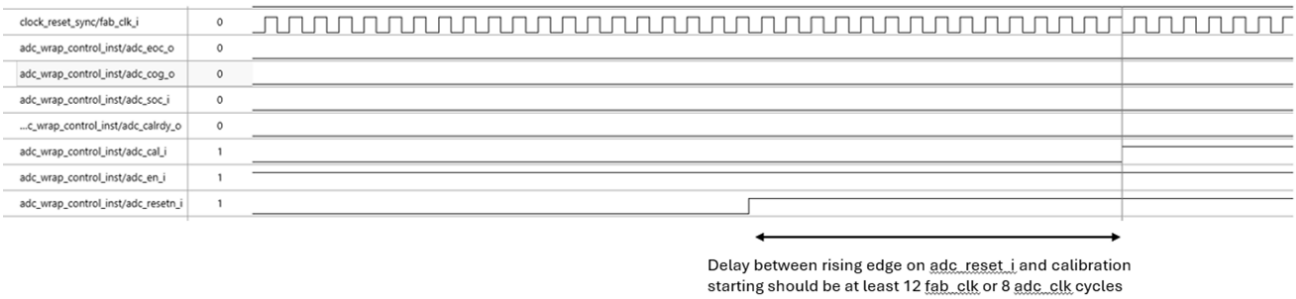

Figure 3.1. ADC Timing Diagram: Calibration Starting After Reset

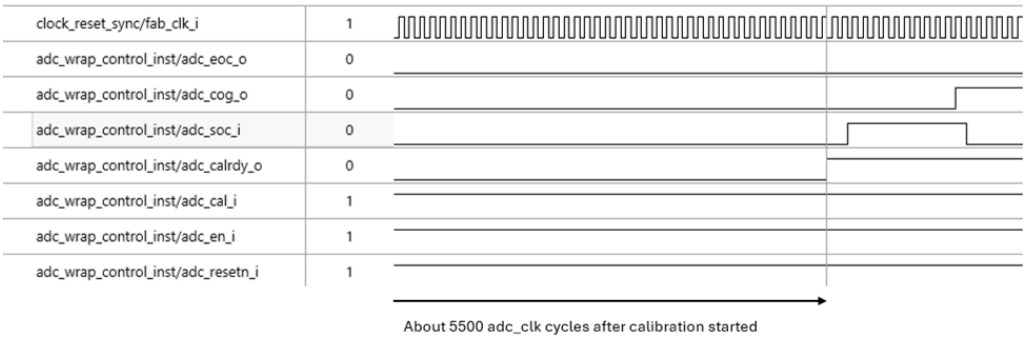

Figure 3.2. ADC Calibration Ending and Conversion Starting (After 5500 adc_clk_i Cycles)

3.4. Reference Voltage

ADC reference voltages are provided on pin ADC_REFP0 for ADC 0 and pin ADC_REFP1 for ADC 1. These pins can be connected to a single reference or separate references. The reference sets both the maximum conversion level of the ADC and the accuracy of the conversion. The ADC can also be configured to use its internal reference. It is not recommended to use the internal reference because it has 10% accuracy (worst case 20%) and is not fully tested in production.

3.5. ADC Analog Input Selection

Inputs adc0_ch_sel_i and adc1_ch_sel_i are used to select the input channel to the ADC. Refer to Table 3.2 and Table 3.3 for input selection.

The ADC's reference voltage must be higher than or equal to any voltage applied to any of the channel inputs; this includes the divided-down power supply sensor channels. If the reference voltage is as little as 0.1 V lower than an applied input, then the ADC result from all channels are unreliable, as the internal circuitry inside the ADC does not work.

Table 3.2. ADC 0 Inputs

adc0_ch_sel[3:0]	Input selected
0	ADC_CP0, ADC_CN0
1	ADC_CP1, ADC_CN1
2	ADC_CP2, ADC_CN2
3	ADC_CP3, ADC_CN3
4	ADC_CP4, ADC_CN4
5	ADC_CP5, ADC_CN5
6	ADC_CP6, ADC_CN6
7	ADC_CP7, ADC_CN7
8	VCCM. Internal divider of 2.5 before ADC input.

adc0_ch_sel[3:0]	Input selected
9	VCC. Internal divider of 2.5 before ADC input.
A	VCCAUX. Internal divider of 2.5 before ADC input.
B	ADC_DP0, ADC_DN0

Note: Other inputs are not valid and can lead to undefined behavior.

Table 3.3. ADC 1 Inputs

adc1_ch_sel[3:0]	Input selected
0	ADC_CP8, ADC_CN8
1	ADC_CP9, ADC_CN9
2	ADC_CP10, ADC_CN10
3	ADC_CP11, ADC_CN11
4	ADC_CP12, ADC_CN12
5	ADC_CP13, ADC_CN13
6	ADC_CP14, ADC_CN14
7	ADC_CP15, ADC_CN15
8	VCCIO1. Internal divider of 2.5 before ADC input.
9	VCCIO0. Internal divider of 2.5 before ADC input.
A	DTR. Refer to the DTR (Digital Temperature Readout) section for more information.
B	ADC_DP1, ADC_DN1

Note: Other inputs are not valid and can lead to undefined behavior.

3.5.1. ADC Voltage Scaling

In unipolar mode, for external voltage inputs, an ADC output of 0 means that the P input and N input are the same voltage. If the output is 12'h3FF, it indicates that the difference between the P input and the N input is equal to Vref. The measured voltage of the ADC can be calculated using the following equation:

$$VADC = \frac{ADC\ output\ code}{4096} \times Vref \qquad Eq.1.$$

3.5.2. Power Supply Sensor

There are five on-chip power supply sensors that can monitor the FPGA power supply voltages. The sensors can measure VCCIO0, VCCIO1, VCCAUX, and VCC on the package power supply balls. VCCM is an internal supply that is derived from VCCAUX and should measure around 1.25 V. Measuring of VCCM needs to be done for internal testing only, because if VCCAUX is within the correct voltage range then VCCM is also within range. All the power supply sensor signals are passed through a divider before being measured; the power supply voltage can be calculated from the ADC code using Equation 2.

$$Power\ Supply\ Voltage = \frac{ADC\ output\ code}{4096} \times 2.5 \times Vref \qquad Eq.2.$$

If a supply is to be monitored by the ADC, its maximum possible value, under all conditions (such as startup, overvoltage, or load changes) should never exceed Vref/2.5 V. Violating this specification can cause all ADC readings to be incorrect, and potentially damage the die.

3.5.3. DTR (Digital Temperature Readout)

On-die junction temperature can be monitored using internal NTAT (Inversely Proportional to Absolute Temperature) characteristic BJT diode voltage, see Figure 3.3. The DTR output code values are shown in Table 3.4 for various configurations and reference voltages.

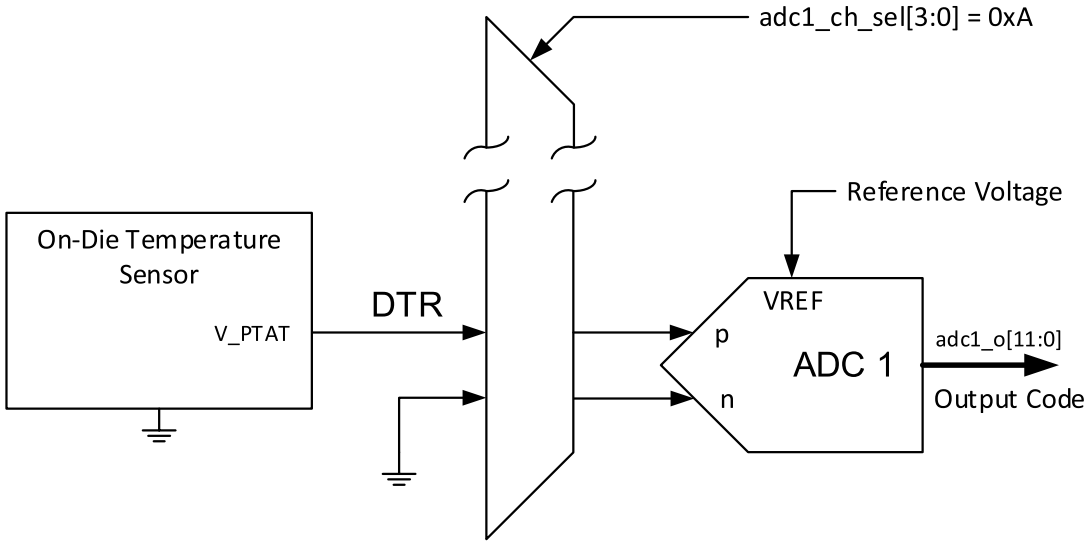

Figure 3.3. Digital Temperature Readout

Table 3.4. DTR Output Code Values as a Function of Temperature and Configuration

Temperature (°C)	V_NTAT (V)	ADC Output Code (Decimal)[1]					
		Uni-Polar VREF				Bi-Polar VREF	
		1.0 V	1.2 V	1.5 V	1.8 V	1.5 V	1.8 V
–40	0.8337	3414	2845	2276	1897	4096	3794
–30	0.8164	3343	2786	2229	1857	4096	3715
–20	0.7990	3272	2727	2181	1818	4096	3636
–10	0.7817	3201	2668	2134	1778	4096	3557
0	0.7643	3130	2608	2087	1739	4096	3478
10	0.7470	3059	2549	2039	1699	4079	3399
20	0.7296	2988	2490	1992	1660	3984	3320
30	0.7123	2917	2431	1945	1620	3890	3241
40	0.6949	2846	2372	1897	1581	3795	3162
50	0.6776	2775	2312	1850	1541	3700	3083
60	0.6602	2704	2253	1802	1502	3605	3004
70	0.6429	2633	2194	1755	1462	3511	2925
80	0.6256	2562	2135	1708	1423	3416	2846
90	0.6082	2491	2076	1660	1384	3321	2768
100	0.5909	2420	2016	1613	1344	3226	2689
110	0.5735	2349	1957	1566	1305	3132	2610
120	0.5562	2278	1898	1518	1265	3037	2531

Note:

1. Values of 4096 do not correspond to a temperature, rather this indicates the ADC at full-scale.

3.6. Output Format

The output can be provided as either straight binary or 2's complement the Analog Input signal conversion to a digital code in bipolar mode is shown in Figure 3.4 and unipolar mode is shown in Figure 3.5.

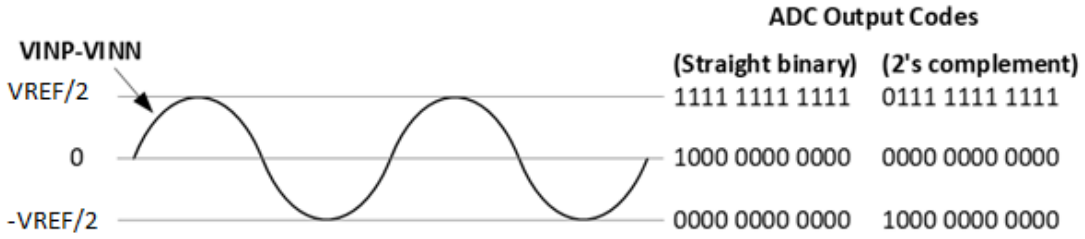

Figure 3.4. Bipolar Input Signal and Output Code Sample

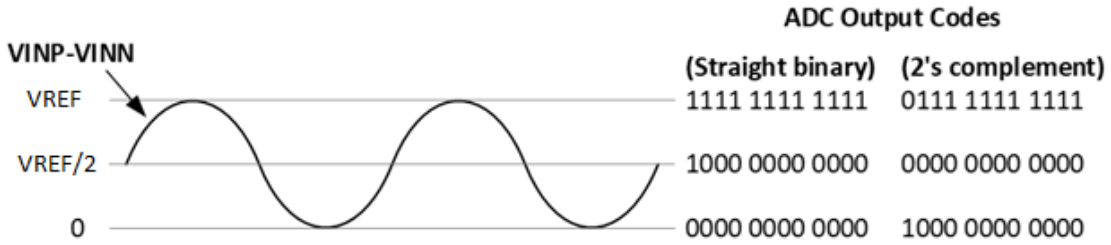

Figure 3.5. Unipolar Input Signal and Output Code Sample

3.7. ADC Conversion and Timing

The ADC samples its input signal when adc_soc_i = 1. While the conversion is ongoing, adc_cog_o = 1. When the conversion is complete, adc_eoc_o = 1 and new data appear on adc0_o and adc1_o. All these signals are in the fab_clk_i domain.

Both ADC 0 and ADC 1 are sampled synchronously. This is shown in Figure 3.6. While the adc_soc_i is high, the ADC is sampling the input. Then adc_cog_o becomes 1 and the ADC is processing the input. The output is latched when adc_eoc_o becomes 1.

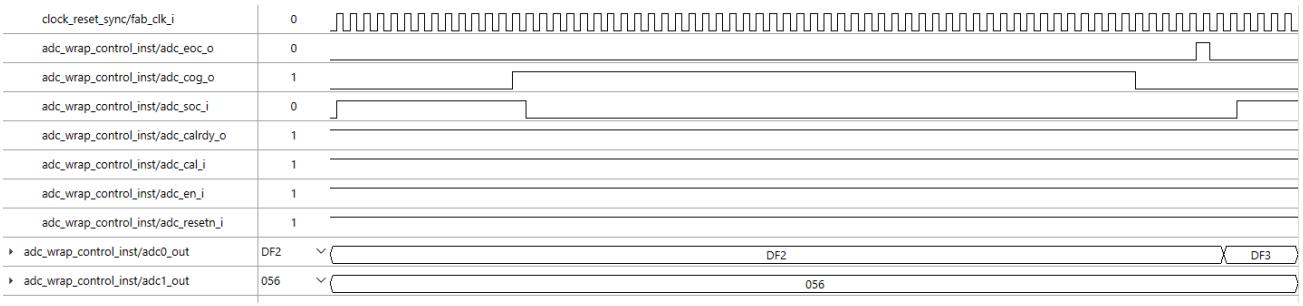

Figure 3.6. Basic ADC Conversion

For proper functioning, adc_soc_i should be made 1 and kept 1 until adc_cog_o becomes 1. If a longer sample is desired, adc_soc_i can remain 1 until the sample period ends. Once adc_cog_o is 1, the control RTL should wait till adc_eoc_o becomes 1 and then latch the output data. This is shown in Figure 3.7.

All ADC control and status signals, such as adc_soc_i, adc_cog_o, and adc_eoc_o, are common to both ADCs. Both ADCs sample together, run their conversion together and complete their conversion together. Both ADCs have to have the same sampling rate.

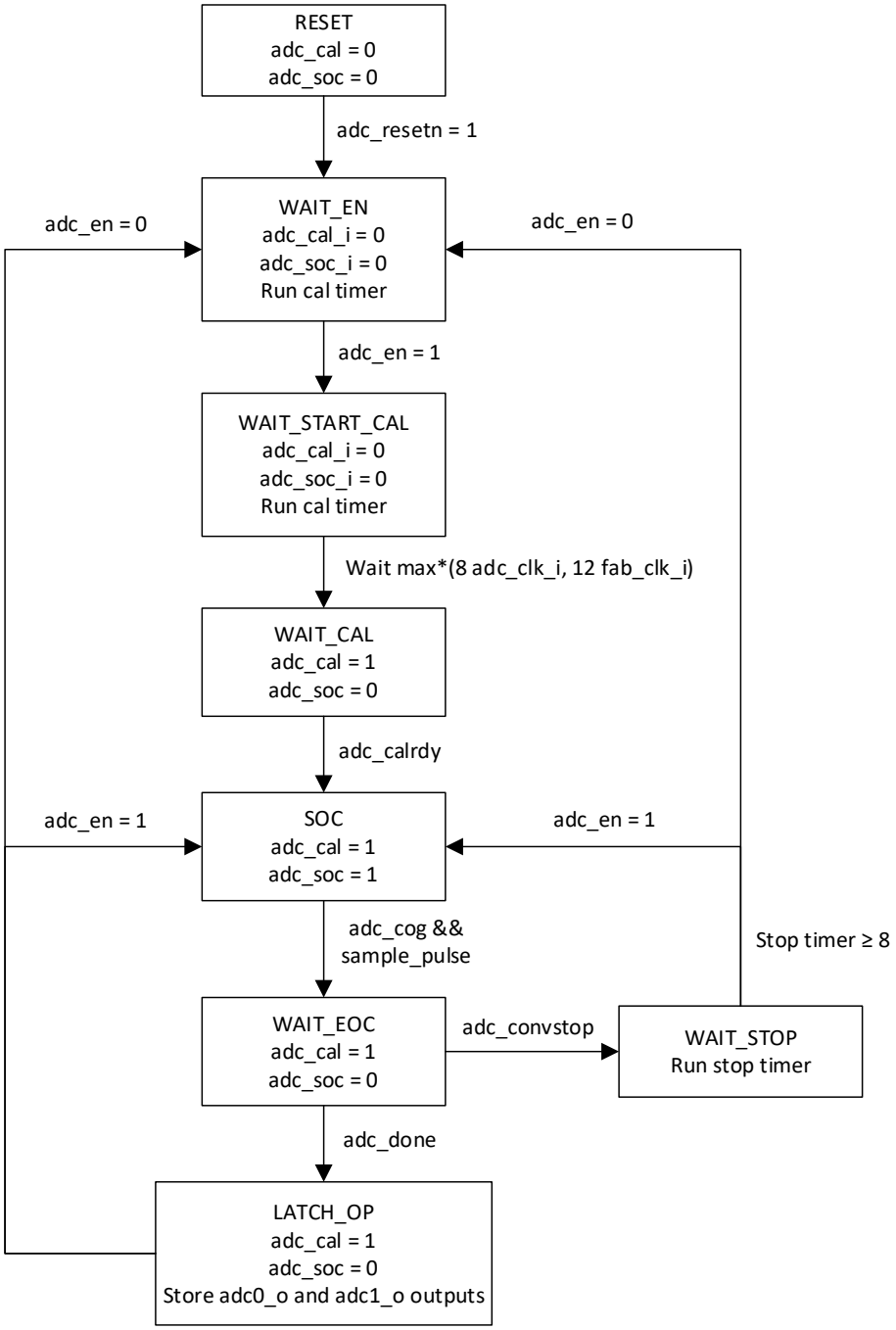

Figure 3.7. ADC Functioning

3.7.1. Stopping and Restarting a Conversion

Normally, a new conversion starts only after an existing conversion is complete. However, if a new sample is required before the existing conversion is complete, you can use the adc_convstop signal which stops the conversion in progress before the EOC pulse comes out. If the control RTL makes the adc_convstop = 1, it should keep adc_convstop at 1 until adc_cog_o becomes 0, wait for 8 fab_clk_i cycles, and then give a new SOC pulse.

In Figure 3.8, the first conversion (left of cursor) starts normally with adc_soc_i but at the cursor adc_convstop becomes 1 and the conversion stops. The signal adc_cog_o becomes 0, but adc_eoc_o does not become 1 as the conversion did not end normally. The output data adc0_dout and adc1_dout are not updated. The conversion is restarted by the control RTL in the FPGA. The next conversion ends normally.

Due to internal clock domain crossings, the ADC might hang if the adc_soc_i pulse is given as soon as adc_cog_o becomes 0. After the adc_cog_o becomes 0, the control logic should wait for 8 fab_clk_i cycles before giving a new adc_soc_i pulse.

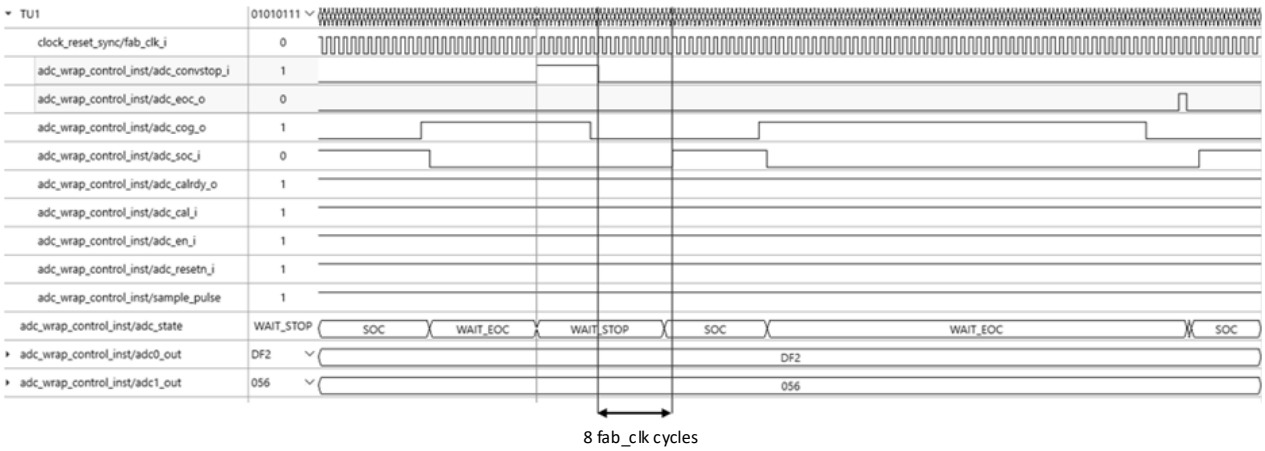

Figure 3.8. ADC Conversion Stop and Restart

3.7.2. Comparators

There are three continuous-time comparators. The comparator inputs are selected in the design between ADC channel inputs (in parallel) or separate comparator inputs. Each comparator has two outputs, a continuous output and a latched output. The latched output is synchronized to fab_clk_i and cleared by adc_resetn_i. When the latched comparator outputs are cleared with adc_resetn_i, the ADC calibration will also be reset.

3.7.3. Controlling Sampling Frequency

The adc_clk_i and fab_clk_i can be asynchronous. If they are asynchronous, then signals going between domains have a variable latency. However, you can derive formulae that give the maximum latency or the minimum sampling frequency based on adc_clk_i and fab_cl_i k frequency. In the formulae, SOC time T_{SOC} is the time from the SOC going high to the time adc_soc_i goes 0 in response to adc_cog_o becoming 1. The EOC time T_{EOC} is the time from adc_cog_o becoming 1 and adc_eoc_o becoming 1. These are shown in Figure 3.9.

$$T_{SOC} = 10T_{ADCCLK} * CLKDIV + 6T_{FABCLK} \qquad \boldsymbol{Eq.3.}$$

$$T_{EOC} = 34T_{ADCCLK} * CLKDIV + 16T_{FABCLK} \qquad \boldsymbol{Eq.4.}$$

Where:

- T_{ADCCLK} is the period of adc_clk_i
- T_{FABCLK} is the period of fab_clk_i.
- The sampling period of the ADC is the sum of T_{SOC} and T_{EOC}.
- CLKDIV is the value by which the adc_clk_i is internally divided, as shown in Figure 3.10.

The values listed above are a floor on the ADC sampling frequency, and the ADC sampling frequency can exceed the value predicted by the equations above.

If a lower sampling frequency is required, then adc_soc_i can be kept 1 even after adc_cog_o becomes 1 to increase the sampling period. If adc_clk_i and fab_clk_i are of different frequencies or have a variable phase shift, the sampling period changes slightly due to the latency of signals going between the clock domains. If a constant sampling frequency is required (one such use case is spectral analysis of an input signal), adc_clk_i and fab_clk_i can be made synchronous and come from lower right PLL.

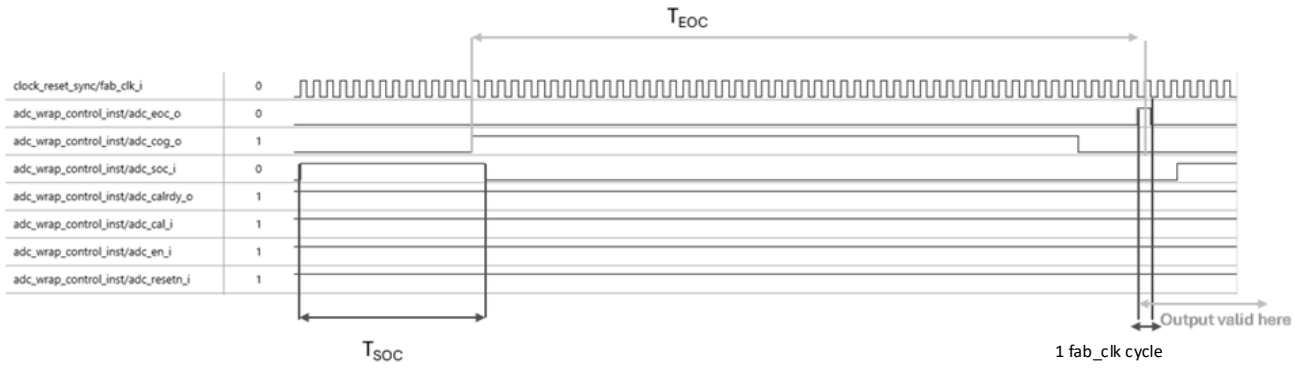

Figure 3.9. ADC Timing Information

3.7.4. Maximum Sampling Frequency

To get the maximum sampling frequency, select fab_clk_i as 40 MHz and the internal ADC clock as 25 MHz, resulting in a sampling frequency of 623 ksps. The internal ADC clock is the adc_clk_i divided by the internal ADC clock divider. This can be achieved by setting adc_clk_i as 50 MHz and clock divider as 2, or adc_clk_i as 100 MHz and clock divider as 4 and so on. The ADC has less jitter if adc_clk_i is higher frequency and divided by a larger number.

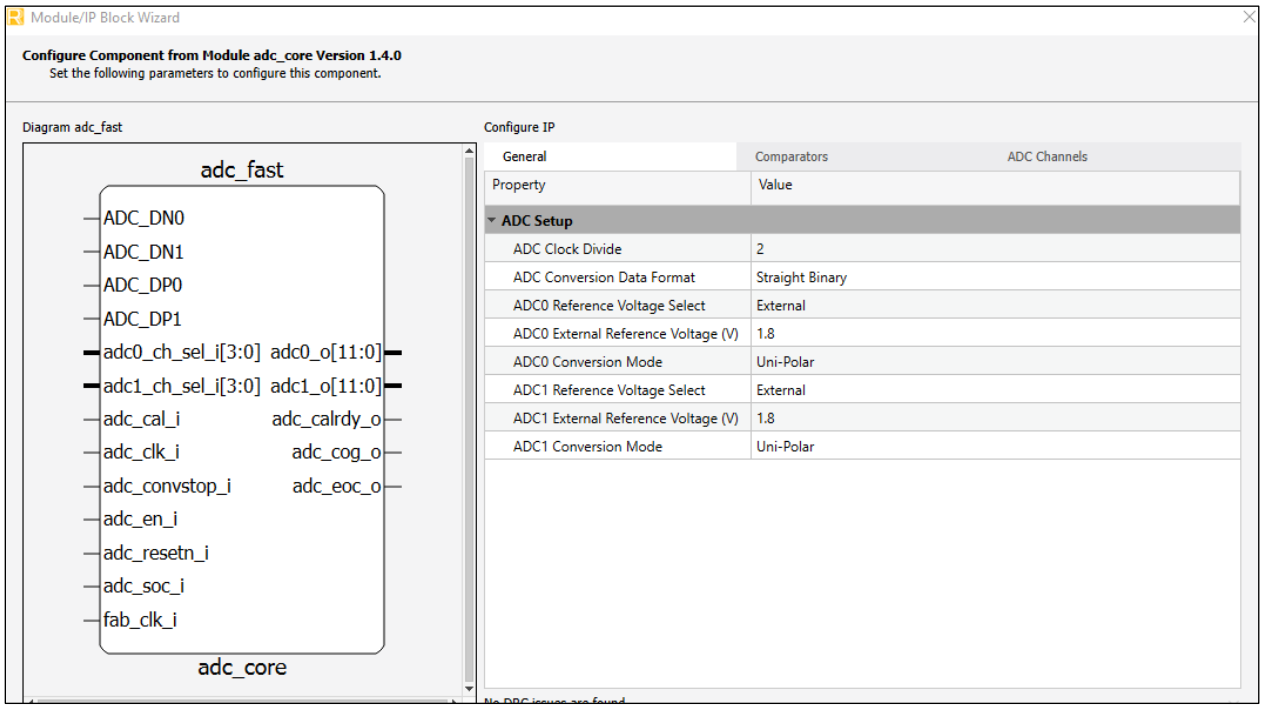

Figure 3.10. ADC Configuration Options

3.7.5. End of Conversion Timeout

If the conversion does not complete and adc_eoc_o pulse does not occur at the expected time due to timing violations in the ADC HIP, the RTL can implement a timeout mechanism to issue a new adc_soc_i pulse and restart the conversion. This approach is recommended for safety-critical applications. Refer to the ADC Demo Design on CertusPro-NX Versa Board reference design for more information.

4. Hardware Considerations

4.1. Supply Decoupling

The ADCs are powered by two supplies: one for the analog circuitry ($V_{CC\_ADC\_18}$) and one for the digital circuitry (V_{CC_ADC}). It is a good practice to decouple the digital noise from the analog circuits, when possible. Figure 4.1 shows an example of how to isolate the digital power and ground from the analog power and ground. The inductor L_1 should be a Ferrite bead with a minimum impedance of 150 Ω at 1 MHz. The bulk capacitor C_1 should be in the range of 2 to 3 µF. The decoupling capacitors (C_2, C_3, and C_4) should be typical 0.1 µF. A single zero-Ω resistor (R_G) is used to connect the digital ground plane to the analog ground plane. Not all Nexus packages have separate connections for the analog power and ground.

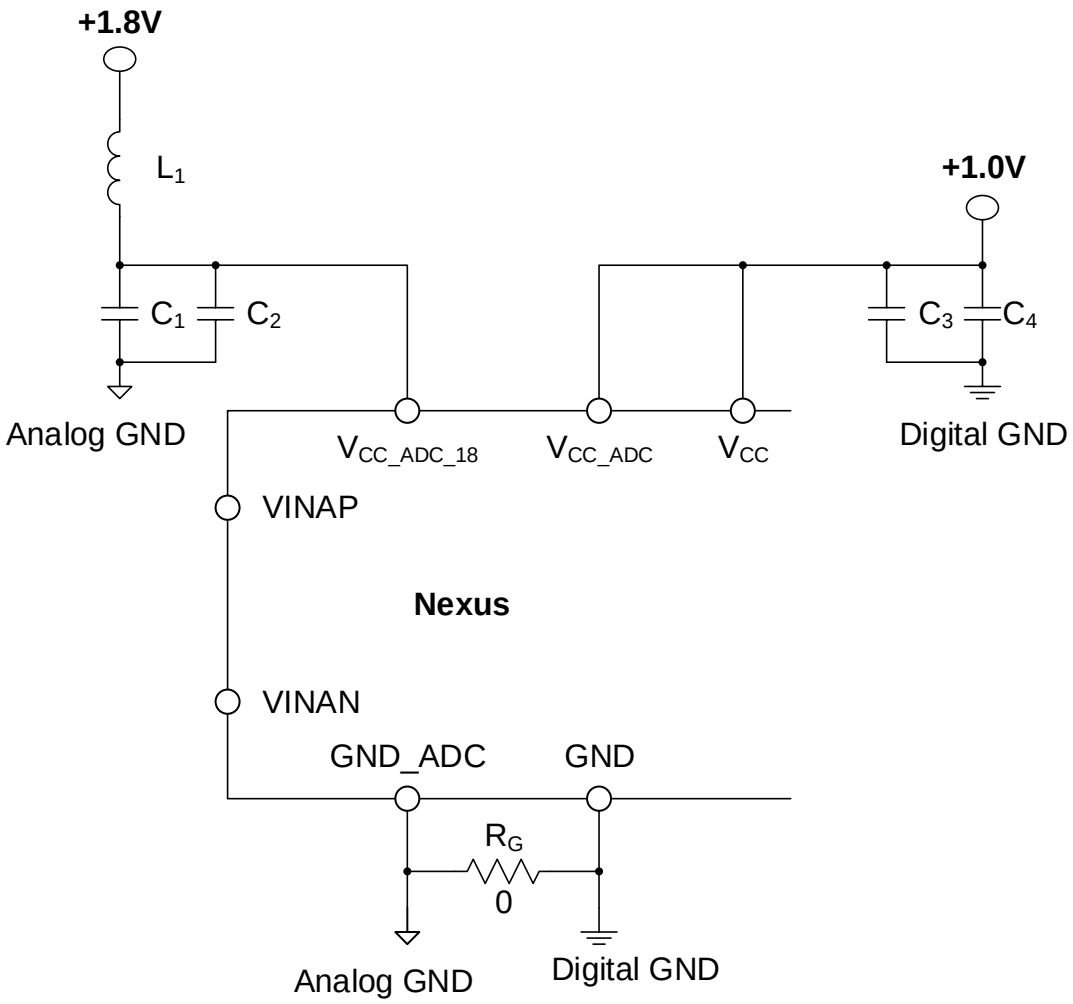

Figure 4.1. ADC Power Supply Decoupling

4.2. Anti-Aliasing Input Filtering

When an ADC is used to sample real time data, it is important to limit the frequency content of the input signal in order to maintain fidelity of the signal that is being processed. Typically, a low-pass filter is used with corner frequency (f_c) set to the Nyquist frequency which is half the sampling frequency (f_s). Signal processing applications often utilize a multiple order low-pass filter (such as a fifth order or more) to effectively block frequency content above f_c. The discussion of such filters is beyond the scope of this technical note. Instead, a single-order resistor-capacitor (RC) low-pass filter is discussed. RC low-pass filters attenuate frequency content above f_c at the rate of -20 dB/decade as shown in Figure 4.2.

A differential RC filter is shown in Figure 4.3 for one of the ADCs supported by Nexus devices. The corner frequency for the RC filter is given by Equation 5.

$$f_c = \frac{1}{2\pi(R_{F1} + R_{F2})C_F} \qquad Eq. 5.$$

Where the f_c is in Hertz, R_{F1} and R_{F2} are in Ω, and C_F is in Farads. At the corner frequency, the amplitude of the input signal is attenuated by –3 dB. Table 4.1 provides some example values for typical sampling frequencies.

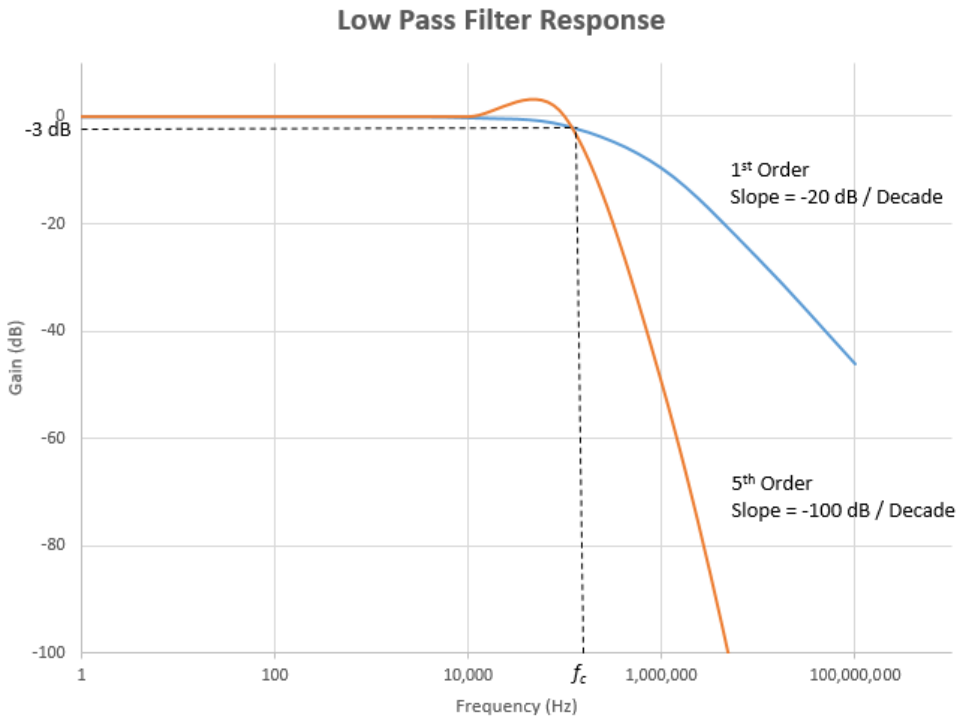

Figure 4.2. Low Pass Filter Response

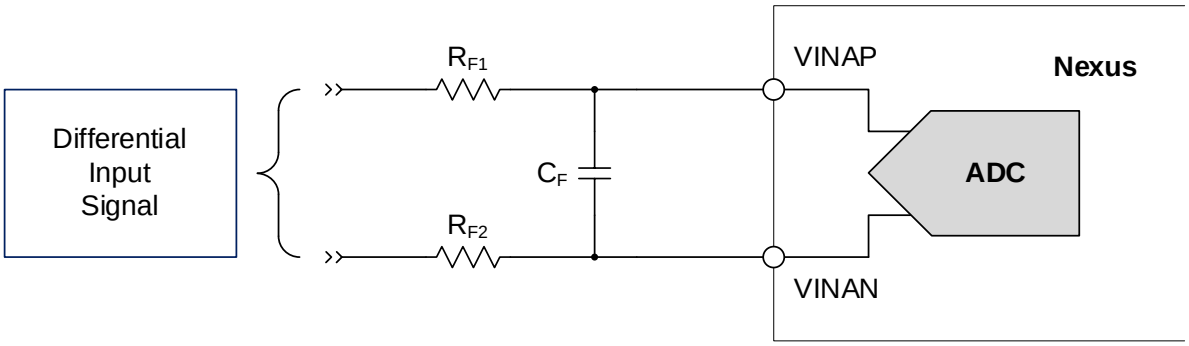

Figure 4.3. Differential ADC Input Filter

Table 4.1. RC Values for Differential Low Pass Filter

$R_{F1} + R_{F2}$ (Ω)	C_F (pF)	F_c (kHz)	f_s (MHz)
200	1,600	500	1.0
200	3,200	250	0.5
720	2,200	100	0.2

The ADCs in Nexus devices can also operate in single-ended input mode. Figure 4.4 shows an example of how a single-ended RC low-pass filter can be connected to the ADCs within Nexus devices. The values in Table 4.1 can also be used for the single-ended filter by setting the value of R_{F2} to zero.

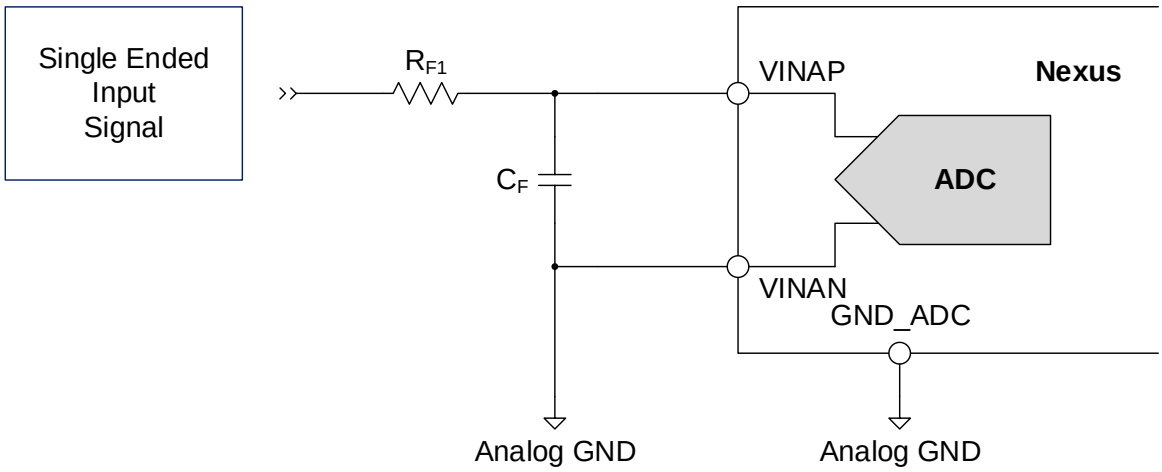

Figure 4.4. Single Ended ADC Input Filter

4.3. External Voltage Reference

An external voltage reference is needed for the ADCs to meet the data sheet specifications. Each ADC has a dedicated external voltage reference pin. These pins can be connected to a single reference or to separate references. The reference voltage source should be accurate to 0.2% over temperature and supply voltage. The external voltage reference should be connected to the analog ground. The supply for the external reference should be isolated from the digital supply using a Ferrite bead and bypass capacitors and/or a separate supply.

The ADC reference voltage must be higher than or equal to any voltage applied to any of the channel inputs; this includes the divided-down power supply sensor channels. If the reference voltage is as little as 0.1 V lower than an applied input, then the ADC result from all channels are unreliable and the ADC circuitry may be damaged.

4.4. Internal Voltage Reference

The internal voltage reference has up to 20% error and may not be fully tested in production, as it is used in internal testing. Use of internal reference is not recommended for accurate measurement. Customers are advised to get an external reference voltage. If the internal reference is used, then VCCIO0, VCCIO1, and VCCAUX should be 1.2 V or less.

4.5. ADC Error Calculation

The ADC error calculation consists of the following two parts:

- Reference error, which describes how far the reference is from the nominal value.
- Conversion error, detailed in Table 4.2, that shows errors that accumulate during the conversion process. One inevitable conversion error that always occur is the quantization error, but this is absorbed into the DNL and INL.

Table 4.2. ADC Conversion Errors

Error type	Error value	Error Unit	Description	Note
Differential non-linearity (DNL)	±0.9	LSB	An ideal ADC exhibits code transitions that are exactly 1 LSB apart. DNL is the deviation from this ideal value. A DNL error specification of less than 1 LSB guarantees no missing codes and a monotonic transfer function.	No missing codes. This only affects dynamic error but not static error. For a DC input, any error caused by it is absorbed into INL.

Error type	Error value	Error Unit	Description	Note
Integral non-linearity (INL)	±1.5	LSB	INL refers to the deviation of each individual code from a line drawn from negative full scale through positive full scale. The point used as negative full scale occurs at 0.5 LSB before the first code transition. Positive full scale is defined at a level of 1.5 LSB beyond the last code transition. The deviation is measured from the middle of each particular code to the true straight line.	This is a static error that can occur even with a constant input.
Gain error (E_G)	±1.0%	Full scale	The gain error is the difference in slope between the ideal transfer curve and the measured transfer curve. It is proportional to the input voltage. For more details, refer to Figure 4.5.	The worst gain error is 1% of full scale, at lower V_{IN} the gain error is reduced.
Input referred offset error (E_O)	±1.0%	Full scale	Offset error in the comparator used in the ADC. It is independent of the input voltage (simplified model). For more details, refer to Figure 4.6.	—

For a DC input, the worst-case ADC conversion error is approximately 2% of the full scale. The error in the reference must be added to get the total ADC error. However, this is relevant only for a static DC input at full scale, which is the worst possible case of error.

When measuring a voltage expected to be in the middle of the ADC's full-scale range, the gain error can be halved since it is proportional to the input voltage. Hence, the total error is reduced.

The gain error can be calculated using the following equation:

$$E_G = EGFS \times \frac{V_{IN}}{VREF} \qquad \textbf{Eq. 6.}$$

Where:

- E_G is the gain error at a given voltage V_{IN}.
- EGFS is the gain error at full scale.

The offset error is common to all measurements done by the ADC. All measurements done by ADC 0 have the same offset error, as do all measurements done by ADC1. If 2 measurements from the same ADC are to be compared, then the offset error can be ignored. This alone will reduce the total error by 1%. All measurements intended for comparison should be taken using the same ADC.

If an AC measurement is needed, then only the difference between successive samples matters. Hence, the main parameter of interest is DNL, which is the difference between the input voltages for successive output codes. If the DNL is 0 and when there is a change in the ADC code by 1 LSB, you can assume that the input voltage has changed by VREF/2N, where N is the number of bits. If the DNL is not 0 and when there is a change in the ADC code by 1 LSB, you can assume that the input voltage has changed to at least (1 – DNL) × VREF/2N and at most (1 + DNL) × VREF/2N.

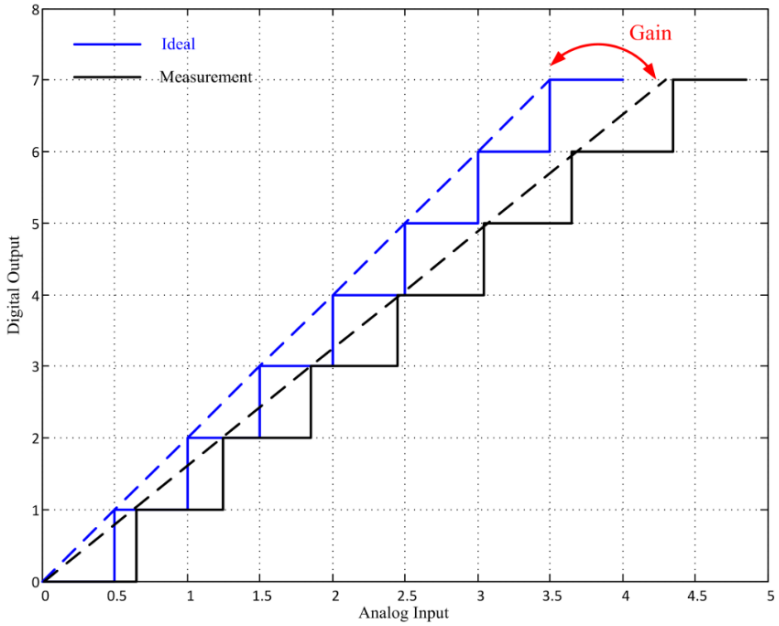

Figure 4.5. Gain Error

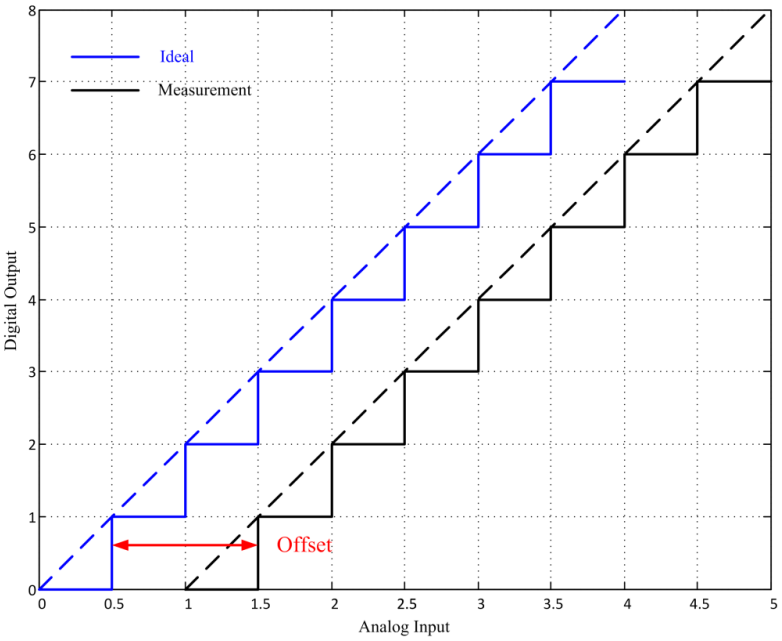

Figure 4.6. Offset Error

The INL, gain error and offset error are in addition to the error introduced by the reference voltage. The on-die reference voltage has about 10% error, so an external reference voltage is recommended for accurate calculations. The DNL is not affected by errors in the reference voltage.

4.6. Drive Signal to ADC Channel Before Device Power Up

In general, the analog inputs of the ADC, such as reference or dedicated ADC inputs, should not have any voltage on them if the ADC supply (VCCADC1V8 or VCCAUX) is off. During power on and power off transient conditions, the input signal and the ADC supply can ramp at approximately the same time.

5. ADC Instantiation

All information in this section is for the Lattice Radiant™ software version 3.0 and newer. It is recommended to upgrade to the latest version of the Radiant software to get the latest ADC.

5.1. IP Catalog in Lattice Radiant Software

The ADC can be instantiated from the Radiant software using the IP Catalog, and selecting ADC under architecture modules, as shown in Figure 5.1 and Figure 5.2. Do not select the ADC sequencer module.

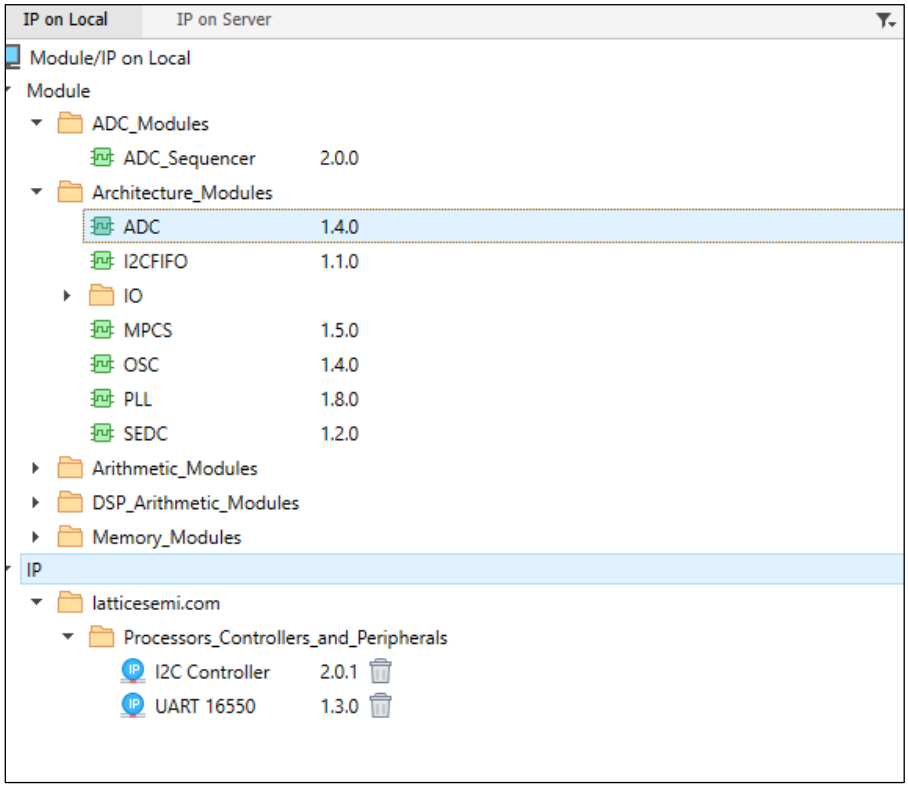

Figure 5.1. Selecting ADC from IP Catalog in Lattice Radiant Software 2023.2 or Older

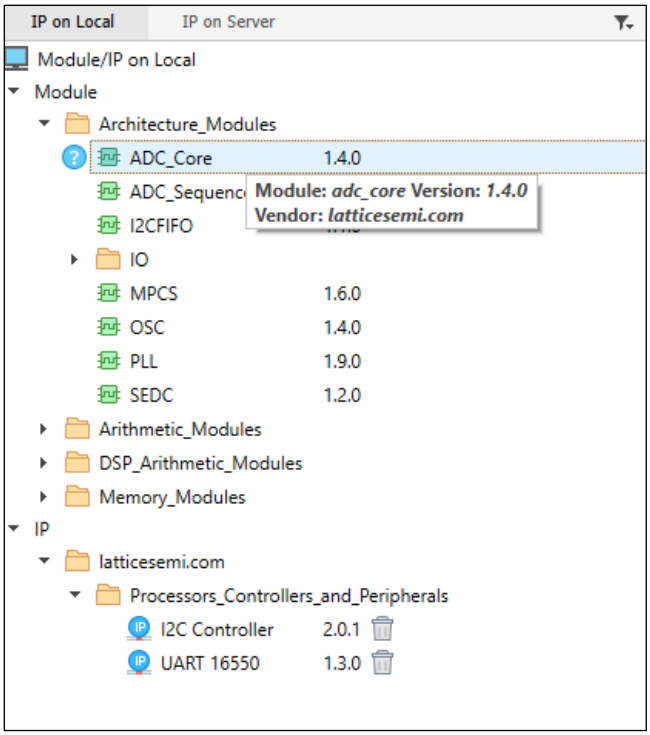

Figure 5.2. Selecting ADC from IP Catalog in Lattice Radiant Software 2024

5.2. ADC Options

It is recommended to configure all options using the IP catalog. It is not recommended to edit the RTL parameters, which was the procedure for older versions of the Radiant software. The ADC options are listed in the table below.

Table 5.1. ADC Options in Lattice Radiant Software

Option	Description
ADC clock divide	Divide adc_clk_i by this number to get the clock used for sampling and successive approximations in the ADC. This number should be an even number and the smallest value is 2.
ADC conversion data format	Output data can be either: • Straight binary (from 0 to 12'hFFF) or • 2's complement. For more details, refer to the Output Format section.
ADC0 reference voltage select	Either internal (up to 20% error) or external (recommended).
ADC0 reference voltage	Choose from 1.8 V to 1.2 V. Reference voltage has to be greater than the highest input voltage.
ADC0 conversion mode	Unipolar conversion mode: DP will always be greater than DN. ADC will be 0 when DP = DN, and at full scale when DP – DN = reference voltage. Always use unipolar mode when measuring supply voltages, which are scaled by 2.5 before being given as a unipolar input. Bi-polar conversion mode: DP can be greater or lesser than DN as long as abs(DP – DN) < Vref and 0 < DP < VRref and 0 < DN < VRref.
ADC1 reference voltage select	Either internal (up to 20% error) or external (recommended).
ADC1 reference voltage	Choose from 1.8 V to 1.2 V. Reference voltage has to be greater than the highest input voltage.
ADC1 conversion mode	Unipolar conversion mode: DP will always be greater than DN. ADC will be 0 when DP = DN, and at full scale when DP – DN = reference voltage. Always use unipolar mode when measuring supply voltages, which are scaled by 2.5 before being given as a unipolar input. Bi-polar conversion mode: DP can be greater or lesser than DN as long as abs(DP – DN) < Vref and 0 < DP < VRref and 0 < DN < VRref.

Some combinations of the FPGA device and package have comparators enabled. If a device is marked with a *Y* in the ADC_C column of Table A.1, it indicates that the device has comparators available, which are connected as follows:

Table 5.2. Comparator Connections to ADC Inputs 1

Comparator	P input	N input
Comp1	ADC_CP0	ADC_CN0
Comp2	ADC_CP1	ADC_CN1
Comp3	ADC_CP2	ADC_CN2

If the *COMP* column is also marked with a *Y*, then the comparators have 2 possible sets of inputs that are selectable during configuration using the configuration tab.

Table 5.3. Comparator Connections to ADC Inputs 2

Comparator	Option 1		Option 2	
	P input	N input	P input	N input
Comp1	ADC_CP0	ADC_CN0	COMP1IP	COMP1IN
Comp2	ADC_CP1	ADC_CN1	COMP2IP	COMP2IN
Comp3	ADC_CP2	ADC_CN2	COMP3IP	COMP3IN

Each comparator has to be enabled during configuration using the checkbox function.

Appendix A. Package Options

The supported ADC features are based on the device family, package, and speed grade as defined in Table A.1.

Table A.1. ADC Core Module Features versus Device and Package

Device[1]	Package	COMP[2]	ADC_D[3]	ADC_C[4]	Ext VREF	ADC Supply
LIFCL-17 CrossLink-NX	CABGA256	N	Y	List 4	Y	VCCAUX
	CSFBGA121	N	N	List 4	N	VCCAUX
	QFN72	N	N	List 5	N	VCCADC18
	WLCSP72	N	N	List 5	N	VCCAUX
LIFCL-33 CrossLink-NX	ADC absent					
LIFCL-33U CrossLinkU-NX	ADC absent					
LIFCL-40 CrossLink-NX	CABGA400	Y	Y	Y	Y	VCCADC18
	CSBGA289	Y	Y	Y	Y	VCCADC18
	CABGA256	Y	Y	Y	Y	VCCADC18
	CSFBGA121	N	N	List 1	N	VCCAUX
	QFN72	N	N	List 2	N	VCCADC18
LFD2NX-9 Certus-NX	CSFBGA121	N	Y	List 3	Y	VCCADC18
	CABGA196	N	Y	List 3	Y	VCCADC18
LFD2NX-15 Certus-NX	CABGA256	Y	Y	List 8	Y	VCCADC18
	CABGA400	Y	Y	Y	Y	VCCADC18
LFD2NX-17 Certus-NX	CSFBGA121	N	Y	List 3	Y	VCCADC18
	CABGA196	N	Y	List 3	Y	VCCADC18
LFD2NX-25 Certus-NX	CABGA256	Y	Y	List 8	Y	VCCADC18
	CABGA400	Y	Y	Y	Y	VCCADC18
LFD2NX-28 Certus-NX	CABGA256	Y	Y	Y	Y	VCCADC18
	CABGA196	Y	Y	Y	Y	VCCADC18
	CSFBGA121	Y	N	Y	N	VCCAUX
LFD2NX-35 Certus-NX	BBG484	Y	Y	Y	Y	VCCADC18
	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	List 7	Y	VCCADC18
LFD2NX-40 Certus-NX	CABGA256	Y	Y	Y	Y	VCCADC18
	CABGA196	Y	Y	Y	Y	VCCADC18
	CSFBGA121	Y	N	Y	N	VCCAUX
LFD2NX-65 Certus-NX	BBG484	Y	Y	Y	Y	VCCADC18
	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	List 7	Y	VCCADC18
LFCPNX-50 CertusPro-NX	BBG484	Y	Y	Y	Y	VCCADC18
	BFG484	Y	Y	Y	Y	VCCADC18
	ASG256	Y	Y	Y	Y	VCCADC18
	CBG256	Y	Y	Y	Y	VCCADC18
LFCPNX-100 CertusPro-NX	LFG672	Y	Y	Y	Y	VCCADC18
	BBG484	Y	Y	Y	Y	VCCADC18
	ASG256	Y	Y	Y	Y	VCCADC18
	CBG256	Y	Y	Y	Y	VCCADC18
	BFG484	Y	Y	Y	Y	VCCADC18
LFMXO5-15D MachXO5-NX	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	List 6	Y	VCCADC18

Device[1]	Package	COMP[2]	ADC_D[3]	ADC_C[4]	Ext VREF	ADC Supply
LFMXO5-25 MachXO5-NX	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	Y	Y	VCCADC18
LFMXO5-35/T MachXO5-NX	BBG484	Y	Y	Y	Y	VCCADC18
	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	List 7	Y	VCCADC18
LFMXO5-55T MachXO5-NX	BBG400	Y	Y	Y	Y	VCCADC18
LFMXO5-55TD MachXO5-NX	BBG400	Y	Y	Y	Y	VCCADC18
LFMXO5-65/T MachXO5-NX	BBG484	Y	Y	Y	Y	VCCADC18
	BBG400	Y	Y	Y	Y	VCCADC18
	BBG256	Y	Y	List 7	Y	VCCADC18
LFMXO5-100T MachXO5-NX	BBG400	Y	Y	Y	Y	VCCADC18

Notes:

1. The ADC is only supported in the Automotive devices (all speed grades) and the higher speed grades of the Commercial/Industrial devices (–8 and –9).
2. All three Comparator external inputs available (COMP1, COMP2, and COMP3). If this column is marked with an *N*, but the *ADC_C* column indicates that the ADC input channels 0, 1, 2, and 3 are present, you can still use the comparator by using their inputs.
3. Both dedicated external inputs available (ADC_D0 and ADC_D1).
4. All 16 dual-function external inputs available (ADC_C0 – ADC_C15).

List 1: Channels 0, 1, 2, 3, 5, 6, 7, 8, 9, 13, and 14.

List 2: Channels 0, 1, 2, 3, 5, 6, 8, 13, and 14.

List 3: Channels 0, 1, 2, 3, 5, 6, 7, 8, 9, 13, and 14.

List 4: Channels 0, 1, 2, 3, 5, 6, 7, 8, 9, 13, and 14.

List 5: Channels 0, 1, 2, 3, 5, 6, 8, 13, and 14.

List 6: Channels 0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, and 13.

List 7: Channels 0, 1, 2, 3, 4, 5, 8, 10, and 13.

List 8: Channels 0, 2, 5, 6, 7, 10, 11, and 12.

References

For more info on this FPGA device, refer to the following:

- Certus-NX Family Datasheet (FPGA-DS-02078)
- CertusPro-NX Family Datasheet (FPGA-DS-02086)
- CrossLink-NX Family Datasheet (FPGA-DS-02049)
- MachXO5-NX Family Datasheet (FPGA-DS-02102)
- Certus-NX Hardware Checklist (FPGA-TN-02151)
- CertusPro-NX Hardware Checklist (FPGA-TN-02255)
- CrossLink-NX Hardware Checklist (FPGA-TN-02149)
- MachXO5-NX Hardware Checklist (FPGA-TN-02274)
- sysCLOCK PLL Design and User Guide for Nexus Platform (FPGA-TN-02095)
- ADC Core Module – Lattice Radiant Software User Guide (FPGA-IPUG-02168)
- Debugging with Reveal Usage Guidelines and Tips (FPGA-AN-02060)
- ADC Demo Design on CertusPro-NX Versa Board User Guide (FPGA-EB-02061)
- ADC Demo Design on CertusPro-NX Versa Board web page
- CrossLink-NX web page
- Certus-NX web page
- CertusPro-NX web page
- MachXO5-NX web page
- Lattice Radiant Software web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

For technical support or additional information regarding ADC clock handling, simulation and hardware implementation, submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 2.0, July 2025

Section	Change Summary
Functional Overview	Added the End of Conversion Timeout section.

Appendix A. Package Options	• Updated Table A.1. ADC Core Module Features versus Device and Package:
	  • Added the *LFD2NX-15*, *LFD2NX-25*, *LFD2NX-35*, *LFD2NX-65*, *LFMXO5-35/T*, and *LFMXO5-65/T* devices.
	  • Reordered the list of devices in ascending alphabetical order.
	• Added channels for *List 7* and *List 8*.

References	Added the *ADC Demo Design on CertusPro-NX Versa Board* web page.

Revision 1.9, February 2025

Section	Change Summary
Acronyms in This Document	Added *Graphical User Interface* (*GUI*) and *Intellectual Property* (*IP*).
ADC Module Description	• Updated Figure 2.4. Comparator 1 Block Diagram. • Updated the Comparators section.

Revision 1.8, November 2024

Section	Change Summary
All	• Made editorial fixes. • Removed the *ADC Primitive* section.
Disclaimers	Updated boilerplate.
Inclusive Language	Added boilerplate.
Acronyms in This Document	Removed *Mega Samples per Second (MSPS)* and added the following acronyms: • *Differential Non-Linearity (DNL)* • *Input/Output (I/O)* • *Integral Non-Linearity (INL)* • *Kilo Samples Per Second (KSPS)* • *Least Significant Bit (LSB)* • *Register Transfer Level (RTL)*
Introduction	• Updated the Overview section. • Updated the *conversion rate* value and *selectable input signal* descriptions in the Features section.

ADC Module Description	• Renamed this section from *Functional Description* to *ADC Module Description* and updated the contents of the following subsections:
	  • Overview
	  • ADC Cores
	  • Calibration Controller
	  • Voltage Reference
	  • Signal Description
	• Updated Figure 2.1. ADC Core Module Control and Status Block and Figure 2.3. ADC Core Module ADC 1 Block.
	• In Table 2.1, updated the details for the following signals: *adc_clk_i*, *fab_clk_i*, *adc_resetn_i*, *adc_soc_i*, *adc_eoc_o*, *adc_cog_o*, *adc_convstop_i*, *adc_cal_i*, *adc_calrdy_o*, *comp1_o*, *comp1l_o*, *comp2_o*, *comp2l_o*, *comp3_o*, and *comp3l_o*.

Functional Overview	• Added the Introduction section. • Added *and Reset* to ADC Enable and Reset. • Added *Analog* to ADC Analog Input Selection. • Added *and Timing* to ADC Conversion and Timing.

Section	Change Summary

	• Updated the contents of the following subsections:
	  • ADC Enable and Reset
	  • Calibration
	  • Reference Voltage
	  • ADC Analog Input Selection
	  • Power Supply Sensor
	  • DTR (Digital Temperature Readout)
	  • ADC Conversion and Timing
	• Added the ADC Voltage Scaling section.
	• Added Table 3.2. ADC 0 Inputs and Table 3.3. ADC 1 Inputs.
	• Updated Figure 3.1. ADC Timing Diagram: Calibration Starting After Reset and added Figure 3.2. ADC Calibration Ending and Conversion Starting (After 5500 adc_clk_i Cycles).
	• Replaced all subsections in the ADC Conversion and Timing section with new subsections.

Hardware Considerations	• Updated the minimum impedance value in the Supply Decoupling section. • Updated the External Voltage Reference section. • Added the Internal Voltage Reference, ADC Error Calculation, and Drive Signal to ADC Channel Before Device Power Up subsections.
ADC Instantiation	Added this section.

Appendix A. Package Options	• Renamed this section from *Limitations* to *Package Options* and updated its contents.
	• Updated Table A.1. ADC Core Module Features versus Device and Package:
	  • Updated details for the *LIFCL-17*, *LFD2NX-17*, *LFD2NX-17*, and *LFD2NX-9* devices.
	  • Added the *LIFCL33*, *LIFCL33U*, *LFD2NX-9*, *LFD2NX-28*, *LFCPNX-50*, *LFMXO5-100T*, *LFMXO5-15D*, *LFMXO5-55T*, and *LFMXO5-55TD* devices.
	  • Updated *Notes*.
	• Updated channels for *List 1* and *List 2*.
	• Added channels for *List 3*, *List 4*, *List 5*, and *List 6.*

References	• Added the following references:
	  • *Certus-NX Hardware Checklist (FPGA-TN-02151)*
	  • *CertusPro-NX Hardware Checklist (FPGA-TN-02255)*
	  • *CrossLink-NX Hardware Checklist (FPGA-TN-02149)*
	  • *MachXO5-NX Hardware Checklist (FPGA-TN-02274)*
	  • *Debugging with Reveal Usage Guidelines and Tips (FPGA-AN-02060)*
	  • *ADC Demo Design on CertusPro-NX Versa Board User Guide (FPGA-EB-02061)*
	• Removed the *Lattice Radiant Software 2.0 User Guide*.

Revision 1.7, August 2023

Section	Change Summary
Functional Description	• Updated naming convention of the ADC input channels in Figure 2 2. ADC Core Module ADC0 Block and Figure 2 3. ADC Core Module ADC1 Block. • Updated ADC ports in Table 2 1. ADC Module Ports.
Appendix B. Limitations	Added CBG256 and BFG484 packages for the LFCPNX-100 device in Table B.5 2. ADC Core Module Features versus Device and Package.
Technical Support Assistance	Replaced text *For technical support or for additional information regarding security, lock policy settings, and authentication commands* with *For technical support or additional information regarding ADC clock handling, simulation and hardware implementation*.

Revision 1.6, March 2023

Section	Change Summary
Functional Description	• Updated Figure 2.3. ADC Core Module ADC1 Block to interchange position of VCCIO0 and VCCIO1. • Added another paragraph for ADC reference voltage regarding higher or equal voltage applied in Voltage Reference section.
Functional Overview	Added another paragraph for ADC reference voltage regarding higher or equal voltage applied in Reference Voltage section.
Hardware Considerations	Added another paragraph for ADC reference voltage regarding higher or equal voltage applied in External Voltage Reference section.
Appendix A. ADC Primitive	Updated note in Table A.1. ADC Parameters to add verbiage regarding internal and external voltage reference.
Technical Support Assistance	Added reference link to Lattice Answer Database.

Revision 1.5, August 2022

Section	Change Summary
Functional Description	Updated Figure 2.2. ADC Core Module ADC0 Block exchanging VCC and VCCM positions.

Revision 1.4, March 2022

Section	Change Summary
All	Minor adjustments in formatting across the document, including changing reference document titles with Usage Guide to User Guide.
Acronyms in The Document	Added definitions for CAL, CDC, FIFO, and LRC
Introduction	Updated section content to include MachXO5-NX support.
Functional Description	• Updated overall section content. • Updated Figure 2.1. ADC Core Module Control and Status Block, Figure 2.2. ADC Core Module ADC0 Block, Figure 2.3. ADC Core Module ADC1 Block, and Figure 2.4. Comparator 1 Block Diagram. • Updated overall content of Table 2.1. ADC Module Ports.
Functional Overview	• Updated overall Calibration section content. • Updated overall ADC Conversion section. • Updated Figure 3.1. ADC Timing Diagram: Calibration, Figure 3.6. ADC Timing Diagram: Programmable Input Signal Sample, Figure 3.7. ADC Timing Diagram: Single-Pass Conversion Mode , Figure 3.8. ADC Timing Diagram: Continuous Mode , and Figure 3.9. ADC Timing Diagram: Aborting an Ongoing Conversion.
Appendix B. Limitations	Added this section that defines limitations of ADC and feature availability based on family, package, and speed grade.
References	Added reference to MachXO5-NX datasheet and ADC Core Module Radiant IP User Guide.

Revision 1.3, June 2021

Section	Change Summary
All	Minor adjustments in formatting.
Acronyms in This Document	Removed definition for GPIO.
Introduction	Updated section content to include CertusPro-NX support.
Functional Description	• Updated overall section content. • Updated Figure 2.1 and Figure 2.2. • Updated Table 2.1.
Functional Overview	• Updated overall section content. • Updated Figure 3.2 and Figure 3.5.
Appendix A. ADC Primitive	• Added this section to move Parameters and Instantiation. Moved ADC hardware

Section	Change Summary
	primitive to this section and added documentation of the software generated ADC IP wrapper. • Updated Table A.1 to add table note. • Updated Listing 3 in Instantiation.
References	Added reference to CertusPro-NX datasheet.

Revision 1.2, June 2020

Section	Change Summary
All	• Changed document title from CrossLink-NX ADC Usage Guide to *ADC Usage Guide for Nexus Platform*. • Changed CrossLink-NX to Nexus across the document.
Introduction	Updated Features section.
Functional Description	• Updated Overview and Voltage Reference section. • Updated Table 2.1.
Functional Overview	• Updated ADC Enable, Reference Voltage, and ADC Input Selection section. • Updated Table 3.1.
Parameters	Updated Table 4.1 to add table note.
Instantiation	Updating Listing 3 instantiation.
Hardware Considerations	Added External Voltage Reference section.
References	Updated this section.

Revision 1.1, April 2020

Section	Change Summary
Acronyms in This Document	Updated this section.
Functional Description	• Updated Overview section. • Updated Figure 2.1. ADC Block Diagram. • Updated ADC Cores section. • Updated Comparators section. • Added Figure 2.2. Comparator 1 Block Diagram. • Updated Calibration Controller section. • Updated Table 2.1. ADC Ports.
Functional Overview	• Updated Calibration section and adjusted heading number. • Updated Figure 3.2. Power Supply Voltage Conversion Graph (Vref = 1.2 V). • Updated Table 3.1. DTR Output Code Values as a Function of Temperature and Configuration. • Added Equation 2. • Moved the Power Supply Sensor and the DTR (Digital Temperature Readout) sub-sections under ADC Input Selection.
Hardware Considerations	Added this section.

Revision 1.0, December 2019

Section	Change Summary
All	Initial release

www.latticesemi.com