

Introduction

When digital systems are used in real-world applications, it is often necessary to convert an analog voltage level to a binary number. The value of this number is directly or inversely proportional to the voltage. The analog to digital conversion is realized in the OPB Delta-Sigma ADC (OPB ADC) using Delta-Sigma conversion techniques. This soft IP core is designed to interface with the OPB (On-chip Peripheral Bus).

Features

- 32 bit OPB slave interface
- Supports single beat transactions
- OPB Latency ≤ 3 Clock cycles
- able to operate at OPB Clock frequency ≥ 100 MHz
- 16 entry deep data FIFO
- Selectable ADC resolution

Figure 1 shows how a typical implementation of analog to digital conversion is performed using the OPB ADC. A Delta-Sigma DAC, which is a primary block of the OPB ADC core, is used to generate a reference voltage ADC_{ref} for the negative input to the external comparator.

The analog signal, AnalogIn, feeds the positive input of the comparator. The voltage range of the Delta-Sigma DAC output is 0V to V_{CC0} , where V_{CC0} is the supply voltage applied to the FPGA I/O bank. This is also the range of analog voltage that can be converted.

If the analog input voltage is outside the range 0V to V_{CC0} , either the Delta-Sigma DAC output or the analog signal itself may be biased, attenuated, or amplified with external components to achieve the desired voltage range compatibility.

The analog voltage level is determined by performing a serial binary voltage search, starting at the middle of the voltage range.

LogiCORE™ Facts		
Core Specifics		
Supported Device Family	QPro™-R Virtex™-II, QPro Virtex-II, Spartan™-II, Spartan-II-E, Spartan-3, Spartan-3E, Virtex, Virtex-II, Virtex-II Pro, Virtex-4, Virtex-E	
Version of Core	opb_deltasigma_adc	v1.01a
Resources Used		
	Min	Max
SLICES	191	209
LUTs	199	229
FFs	239	253
Block RAMs	N/A	N/A
Provided with Core		
Documentation	Product Specification	
Design File Formats	VHDL	
Constraints File	N/A	
Verification	N/A	
Instantiation Template	N/A	
Reference Designs		
Design Tool Requirements		
Xilinx Implementation Tools	ISE 6.1i SP1 or later	
Verification	ModelSim SE/EE 5.8e or later	
Simulation	ModelSim SE/EE 5.8e or later	
Synthesis	XST	
Support		
Support provided by Xilinx, Inc.		

Because of the serial nature of both the Delta-Sigma DAC and the analog sampling process, this OPB ADC is useful only on signals that are changing at a lower rate.

If the analog input voltage changes during the sampling process, it effectively causes the sample point to move randomly. This adds a noise component that becomes larger as the input frequency increases. This noise component can be removed with an external sample and hold circuit for the analog input signal.

A 24 mA LVTTTL output buffer is normally used to drive the RC filter. Most comparators have uncommitted collector/drain outputs, so R_p is usually needed.

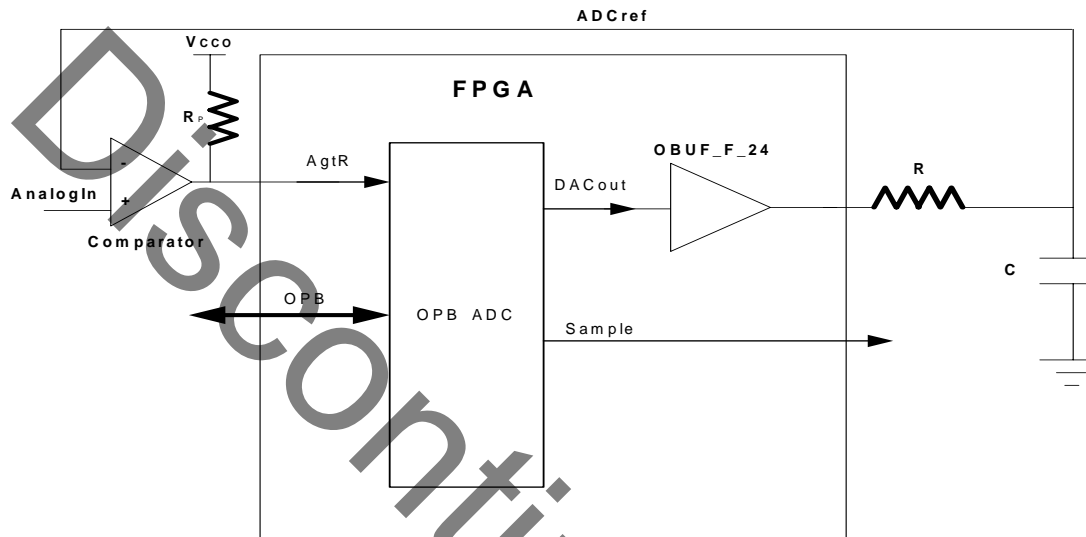


Figure 1: Typical Implementation of Analog to Digital Converter using OPB ADC

OPB ADC Design Parameters

To allow the user to obtain an OPB ADC that is uniquely tailored for their system, certain features are parameterized in the OPB ADC design. This allows the user to have a design that only utilizes the resources required by their system and runs at the best possible performance. The features that can be parameterized in the OPB ADC design are shown in [Table 1](#).

Table 1: OPB ADC Design Parameters

Grouping / Number	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type	
OPB ADC Features	G1	Delta-Sigma DAC input width	C_DACIN_WIDTH	9 or 11	9	Integer
	G2	Filter Settle Time Multiplier(FSTM) width	C_FSTM_WIDTH	4 - 8	4	Integer

Table 1: OPB ADC Design Parameters (Continued)

Grouping / Number	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type	
OPB Interface	G3	OPB Base Address	C_BASEADDR ⁽²⁾	See Allowable Parameter Combinations	None ⁽¹⁾	std_logic_vector
	G4	OPB High Address	C_HIGHADDR ⁽²⁾	See Allowable Parameter Combinations	None ⁽¹⁾	std_logic_vector
	G5	OPB Address Bus Width	C_OPB_AWIDTH	32	32	std_logic_vector
	G6	OPB Data Bus Width	C_OPB_DWIDTH	32	32	std_logic_vector

Notes:

1. No default value will be specified to insure that the actual value is set, i.e. if the value is not set, a compiler error will be generated.
2. C_BASEADDR must be a multiple of the range, where the range is C_HIGHADDR - C_BASEADDR + 1.

Allowable Parameter Combinations

The address range specified by C_BASEADDR and C_HIGHADDR must be a power of 2, and must be at least 0x200.

For example, if C_BASEADDR = 0xE0000000, C_HIGHADDR must be at least = 0xE00001FF.

OPB ADC I/O Signals

The I/O signals for the OPB ADC are listed in [Table 2](#). All signals are active high.

Table 2: OPB_ADC I/O Signals

Grouping	Signal Name	Interface	I/O	Initial State	Description	
ADC Signals	P1	DACout	ADC	O	0	Pulse string that drives the external low pass filter
	P2	Sample	ADC	O	0	Sample and Hold. This signal is true when ADC starts sampling the input and can drive an external Sample and Hold circuit
	P3	AgtR	ADC	I		Analog greater than Reference. This is the output of external comparator

Table 2: OPB_ADC I/O Signals (Continued)

Grouping	Signal Name	Interface	I/O	Initial State	Description	
OPB Signals	P4	OPB_Clk	OPB	I		OPB Clock
	P5	OPB_Rst	OPB	I		OPB Reset
	P6	OPB_ABus(0:C_OPB_AWIDTH-1)	OPB	I		OPB Address Bus
	P7	OPB_BE(0:C_OPB_DWIDTH/8-1)	OPB	I		OPB Byte Enables
	P8	OPB_DBus(0:C_OPB_DWIDTH-1)	OPB	I		OPB Data Bus
	P9	OPB_RNW	OPB	I		OPB Read, Not Write
	P10	OPB_Select	OPB	I		OPB Select
	P11	OPB_seqAddr	OPB	I		OPB Sequential Address
	P12	SIn_DBus(0:C_OPB_DWIDTH-1)	OPB	O	0	Slave read bus
	P13	SIn_errAck	OPB	O	0	Slave Error acknowledge
	P14	SIn_retry	OPB	O	0	Slave retry
	P15	SIn_toutSup	OPB	O	0	Slave Time-out Suppress
P16	SIn_xferAck	OPB	O	0	Slave transfer acknowledge	
System	P17	IP2INTC_Irpt	OPB	O	0	System interrupt:- Device interrupt output to microprocessor or system interrupt controller

OPB ADC Parameter - Port Dependencies

The dependencies between the OPB ADC design parameters and I/O signals are shown in Table 3.

Table 3: OPB ADC Parameter - Port Dependencies

	Name	Affects	Depends	Relationship Description
Design Parameters	G5	C_OPB_AWIDTH	P6	Width of the OPB Address Bus
	G6	C_OPB_DWIDTH	P7,P8,P12	Width of the OPB Data Bus and OPB Slave Data Bus
I/O Signals	P6	OPB_ABus(0:C_OPB_AWIDTH-1)	G5	Width of the OPB Address Bus varies according to C_OPB_AWIDTH
	P7	OPB_BE(0:C_OPB_DWIDTH/8-1)	G6	Width of the OPB Byte Enable varies according to C_OPB_DWIDTH
	P8	OPB_DBus(0:C_OPB_DWIDTH-1)	G6	Width of the OPB Data Bus varies according to C_OPB_DWIDTH
	P12	SIn_DBus(0:C_OPB_DWIDTH-1)	G6	Width of the OPB Slave Data Bus varies according to C_OPB_DWIDTH

OPB ADC Register Descriptions

Table 4 shows all the OPB ADC registers and their addresses.

Table 4: OPB ADC Registers

Register Name	OPB Address	Access
Device Global Interrupt Enable Register (GIE) ^(1,2)	C_BASEADDR + 0x01C	Read/Write
IP Interrupt Status Register (IPISR) ^(1,3)	C_BASEADDR + 0x020	Read/TOW ⁽⁷⁾
IP Interrupt Enable Register (IPIER) ^(1,3)	C_BASEADDR + 0x028	Read/Write
ADC Control Register(ADCCR) ⁽⁴⁾	C_BASEADDR + 0x100	Read/ Write
ADCout Data FIFO (FIFO) ⁽⁵⁾	C_BASEADDR + 0x104	Read
ADCout Data FIFO Occupancy Register (OCCY) ⁽⁶⁾	C_BASEADDR + 0x108	Read

Notes:

1. Please refer to the processor IP Reference Guide for a complete description of this register.
2. The bit mapping is shown in Figure 2.
3. The bit mapping is shown in Figure 3.
4. The bit mapping is shown in Figure 4.
5. The bit mapping is shown in Figure 5.
6. The bit mapping is shown in Figure 6.
7. TOW:Toggle-On Write.

Device Global Interrupt Enable Register (GIE)

The Device Global Interrupt Enable Register provides the final enable/disable for the interrupt output to the processor and resides in the IPIF. This is a single bit read/write register as shown in Figure 2. Table 5 shows the GIE bit definitions.

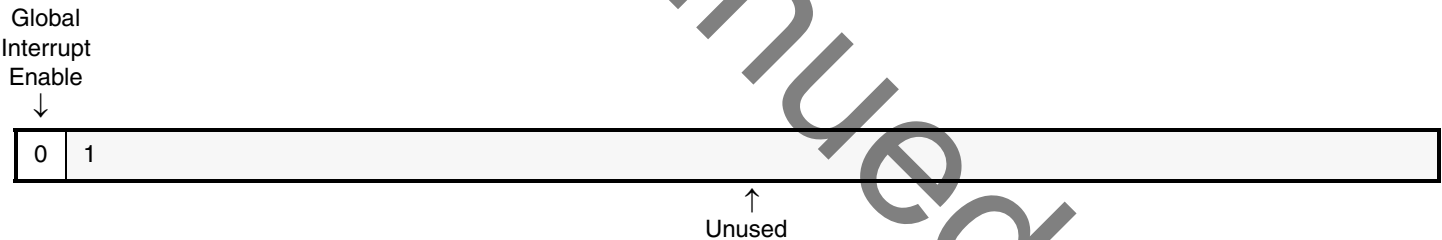


Figure 2: Device Global Interrupt Enable Register

Table 5: Device Global Interrupt Enable Register (GIE) Bit Definitions

Bit position(s)	Name	Core Access	Reset Value	Description
0	Global Interrupt Enable	Read/Write	'0'	Master Enable for routing Device Interrupt to the System Interrupt Controller '1' = Enabled '0' = Disabled
1 to 31	Unused	N/A	0	Unused. Set to zeroes on read

IP Interrupt Status and Interrupt Enable Registers

The IP Interrupt Status (IPISR) and Interrupt Enable Register (IPIER) are located inside the IPIF and provide a bit per interrupt. The interrupt enables have a one-to-one correspondence with the interrupt bits in the status register. The ADCout Data FIFO generates a single interrupt(FIFO Non-empty), when the ADCout Data FIFO contains the converted data for the

analog signal being sampled. The mapping of this interrupt to the IPIER and IPISR is shown in Figure 3. Table 6 and Table 7 shows the bit definitions for IPISR and IPIER.

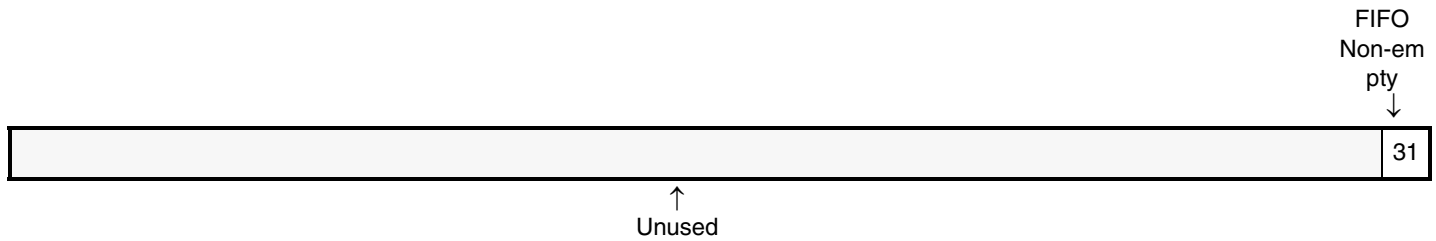


Figure 3: Interrupt Status and Interrupt Enable Registers

Table 6: Interrupt Status Register (IPISR) Bit Definitions

Bit position(s)	Name	Access	Reset Value	Description
0 - 30	Unused	N/A	0	Unused. Set to zeros.
31	FIFO Non-empty	Read/TOW	'0'	FIFO Non-empty Interrupt. <ul style="list-style-type: none"> '1' = ADCout Data FIFO contains the converted data '0' = ADCout Data FIFO is empty

Table 7: Interrupt Enable Register (IPIER) description

Bit(s)	Name	Core Access	Reset Value	Description
0 - 30	Unused	N/A	0	Unused. Set to zeros.
31	FIFO Non-empty	Read/Write	'0'	Enable/Disable the FIFO Non-empty Interrupt. <ul style="list-style-type: none"> '1' = Enabled '0' = Disabled (masked)

ADC Control Register (ADCCR)

The ADC Control Register contains the Enable Conversion bit (EC) and the Filter Settle Time Multiplier (FSTM). The Enable Conversion bit will enable/disable the Analog to Digital Conversion process. FSTM is a binary value, which depends on the RC characteristics of the low pass filter being used for conversion of DACout pulse train into equivalent analog signal. Bit sample time is effectively multiplied by Filter Settle Time Multiplier(FSTM) +1, so the user can configure the bit sample rate to match the Filter Settle Time characteristics. The width of FSTM value is configurable with the parameter C_FSTM_WIDTH. For most of the applications a 4-bit value is sufficient. As shown in Figure 4, the ADCCR contains the EC and FSTM. The bit definitions for ADC Control Register are shown in Table 8, when C_FSTM_WIDTH = 4.

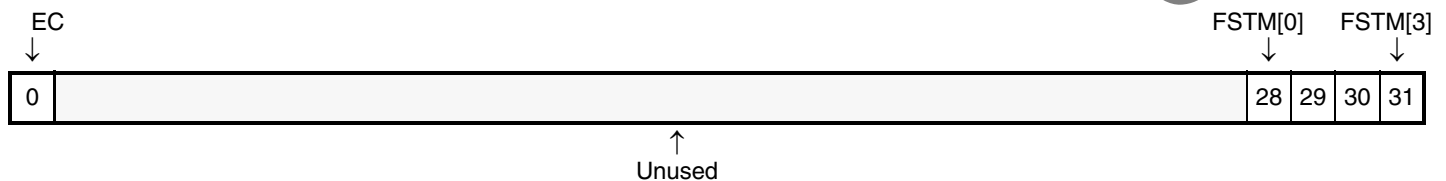


Figure 4: ADC Control Register (C_FSTM_WIDTH = 4)

Table 8: ADC Control Register Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
0	Enable Conversion bit(EC)	Read/Write	'0'	<ul style="list-style-type: none"> '1' = Enable Conversion '0' = Disable Conversion
1 - (31-C_FSTM_WIDTH)	Unused	N/A	0	Unused.Set to zeros
(32-C_FSTM_WIDTH) - 31	Filter Settle Time Multiplier(FSTM)	Read/Write	0	Filter Settle Time Multiplier(FSTM). These bits hold a binary value, which depends on the RC characteristics of Low pass filter

ADCout Data FIFO (FIFO)

This 16 entry deep SRL FIFO contains data to be output by OPB Delta-Sigma ADC. The FIFO bit definitions are shown in Table 9. Reading of this location will result in reading the current word out from the FIFO. Attempting to write to a full FIFO is not recommended and the results in that data being lost. Figure 5 shows the location for data on the OPB when C_DACIN_WIDTH is set to 9.

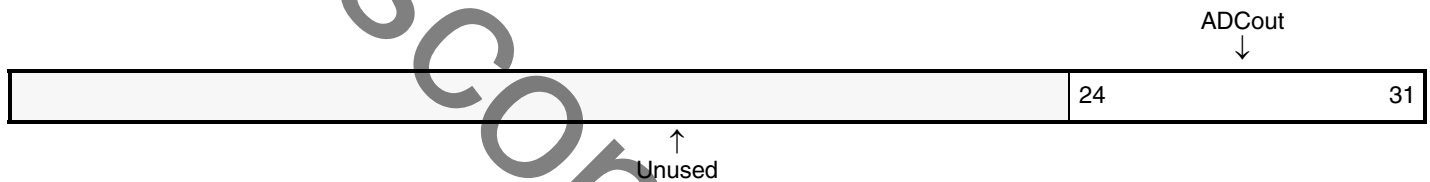


Figure 5: ADCout Data FIFO (C_DACIN_WIDTH = 9)

Table 9: ADCout Data FIFO Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
0 - [32-C_DACIN_WIDTH]	Unused	N/A	0	Unused. Set to zeros
[(32-C_DACIN_WIDTH)+1] - 31	ADCout (0 to C_DACIN_WIDTH-2)	Read/Write	0	ADCout. Digital value equivalent to the input analog sample

ADCout Data FIFO Occupancy Register (OCCY)

The ADCout Data FIFO Occupancy register contains the occupancy value of the ADCout Data FIFO. The value in this register is one less than the actual number of entries inside the ADCout Data FIFO. A zero value in this register when the Non-empty interrupt asserts indicates that there is one entry in the ADCout Data FIFO. A zero value in this register when the Non-empty interrupt negated indicates there are no entries in the ADCout Data FIFO. Figure 6 shows the location of Data Occupancy value in the 32-bit wide ADCout Data FIFO Occupancy register. OCCY bit definitions are shown in Table 10.

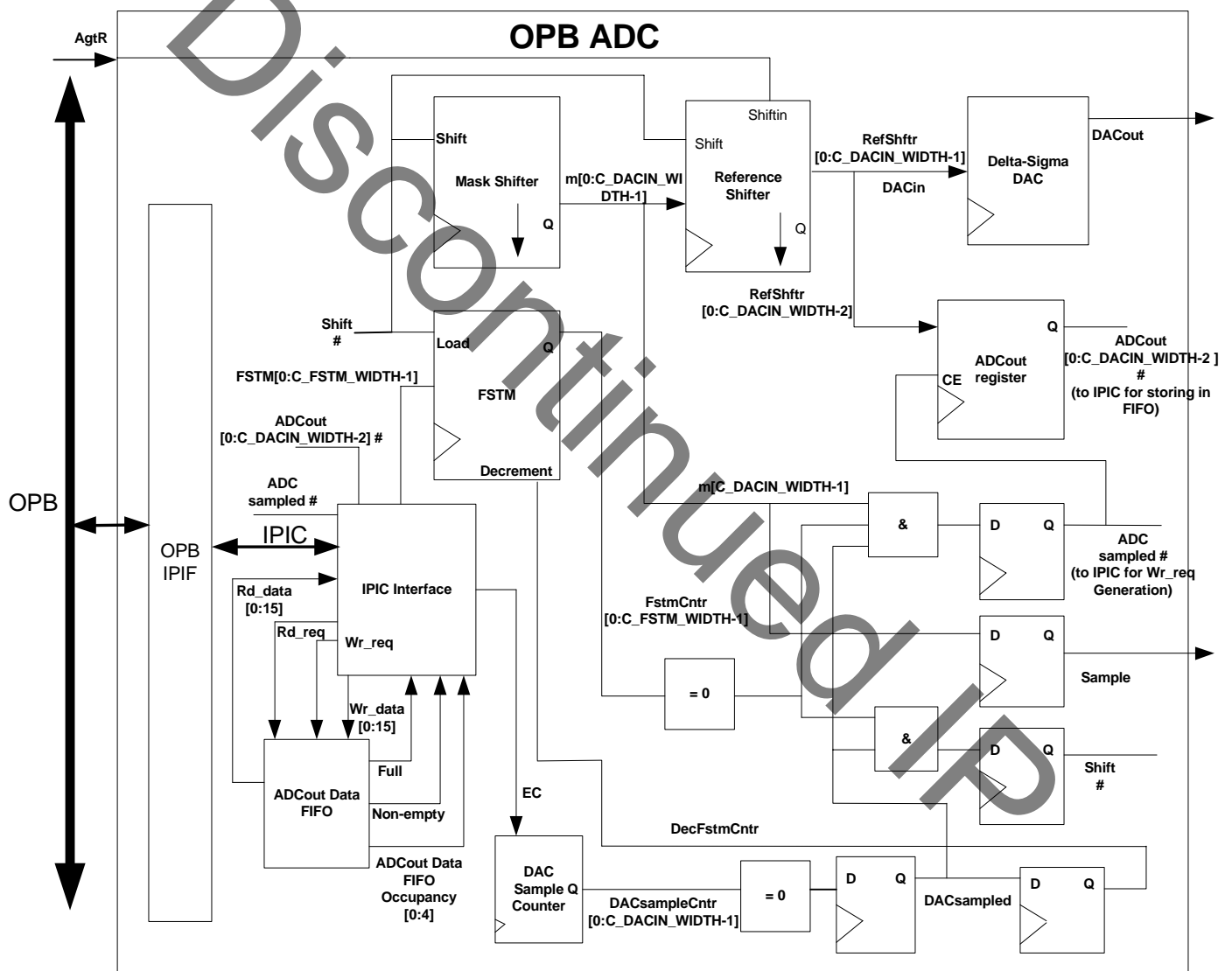


Figure 6: ADCout Data FIFO Occupancy Register (C_DACIN_WIDTH = 9)

Table 10: ADCout data FIFO Occupancy Register Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
0 - 26	Unused	N/A	0	Unused. Set to zeros
27 - 31	Occupancy Value	Read	0	Number of data words currently in ADCout Data FIFO. A value of "01000" implies that 9 locations in the FIFO are full

OPB ADC Block Diagram



Note: # indicates the signal is internally connected within the OPB ADC

Figure 7: Block Diagram of OPB ADC

Figure 7 shows the detailed block diagram of OPB ADC. The OPB ADC consists of the modules listed below.

- Delta-Sigma DAC
- DAC sample counter
- ADC Control Register
- Mask shifter
- Reference shifter
- ADCout Register
- ADCout Data FIFO
- OPB IPIF

The OPB ADC is implemented in a single VHDL module that instantiates the Delta-Sigma DAC. The ADCout Register width is configurable with defined parameter C_DACIN_WIDTH (C_DACIN_WIDTH specifies the width of DACin. The width of the ADCout Register is one less). Parameter C_FSTM_WIDTH is used to configure the width of FSTM.

Sample rate

The OPB ADC sample rate may be expressed as follows:

$$\text{OPB ADC}_{\text{SR}} = f_{\text{Clk}} / (2^{(\text{C_DACIN_WIDTH}+1)} \times (\text{FSTM} + 1) \times (\text{C_DACIN_WIDTH}+1)) \text{ samples/second.}$$

Conventional Analog to Digital Converters require at least twice the highest input frequency as sample rate. Delta-Sigma converters require higher f_{Clk} , so that sufficient number of bit-stream pulses can be produced. Obviously the more bit stream pulses can be produced, the better the approximation of the input signal by the average bit-stream. The average (low pass filtered) bit-stream never exactly represents the input signal. It is always superimposed with noise. One way to reduce this noise is to further increase the f_{Clk} (f_{Clk} is same as OPB Clock).

Table 11: OPB ADC sample rate calculation

OPB Clock frequency	FSTM loaded value	AnalogIn signal frequency range	ADC sample rate
40 MHz	4	<8457 Hz	17094 samples/second
80 MHz	4	<17094 Hz	34188 samples/second
100 MHz	4	<21367 Hz	42735 samples/second
40 MHz	8	<482 Hz	964 samples/second
80 MHz	8	<965 Hz	1930 samples/second
100 MHz	8	<1205 Hz	2411 samples/second

Table 11 shows the AnalogIn signal frequency range and ADC sample rate for various OPB Clock frequencies and FSTM values. Note that the sample rate is dependent on the OPB Clock frequency and the FSTM value, therefore these should be set appropriately based on the frequency of the AnalogIn signal to be sampled.

Delta-Sigma DAC

A Delta-Sigma DAC uses digital techniques. Delta-Sigma DACs are actually high-speed single bit DACs. Using digital feedback, a string of pulses are generated. The average duty cycle of the pulse string is proportional to the value of the binary input. The analog signal is created by passing the pulse string through an analog low-pass filter.

As a standard practice, the Delta-Sigma DAC input (DACin) in this implementation is an unsigned number with zero representing the lowest voltage level. The analog voltage output is also positive only. A zero on DACin produces zero volts at the output. All ones on DACin sets the output to nearly reach V_{CCO} . For AC signals, the positive bias on the analog signal can be removed with capacitive coupling to the load. Though the low pass filter can be driven with any of the Virtex or Spartan Select I/O output standards that both sink and source current, this design emphasizes the LVTTTL standard.

The Delta-Sigma DAC is one bit wider than the ADCout Register. This is required in order for the lowest numbered bit of the ADCout Register to be significant. When all of the bits have been sampled, the upper bits of the register feeding the Delta-Sigma DAC is transferred to the ADCout Register.

Figure 8 is a block diagram of a Delta-Sigma DAC. The width of DACin in the implementation described below can be configurable by changing the parameter C_DACIN_WIDTH. For simplicity, the block diagram depicts a Delta-Sigma DAC with an 8-bit DACin. The term “Delta-Sigma” refers to the arithmetic difference and sum, respectively. In this implementation, binary adders are used to create both the difference and the sum. Although the inputs to the Delta adder are unsigned, the outputs of both adders are considered signed numbers. The Delta Adder calculates the difference between the Delta-Sigma DACin and the current Delta-Sigma DACout, represented as a binary number. Because the Delta-Sigma DACout is a single bit, it is “all or nothing”; i.e., either all zeroes or all ones. As shown in Figure 8, the difference will result when adding the input to a value created by concatenating two copies of the most significant bit of the Sigma Latch with all zeros. This also compensates for the fact that Delta-Sigma DACin is unsigned. The Sigma Adder sums its previous output, held in Sigma Latch, with the current output of the Delta Adder. In most cases, the Delta adder is optimized out when the high level design is synthesized. This is because all bits on either the A or B inputs are zero, so A and B are simply merged, rather than added. The interface to VHDL Delta-Sigma DAC module in Figure 8 includes one output and three input signals as defined in Table 12.

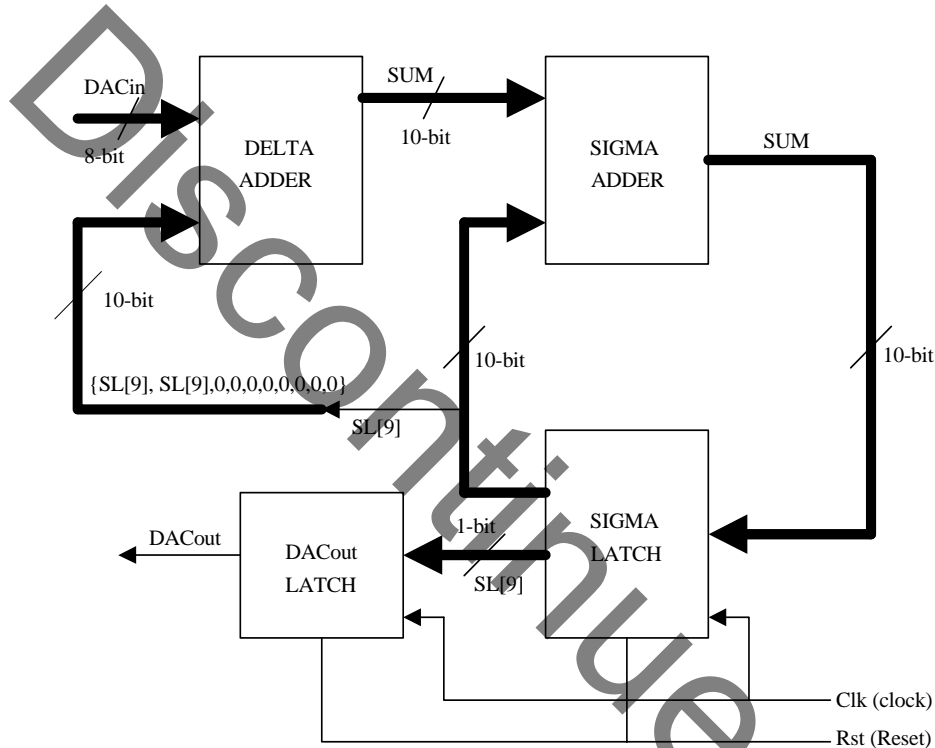


Figure 8: Delta-Sigma DAC Internal Block Diagram(C_DACIN_WIDTH=8)

Table 12: Delta -Sigma DAC Interface Signals

Signal	Direction	Description
DACout	Output	Pulse string that drives the external low pass filter (via an output driver such as OBUF_F_24)
DACin	Input	Digital input bus. Value must be setup to the positive edge of Clk. For high-speed operation, DACin should be sourced from a pipeline register that is clocked with Clk. For full resolution, each DACin value must be averaged over $2^{(C_DACIN_WIDTH+1)}$ clocks, so DACin should change only on intervals of $2^{(C_DACIN_WIDTH+1)}$ clock cycles
Clk	Input	Positive edge clock for the Sigma Latch and the DACout flip-flop
Rst	Input	Reset initializes the Sigma Latch and the DACout flip-flop. In this implementation, Sigma Latch is initialized to a value that corresponds to DACin of 0. If DACin starts at zero, there is no discontinuity

DAC sample counter: This is a binary up counter that is the same width as the Delta-Sigma DACin. When the Delta-Sigma DACin changes, it requires a minimum of one complete cycle of this counter to resolve the new value at the output.

ADC Control Register (ADCCR): The ADC Control Register contains the Enable Conversion bit (EC) and the Filter Settle Time Multiplier (FSTM). The Enable Conversion bit will enable/disable the Analog to Digital Conversion process. In high-precision applications, a large value for RC (external Low Pass Filter) is selected for the Delta-Sigma DAC low pass filter to minimize noise on the reference voltage. Bit sample time is effectively multiplied by FSTM+1, so the user can configure the bit sample rate to match the filter settle time characteristics. The width of FSTM is configurable with the parameter C_FSTM_WIDTH.

Mask shifter: This register, which is the same width as the Delta-Sigma DACin, endlessly rotates a single bit right. This is effectively the state machine that controls the bit sample sequence, implemented so that correct values can easily be loaded into the Reference shifter.

Reference shifter: This register drives the Delta-Sigma DACin. It always starts a sample with only the upper bit set. When only the upper bit is set, the comparator output will be true if the analog input is greater than $1/2 V_{CC0}$, and it will be false if the analog input is less than $1/2 V_{CC0}$. If the comparator output is true, the upper bit remains set, and the next lower bit is set. If the comparator output is false, the upper bit clears, and the next lower bit is set. This process continues all the way to the LSB, which causes voltage ADCref to "home in" on AnalogIn.

ADCout Register: This register latches the high order bits of the Reference shifter when the sample is left justified; that is, the comparator output that was sensed when only the upper bit of the Delta-Sigma DACin was set is in the MSB. The ADCout Register is one bit shorter than the Reference shifter, making the LSB in ADCout Register accurate to 1/2 LSB.

ADCout Data FIFO (FIFO): The ADCout Data FIFO is a 16-bit wide, 16 entry deep SRL FIFO for storing the converted analog values, i.e., a FIFO to store the ADCout values. FIFO Non-empty signal interrupts the processor. The FIFO Non-empty interrupt will be set and remains set as long as ADCout Data FIFO is non-empty. FIFO FULL flag indicates no additional writes can be performed in to the the FIFO. The ADCout Data FIFO Occupancy Register gives the number of ADCout values currently in FIFO.

OPB IPIF: OPB IPIF is a bi-directional interface between a user IP core and the OPB 32-bit bus standard. To simplify the process of attaching a OPB ADC to the OPB, the core make use of a portable, pre-designed bus interface called OPB IPIF, that takes care of the bus interface signals, bus protocols, and other interface issues.

The IPIC is a simple set of signals that connect the OPB ADC to the OPB IPIF logic. The majority of the IPIC signal set is common to all Xilinx IPIFs, so OPB ADC core designed with an IPIC can be easily ported to a different bus simply by using the appropriate IPIF.

Timing Diagram of OPB ADC

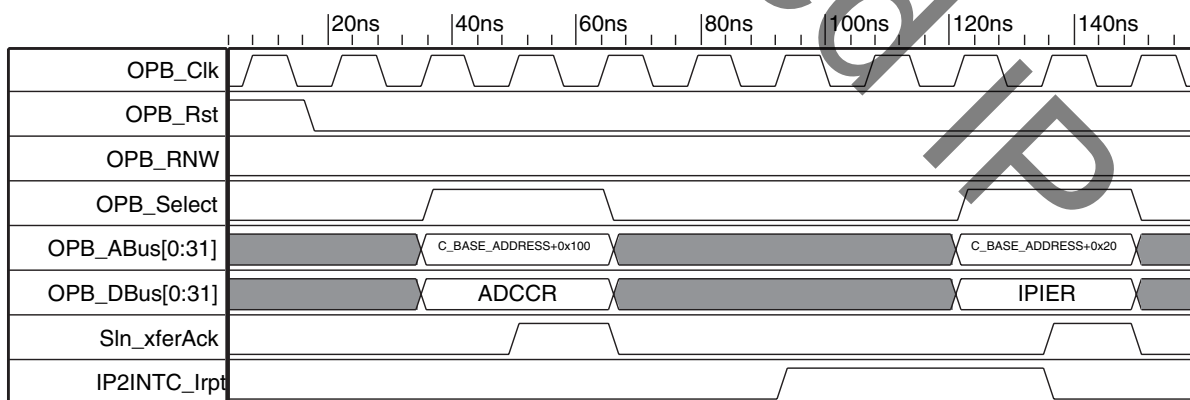


Figure 9: Timing Diagram of OPB ADC (Write Cycle)

Figure 9 shows Timing Diagram for Write cycle of OPB ADC. The complete process commences after OPB_Rst is deasserted.

- **Write Cycle:** Writing the ADCCR to load FSTM and EC (needs to be written at least once)
 - On the rising edge of the OPB_Clk Initiator (processor) asserts OPB_Select.
 - On the same clock edge Initiator places address of the ADCCR on to the OPB_ABus and deasserts OPB_RNW (low).
 - On the next rising edge of the OPB_Clk, Initiator places the ADCCR value on the OPB_DBus. The ADCCR value must set the EC and the FSTM.
 - Slave (OPB ADC) will acknowledge the acceptance of data by asserting SIn_xferAck after one clock cycle.
 - On the next rising edge of OPB_Clk the initiator deasserts the OPB_Select signal and the slave deasserts the SIn_xferAck and this marks the end of the write cycle

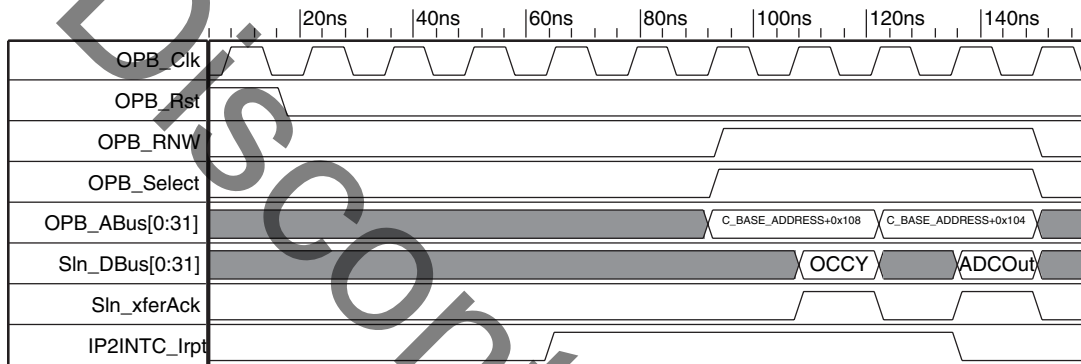


Figure 10: OPB ADC (Read Cycle) Timing Diagram

Figure 10 shows the Timing Diagram for Read cycle of OPB ADC. This process is initiated by the processor after IP2INTC_Irpt is asserted.

- **Read Cycle:** Reading of the ADCout DataFIFO (FIFO). Assertion of IP2INTC_Irpt indicates availability of converted digital values in ADCout Data FIFO i.e. it is asserted when FIFO is non empty.
 - On the rising edge of the OPB_Clk Initiator (processor) asserts OPB_Select.
 - On the same clock edge Initiator places address of the ADCout Data FIFO Occupancy register(OCCY) on to the OPB_ABus and asserts OPB_RNW (high).
 - On the next rising edge of the OPB_Clk, Slave (OPB ADC) places the OCCY value on the SIn_DBus.
 - Slave (OPB ADC) will acknowledge the completion of request by asserting SIn_xferAck after one clock cycle.
 - On the next rising clock edge Initiator places address of the FIFO on to the OPB_ABus and asserts OPB_RNW (high).
 - If the FIFO is not empty then on rising edge of the OPB_Clk, Slave (OPB ADC) places the data read from the FIFO(ADCout) on to the SIn_DBus.
 - Slave (OPB ADC) will acknowledge the completion of request by asserting SIn_xferAck after one clock cycle.
 - On the next rising edge of OPB_Clk the initiator deasserts the OPB_Select and OPB_RNW signal and the slave deasserts the SIn_xferAck. This marks the end of the read cycle.

Design Implementation

Target Technology

The intended target technology is Virtex and Spartan FPGA families.

Device Utilization and Performance Benchmarks

Because the OPB ADC is a module that will be used with other design units in the FPGA, the utilization and timing numbers reported in [Table 13](#) are estimates. The actual results might vary.

Table 13: OPB ADC FPGA Performance and Resource Utilization Benchmarks (Virtex-IIPro-xc2vp7,-6)

Parameter Values		Device Resources			f _{MAX} (MHz)
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	4-input LUTs	f _{MAX}
11	8	209	253	229	115
11	6	205	251	223	109
11	4	202	249	215	112
9	8	198	243	213	112
9	6	194	241	207	112
9	4	191	239	199	114

Reference Documents

Analog Devices Data Converter Reference Manual, Volume I, 1992

High Performance Stereo Bit-Stream DAC with Digital Filter, R. Finck, IEEE Transactions on Consumer Electronics, Vol. 35, No. 4, Nov. 1989.

Xilinx Application note (Xapp 155.pdf)

Revision History

Date	Version	Revision
11/03/04	1.1	Initial Xilinx release.
4/4/05	1.2	Updated for EDK 7.1.1 SP1 release; updated trademarks and supported device listing.
12/1/05	1.3	Added Spartan-3E to supported device listing.