XPS Delta-Sigma Analog to Digital Converter (ADC) (v1.01a)

DS587 December 2, 2009 **Product Specification**

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 XPS Delta-Sigma ADC (XPS ADC) using Delta-Sigma conversion technique. This soft IP core is designed to interface with the PLB (Processor Local Bus).

Features

- Connects as a 32-bit slave on PLB V4.6 buses of 32, 64 or 128 bits
- Supports single beat transactions
- Selectable ADC resolution
- 16 entry deep data FIFO

LogiCORE™ IP Facts	
Core Specifics	
Supported Device Family	Spartan®-3A/3A DSP, Spartan-3, Spartan-3E, Automotive Spartan 3/3E/3A/3A DSP, Spartan-6, Virtex®-4 /4Q/4QV, Virtex-5/5FX, Virtex-6/6CX
Resources Used	
See Table 13, Table 14, Table 15, Table 16, and Table 17.	
Provided with Core	
Documentation	Product Specification
Design File Formats	VHDL
Constraints File	EDK TCL Generated
Verification	N/A
Instantiation Template	EDK
Design Tool Requirements	
Xilinx Implementation Tools	ISE® 11.4 or later
Verification	ModelSim PE/SE 6.4b or later
Simulation	ModelSim PE/SE 6.4b or later
Synthesis	XST
Support	
Provided by Xilinx, Inc.	

Functional Description

The modules comprising the XPS ADC are shown in Figure 1 and described in the subsequent sections. The XPS ADC modules are:

- Interrupt Controller
- PLB Interface Module
- IPIC Interface
- ADCout Data FIFO
- Delta-Sigma DAC
- ADC

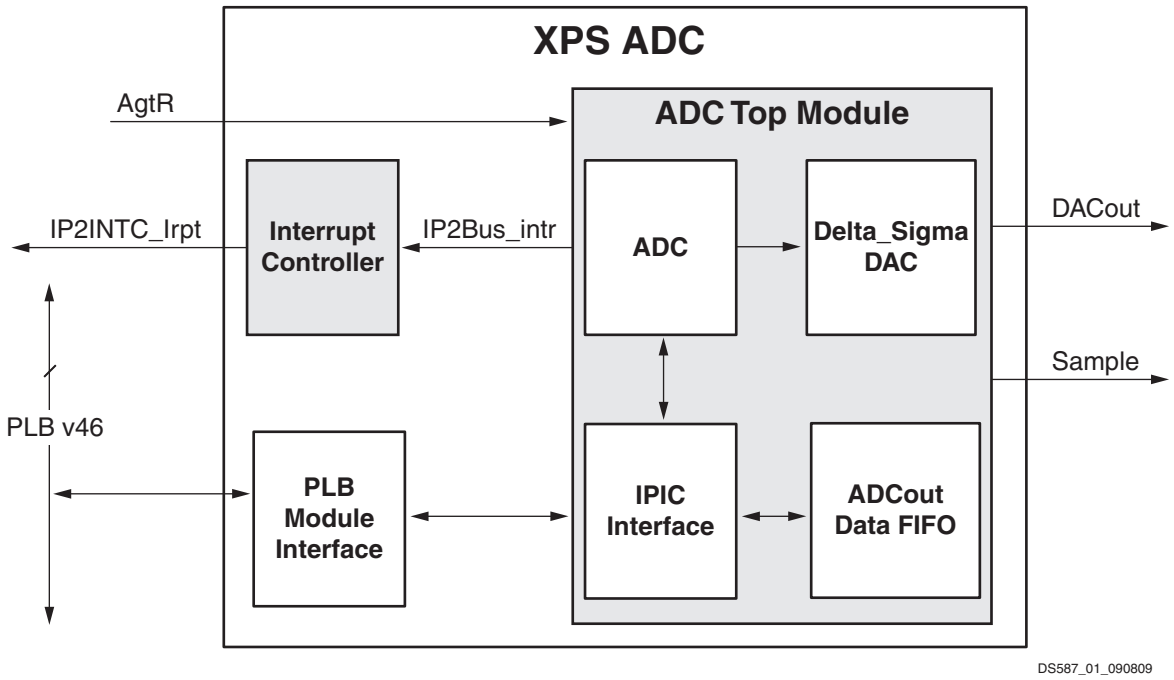

Figure 1: **XPS ADC Block Diagram**

Interrupt Controller: The Interrupt Controller provides interrupt capture support for XPS ADC. The Interrupt Controller is used to collect interrupts from XPS ADC by which XPS ADC requests the attention of the microprocessor.

PLB Interface Module: The PLB Interface Module is a bi-directional interface between a user IP core and the PLB bus standard. To simplify the process of attaching a XPS ADC to the PLB, the core makes use of a portable, pre-designed bus interface called PLB Interface Module, that takes care of the bus interface signals, bus protocols and other interface issues.

IPIC Interface: The IPIC is a simple set of signals that connects the XPS ADC to the PLB Interface Module. This module generates the required read and write request signals by using the output signals of the FIFO.

ADCout Data FIFO: The ADCout Data FIFO is a 16-bit wide, 16 entry deep FIFO for storing the converted analog values, i.e., a FIFO to store the ADCout values. The 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.

Delta-Sigma DAC: The Delta-Sigma DAC is a high-speed single bit DAC that uses digital techniques. Using digital feedback, a string of pulses is 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.

Following standard practice, the Delta-Sigma DAC input (DACin) 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 causes 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® FPGA Select I/O output standards that both sink and source current, this design emphasizes the LVTTL 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 2 is a block diagram of a Delta-Sigma DAC. The width of DACin can be configured by changing the parameter C_DACIN_WIDTH. For simplicity, the block diagram depicts a Delta-Sigma DAC with a 9-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. 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 2, 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 the Sigma Latch, with the current output of the Delta Adder. Since the Delta Adder sums a value with the upper two bits as zeroes ({0,0,DACin}) with a value having all but the upper two bits as zeroes ({SL[10], SL[10], 0,0,0,0,0,0,0,0}), it has a trivial implementation of simply passing through the non-zero bits. No actual adder is needed.

The interface to VHDL Delta-Sigma DAC module in Figure 2 includes one output and three input signals as defined in Table 1.

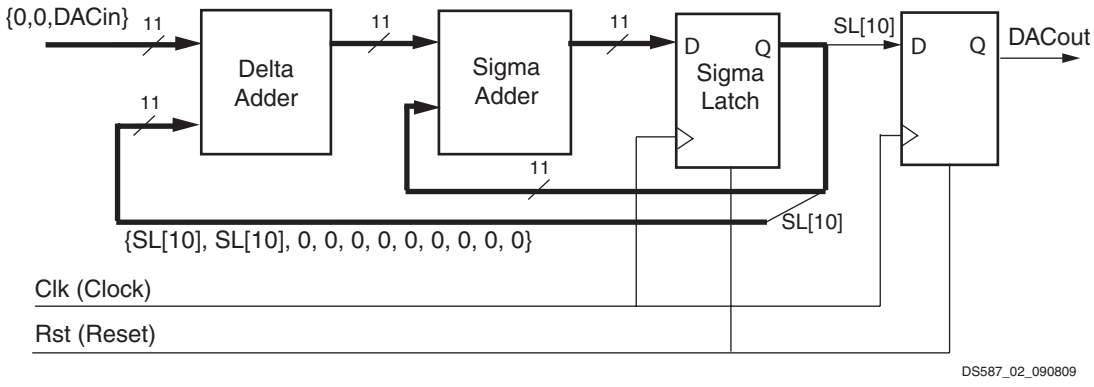

Figure 2: **Delta-Sigma DAC Internal Block Diagram (C_DACIN_WIDTH = 9)**

Table 1: **Delta-Sigma DAC Interface Signals**

Signal	Direction	Description
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.
DACin	Input	Digital input bus. Value must be stable at 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 $2^{(C\_DACIN\_WIDTH)}$ clocks, so DACin should change only on intervals of $2^{(C\_DACIN\_WIDTH)}$ clock cycles.
DACout	Output	Pulse string that drives the external low pass filter (via an output driver such as OBUF_F_24).

The Analog to Digital Converter uses an external analog comparator which compares the input voltage to a voltage generated by the DAC. Figure 3 shows how a typical implementation of analog to digital conversion is performed using the XPS ADC. A Delta-Sigma DAC, which is a primary block of the XPS ADC core, is used to generate a reference voltage ADCref 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_{CCO}, where V_{CCO} 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 0 V to V_{CCO}, 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.

Because of the serial nature of both the Delta-Sigma DAC and the analog sampling process, this XPS ADC is useful only on signals that change slowly. 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 LVTTL output buffer is normally used to drive the RC filter. Most comparators have uncommitted collector/drain outputs, so R_P is usually needed.

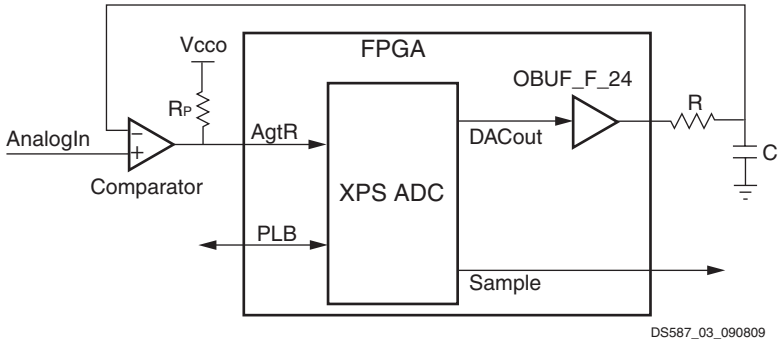

Figure 3: **Implementation of Analog to Digital Converter Using XPS ADC**

Sample rate

The XPS ADC sample rate may be expressed as follows:

$XPS\ ADC_{SR} = f_{Clk} / (2^{(C\_DACIN\_WIDTH)} \times (FSTM + 1) \times (C\_DACIN\_WIDTH))$ 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 PLB Clock).

Table 2 shows the AnalogIn signal frequency range and ADC sample rate for various PLB Clock frequencies and FSTM values. Note that the sample rate is dependent on the PLB Clock frequency and the FSTM value, therefore these should be set appropriately based on the frequency of the AnalogIn signal to be sampled.

Table 2: **XPS ADC Sample Rate Calculation (C_DACIN_WIDTH = 9)**

PLB Clock frequency	FSTM loaded value	AnalogIn signal frequency range	ADC sample rate
40 MHz	4	<868 Hz	1736 samples/second
80 MHz	4	<1736 Hz	3472 samples/second
100 MHz	4	<2170 Hz	4340 samples/second
40 MHz	8	<482 Hz	964 samples/second
80 MHz	8	<965 Hz	1929 samples/second
100 MHz	8	<1205 Hz	2411 samples/second

XPS ADC I/O Signals

The XPS ADC I/O signals are listed and described in Table 3. All signals are active high.

Table 3: **XPS ADC I/O Signal Description**

Port	Signal Name	Interface	I/O	Initial State	Description
System Signals					
P1	SPLB_Clk	System	I	-	PLB clock
P2	SPLB_Rst	System	I	-	PLB reset
P3	IP2INTC_Irpt	System	O	0	Interrupt signal from XPS ADC
PLB Slave Interface Input Signals					
P4	PLB_ABus[0: C_SPLB_AWIDTH - 1]	PLB	I	-	PLB address bus
P5	PLB_PAValid	PLB	I	-	PLB primary address valid
P6	PLB_masterID[0: C_SPLB_MID_WIDTH - 1]	PLB	I	-	PLB current master identifier
P7	PLB_RNW	PLB	I	-	PLB read not write

Table 3: **XPS ADC I/O Signal Description *(Cont'd)***

Port	Signal Name	Interface	I/O	Initial State	Description
P8	PLB_BE[0: (C_SPLB_DWIDTH/8) - 1]	PLB	I	-	PLB byte enables
P9	PLB_size[0:3]	PLB	I	-	PLB size of requested transfer
P10	PLB_type[0:2]	PLB	I	-	PLB transfer type
P11	PLB_wrDBus[0: C_SPLB_DWIDTH - 1]	PLB	I	-	PLB write data bus
Unused PLB Slave Interface Input Signals					
P12	PLB_UABus[0: 31]	PLB	I	-	PLB upper address bits
P13	PLB_SAValid	PLB	I	-	PLB secondary address valid
P14	PLB_rdPrim	PLB	I	-	PLB secondary to primary read request indicator
P15	PLB_wrPrim	PLB	I	-	PLB secondary to primary write request indicator
P16	PLB_abort	PLB	I	-	PLB abort bus request
P17	PLB_busLock	PLB	I	-	PLB bus lock
P18	PLB_MSize	PLB	I	-	PLB data bus width indicator
P19	PLB_lockErr	PLB	I	-	PLB lock error
P20	PLB_wrBurst	PLB	I	-	PLB burst write transfer
P21	PLB_rdBurst	PLB	I	-	PLB burst read transfer
P22	PLB_wrPendReq	PLB	I	-	PLB pending bus write request
P23	PLB_rdPendReq	PLB	I	-	PLB pending bus read request
P24	PLB_wrPendPri[0:1]	PLB	I	-	PLB pending write request priority
P25	PLB_rdPendPri[0:1]	PLB	I	-	PLB pending read request priority
P26	PLB_reqPri[0:1]	PLB	I	-	PLB current request priority
P27	PLB_TAttribute[0:15]	PLB	I	-	PLB transfer attribute
PLB Slave Interface Output Signals					
P28	Sl_addrAck	PLB	O	0	Slave address acknowledge
P29	Sl_SSize[0:1]	PLB	O	0	Slave data bus size
P30	Sl_wait	PLB	O	0	Slave wait
P31	Sl_rearbitrate	PLB	O	0	Slave bus rearbitrate
P32	Sl_wrDAck	PLB	O	0	Slave write data acknowledge
P33	Sl_wrComp	PLB	O	0	Slave write transfer complete
P34	Sl_rdDBus[0: C_SPLB_DWIDTH - 1]	PLB	O	0	Slave read data bus
P35	Sl_rdDAck	PLB	O	0	Slave read data acknowledge
P36	Sl_rdComp	PLB	O	0	Slave read transfer complete

Table 3: **XPS ADC I/O Signal Description *(Cont'd)***

Port	Signal Name	Interface	I/O	Initial State	Description
P37	SI_MBusy[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave busy
P38	SI_MWrErr[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave write error
P39	SI_MRdErr[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave read error
Unused PLB Slave Interface Output Signals					
P40	SI_wrBTerm	PLB	O	0	Slave terminate write burst transfer
P41	SI_rdWdAddr[0:3]	PLB	O	0	Slave read word address
P42	SI_rdBTerm	PLB	O	0	Slave terminate read burst transfer
P43	SI_MIRQ[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Master interrupt request
XPS ADC Signals					
P44	DACout	ADC	O	0	Pulse string that drives the external low pass filter.
P45	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.
P46	AgtR	ADC	I	-	Analog greater than Reference. This is the output of external comparator.

XPS ADC Design Parameters

To allow the user to create a XPS ADC that is uniquely tailored for the user's system, certain features are parameterizable in the XPS ADC design. This allows the user to have a design that utilizes only the resources required by the system and runs at the best possible performance. The features that are parameterizable in the XPS ADC core are as shown in Table 4.

Table 4: **XPS ADC Design Parameters**

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
System Parameter					
G1	Target FPGA family	C_FAMILY	spartan3, spartan3e, spartan3a, spartan3adsp, aspartan3, aspartan3e, aspartan3a, aspartan3adsp,spartan6, virtex4, qvirtex4, qvvirtex4, virtex5, virtex5fx, virtex6, virtex6cx	virtex5	string
PLB Parameters					
G2	XPS ADC Base Address	C_BASEADDR	Valid Address (1)	None (2)	std_logic_vector
G3	XPS ADC High Address	C_HIGHADDR	Valid Address (1)	None (2)	std_logic_vector
G4	PLB address width	C_SPLB_AWIDTH	32	32	integer
G5	PLB data width	C_SPLB_DWIDTH	32, 64, 128	32	integer
G6	Selects point-to-point or shared PLB topology	C_SPLB_P2P	0 = Shared Bus Topology	0	integer
G7	PLB Master ID Bus Width	C_SPLB_MID_WIDTH	$\log_2$(C_SPLB_NUM_MASTERS) with a minimum value of 1	1	integer
G8	Number of PLB Masters	C_SPLB_NUM_MASTERS	1 - 16	1	integer
G9	Width of the Slave Data Bus	C_SPLB_NATIVE_DWIDTH	32	32	integer
G10	Enable burst support	C_SPLB_SUPPORT_BURSTS	0	0	integers
XPS ADC Features					
G11	Delta-Sigma DAC input width. This parameter is set to one less than the desired resolution of the analog-to-digital conversion.	C_DACIN_WIDTH	9, 11	9	integer
G12	Filter Settle Time Multiplier(FSTM) width	C_FSTM_WIDTH	4 - 8	4	integer

1. C_BASEADDR must be a multiple of the range size, where the range size is C_HIGHADDR - C_BASEADDR + 1 and must be a power of two large enough to accommodate all of the registers.
2. 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.

Allowable Parameter Combinations

The address-range size 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.

XPS ADC Parameter - Port Dependencies

The dependencies between the XPS ADC core design parameters and I/O signals are described in Table 5. In addition, when certain features are parameterized out of the design, the related logic will no longer be a part of the design. The unused input signals and related output signals are set to a specified value.

Table 5: **XPS ADC Design Parameter - Port Dependencies**

Generic or Port	Name	Affects	Depends	Relationship Description
Design Parameters				
G4	C_SPLB_AWIDTH	P4	-	Affects number of bits in address bus.
G5	C_SPLB_DWIDTH	P8,P11, P34	-	Affects number of bits in data bus.
G7	C_SPLB_MID_WIDTH	P6	G8	Affects the width of current master identifier signals and depends on $\log_2$(C_SPLB_NUM_MASTERS) with a minimum value of 1.
G8	C_SPLB_NUM_ MASTERS	P37,P38, P39,P43	-	Affects the width of busy and error signals.
I/O Signals				
P4	PLB_ABus[0: C_SPLB_AWIDTH - 1]	-	G4	Width varies with the size of the PLB address bus.
P6	PLB_masterID[0: C_SPLB_MID_WIDTH - 1]	-	G7	Width varies with the size of the PLB master identifier bus.
P8	PLB_BE[0: (C_SPLB_DWIDTH/8) -1]	-	G5	Width varies with the size of the PLB data bus.
P11	PLB_wrDBus[0: C_SPLB_DWIDTH - 1]	-	G5	Width varies with the size of the PLB data bus.
P34	SI_rdDBus[0: C_SPLB_DWIDTH - 1]	-	G5	Width varies with the size of the PLB data bus.
P37	SI_MBusy[0: C_SPLB_NUM_ MASTERS - 1]	-	G8	Width varies with the size of the PLB number of masters.

Table 5: **XPS ADC Design Parameter - Port Dependencies *(Cont'd)***

Generic or Port	Name	Affects	Depends	Relationship Description
P38	SI_MWrErr[0: C_SPLB_NUM_ MASTERS - 1]	-	G8	Width varies with the size of the PLB number of masters.
P39	SI_MRdErr[0: C_SPLB_NUM_ MASTERS - 1]	-	G8	Width varies with the size of the PLB number of masters.
P43	SI_MIRQ[0: C_SPLB_NUM_MASTE RS - 1]	-	G8	Width varies with the size of the PLB number of masters.

XPS ADC Register Descriptions

Table 6 shows the XPS ADC Registers and their addresses.

Table 6: **XPS ADC Registers**

Base Address + Offset (hex)	Register Name	Default Value (hex)	Access	Register Description
C_BASEADDR + 0x01C	GIE	0x0	Read/Write	Device Global Interrupt Enable Register
C_BASEADDR + 0x020	IPISR	0x0	Read/TOW (1)	IP Interrupt Status Register
C_BASEADDR + 0x028	IPIER	0x0	Read/Write	IP Interrupt Enable Register
C_BASEADDR + 0x100	ADCCR	0x0	Read/ Write	ADC Control Register
C_BASEADDR + 0x104	FIFO	0x0	Read (2)	ADCout Data FIFO
C_BASEADDR + 0x108	OCCY	0x0	Read (2)	ADCout Data FIFO Occupancy Register

1. TOW = Toggle On Write. Writing a '1' to a bit position within the register causes the corresponding bit position in the register to toggle.
2. Writing of a read only register has no effect.

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 PLB Interface Module. This is a single bit read/write register as shown in Figure 4. Table 7 shows the GIE bit definitions.

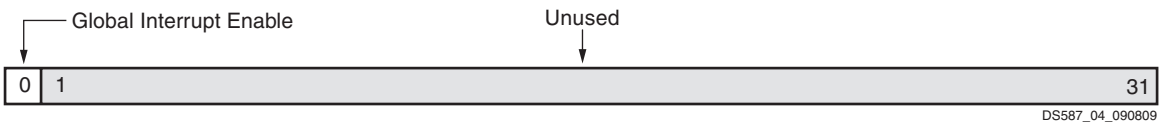

Figure 4: **Device Global Interrupt Enable Register**

Table 7: **Device Global Interrupt Enable Register (GIE) Bit Definitions**

Bit(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

IP Interrupt Status and Interrupt Enable Registers

The XPS ADC supports a single interrupt condition, FIFO Non-empty, which indicates that conversion samples are available. The interrupt status and interrupt enable bits are located at bit-position 31 in the IP Interrupt Status Register (IPISR) and IP Interrupt Enable Register (IPIER), respectively. See Figure 5, Table 8 and Table 9.

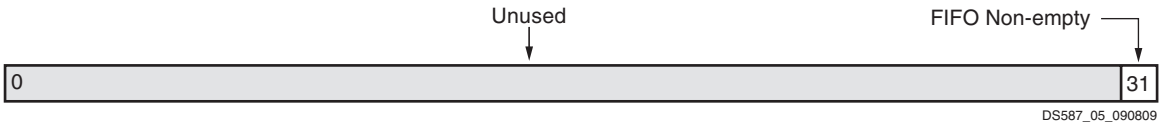

Figure 5: **Interrupt Status and Interrupt Enable Register**

Table 8: **Interrupt Status Register (IPISR) Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 to 30	Unused	N/A	0	Unused
31	FIFO Non-empty	Read/TOW	'0'	FIFO Non-empty Interrupt. '1' = ADCout Data FIFO contains the converted data '0' = ADCout Data FIFO is empty

Table 9: **Interrupt Enable Register (IPIER) Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 to 30	Unused	N/A	0	Unused
31	FIFO Non-empty	Read/Write	'0'	Enable/Disable the FIFO Non-empty Interrupt. '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 EC 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 6, the ADCCR contains the EC and FSTM. The bit definitions for ADC Control Register are shown in Table 10.

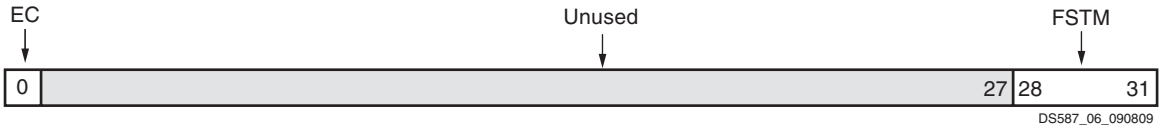

Figure 6: **ADC Control Register (C_FSTM_WIDTH = 4)**

Table 10: **ADCout Control Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0	Enable Conversion bit (EC)	Read/Write	'0'	'1' = Enabled '0' = Disabled (masked)
1 to [(32-C_ FSTM_WIDTH)-1]	Unused	N/A	0	Unused
[32-C_FSTM_ WIDTH] to 31	Filter Settle Time Multiplier (FSTM)	Read/Write	0	These bits hold a binary value which depends on the RC characteristics of Low pass filter, as described above.

ADCout Data FIFO (FIFO)

This 16 entry deep FIFO contains data to be output by XPS ADC. The ADCout Data FIFO bit definitions are shown in Table 11. Reading of this location will result in reading a conversion sample from the FIFO. Software must check for the presence of data before reading. When a read request is issued to an empty FIFO a bus error will be generated and the result is undefined. Timely reading by software is required to maintain vacancy in the FIFO for incoming conversions samples. Incoming samples that encounter a full FIFO are lost. Figure 7 shows the location for data when C_DACIN_WIDTH is set to 9.

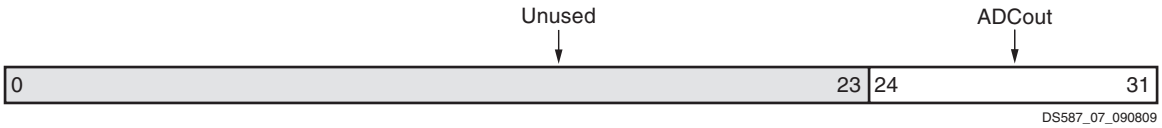

Figure 7: **ADCout Data FIFO (C_DACIN_WIDTH = 9)**

Table 11: **ADCout Data FIFO Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 to [32-C_DACIN _WIDTH]	Unused	N/A	0	Unused
[(32-C_DACIN _WIDTH)+1] to 31	ADCout	Read	0	Digital value equivalent of the input analog sample. The number of bits of resolution of ADCout is C_DACIN_WIDTH - 1.

ADCout Data FIFO Occupancy Register (OCCY)

The ADCout Data FIFO Occupancy Register contains the occupancy value of the ADCout Data FIFO. Reading this register can be used to determine if the FIFO is empty, also the FIFO Non-Empty Interrupt conveys that information.The value read is the binary count value, therefore reading all zeros implies that no location is filled and reading "10000" implies that all sixteen locations are filled. Figure 8 shows the location of Data Occupancy value in the 32-bit wide ADCout Data FIFO Occupancy Register. OCCY bit definitions are shown in Table 12.

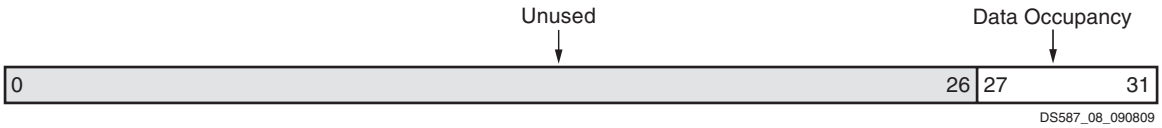

Figure 8: **ADCout Data FIFO Occupancy Register (C_DACIN_WIDTH = 9)**

Table 12: **ADCout Data FIFO Occupancy Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 to 26	Unused	N/A	0	Unused
27 to 31	Data Occupancy	Read	0	Number of data words currently in ADCout Data FIFO.

XPS ADC Timing Diagrams

Figure 9 shows the Timing Diagram for the Read cycle of XPS ADC.

Figure 10 shows the Timing Diagram for the Write cycle of XPS ADC.

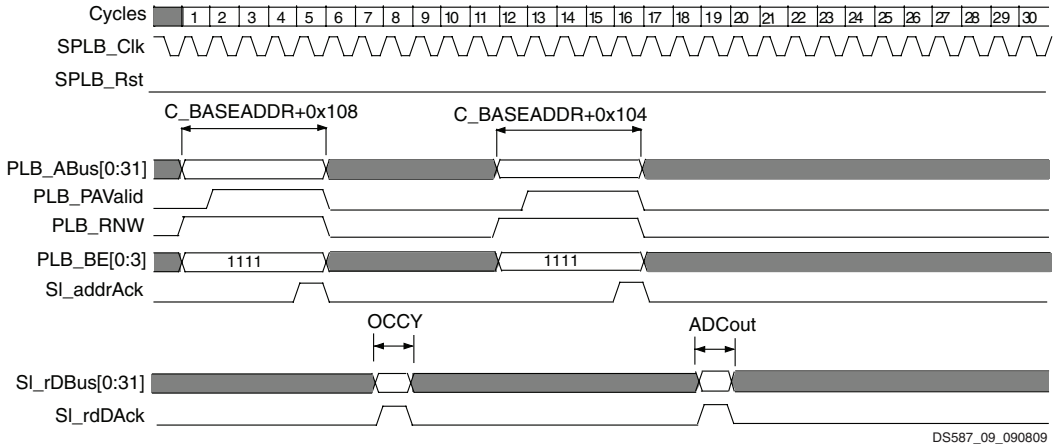

Figure 9: **XPS ADC Read Cycle Timing Diagram**

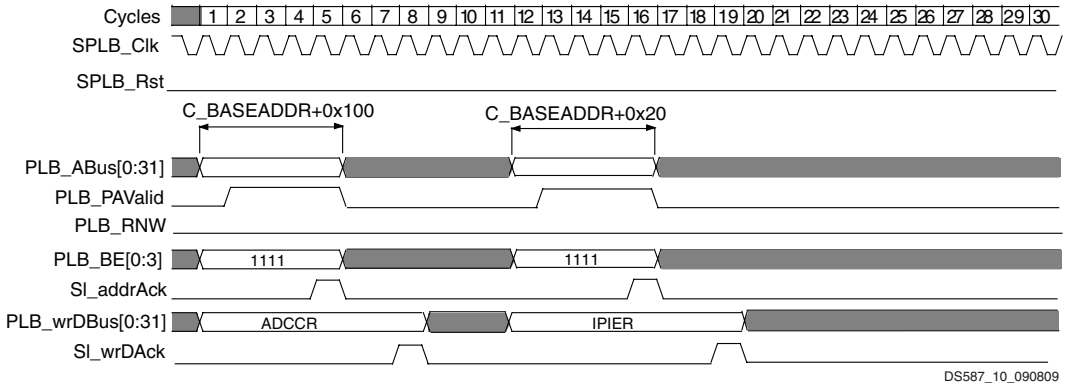

Figure 10: **XPS ADC Write Cycle Timing Diagram**

Design Implementation

Target Technology

The target technology is an FPGA listed in the Supported Device Family field of the LogiCORE IP Facts table.

Device Utilization and Performance Benchmarks

Core Performance

Since the XPS ADC core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When the XPS ADC core is combined with other

designs in the system, the utilization of FPGA resources and timing of the XPS ADC design will vary from the results reported here.

The XPS ADC resource utilization for various parameter combinations measured with Virtex-4 as the target device are detailed in Table 13.

Table 13: **Performance and Resource Utilization Benchmarks on Virtex-4 (xc4vlx80-ff1148-11)**

Parameter Values		Device Resources			Performance
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	LUTs	F_{MAX} (MHz)
11	8	143	142	212	173.76
11	6	157	159	232	165.12
11	4	165	155	253	167.44
9	8	153	150	226	172.44
9	6	152	148	225	172.50
9	4	138	138	202	168.15

The XPS ADC resource utilization for various parameter combinations measured with Virtex-5 as the target device are detailed in Table 14.

Table 14: **Performance and Resource Utilization Benchmarks on Virtex-5 (xc5vlx85-ff676-2)**

Parameter Values		Device Resources			Performance
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	LUTs	F_{MAX} (MHz)
11	8	202	294	215	203.99
11	6	181	239	193	192.71
11	4	180	278	213	179.46
9	8	180	278	213	179.46
9	6	181	248	198	192.71
9	4	181	239	193	203.99

The XPS ADC resource utilization for various parameter combinations measured with Virtex-6 as the target device are detailed in Table 15.

Table 15: **Performance and Resource Utilization Benchmarks on Virtex-6 (xc6vlx115t-1-ff1156)**

Parameter Values		Device Resources			Performance
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	LUTs	F_{MAX} (MHz)
11	8	99	176	232	195.69
11	6	110	172	241	168.29
11	4	103	167	225	169.49
9	8	99	165	217	175.07
9	6	94	142	196	198.05
9	4	87	150	191	176.33

The XPS ADC resource utilization for various parameter combinations measured with Spartan-3 as the target device are detailed in Table 16.

Table 16: **Performance and Resource Utilization Benchmarks on Spartan-3e (xc3s1200e-4-fg400)**

Parameter Values		Device Resources			Performance
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	LUTs	F_{MAX} (MHz)
11	8	254	289	244	159.28
11	6	268	301	239	158.17
11	4	268	301	239	158.17
9	8	234	250	218	160.02
9	6	244	261	224	162.25
9	4	234	250	218	160.02

The XPS ADC resource utilization for various parameter combinations measured with Spartan-6 as the target device are detailed in Table 17.

Table 17: **Performance and Resource Utilization Benchmarks on Spartan-6(xc6slx45-2-fgg676)**

Parameter Values		Device Resources			Performance
C_DACIN_WIDTH	C_FSTM_WIDTH	Slices	Slice Flip-Flops	LUTs	F_{MAX} (MHz)
11	8	96	158	204	101.35
11	6	112	154	211	100.64
11	4	86	149	178	107.47
9	8	111	147	190	101.64
9	6	88	142	174	100.47
9	4	87	138	165	105.08

System Performance

To measure the system performance (Fmax) of this core, this core was added to a Virtex-4 FPGA system, a Virtex-5 FPGA system, and a Spartan-3A FPGA system as the Device Under Test (DUT) as shown in Figure 11, Figure 12, and Figure 13.

Because the XPS Delta-Sigma ADC core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When this core is combined with other designs in the system, the utilization of FPGA resources and timing of the core design will vary from the results reported here.

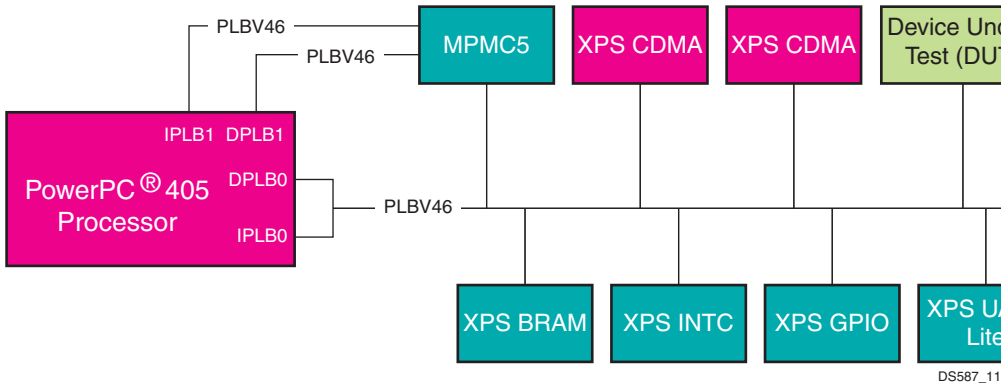

Figure 11: **Virtex-4 FX System**

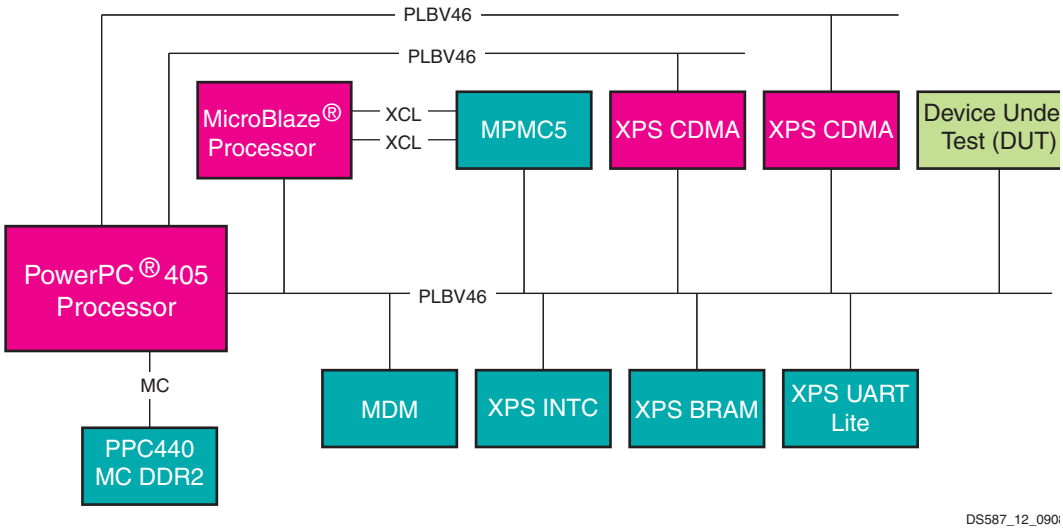

Figure 12: **Virtex-5 FX System**

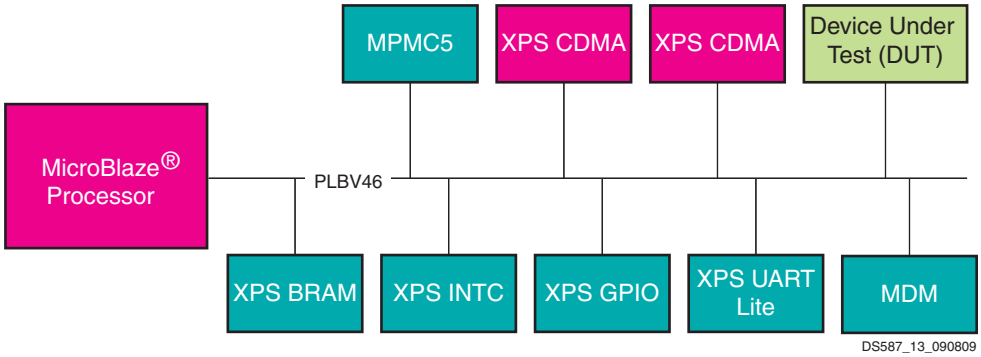

Figure 13: **Spartan-3A System**

The target FPGA was then filled with logic to drive the LUT and BRAM utilization to approximately 70% and the I/O utilization to approximately 80%. Using the default tool options and the slowest speed grade for the target FPGA, the resulting target F_{MAX} numbers are shown in Table 18.

Table 18: **XPS Delta-Sigma ADC Core System Performance**

Target FPGA	Target F_{MAX} (MHz)
S3ADSP400 -4	90
V4FX60 -10	100
V5FXT70 -1	120

The target F_{MAX} is influenced by the exact system and is provided for guidance. It is not a guaranteed value across all systems.

Specification Exceptions

N/A

Reference Documents

1. *Analog Devices Data Converter Reference Manual*, Volume I, 1992
2. *High Performance Stereo Bit-Stream DAC with Digital Filter*, R. Finck, IEEE Transactions on Consumer Electronics, Vol. 35, No. 4, Nov. 1989
3. XAPP155, *Virtex Analog to Digital Converter*
4. *IBM CoreConnect 128-Bit Processor Local Bus, Architectural Specification (v4.6)*

Support

Xilinx provides technical support for this LogiCORE product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Revision History

Date	Version	Description of Revisions
4/17/07	1.0	Initial Xilinx release.
4/20/07	1.0	Added SP-3 support.
10/1/07	1.2	Added FMax Margin "System Performance," page 17 section.
11/27/07	1.3	Added SP-3A DSP support.
1/11/08	1.4	Added Virtex-II Pro support.
3/24/08	1.5	Modified Table 2, page 5 for FSTM = 4.
4/17/08	1.6	Added Automotive Spartan-3, Automotive Spartan-3E, Automotive Spartan-3A, and Automotive Spartan-3 DSP support.
7/24/08	1.7	Added QPro Virtex-4 Hi-Rel and QPro Virtex-4 Rad Tolerant FPGA support.
9/26/08	1.8	Removed Virtex-II Pro support, modified the bit definition in table-10, row-2 and modified the XPS ADC Register Description section.
12/24/08	1.9	Modified Resource utilization tables.
4/24/09	2.0	Replaced references to supported device families and tool names with hyperlink to PDF file.
7/20/09	2.1	Added Performance and Resource Utilization Benchmarks for Virtex6 and Spartan6.
12/2/09	2.2	Listed supported devices families in LogiCORE Table; converted to new DS template.

Notice of Disclaimer