LogiCORE IP AXI System ACE Interface Controller (axi_sysace) (v1.01.a)

DS789 June 22, 2011 **Product Specification**

Introduction

The AXI System ACE™ Interface Controller (or, interchangeably, the AXI SYSACE) is the interface between AMBA® AXI4-Lite and the Microprocessor Unit (MPU) interface of the System ACE Compact Flash solution peripheral.

Features

- Connects as a 32-bit slave on AXI4-Lite Interface.
- The AXI SYSACE is used in conjunction with a System ACE Compact Flash Solution to provide a System ACE memory solution.
- System ACE Microprocessor Unit (MPU) Interface
  - Read/Write from or to a Compact Flash device
  - Supports both 8-bit and 16-bit data bus access modes

LogiCORE™ IP Facts				
Core Specifics				
Supported Device Family(1)	Artix-7, Virtex-7, Kintex-7 Virtex-6, Spartan-6			
Supported User Interfaces	AXI4-Lite			
Resources Used				
Slices	LUTs	FFs	Block RAMs	Frequency
See Table 6 through Table 10			0	See Table 6 through Table 10
Provided with Core				
Documentation	Product Specification			
Design File Formats	VHDL			
Constraints File	N/A			
Verification	N/A			
Instantiation Template	N/A			
Additional Items	N/A			
Design Tool Requirements				
Xilinx Implementation Tools	XPS 13.2			
Verification	Mentor Graphics ModelSim(2)			
Simulation	Mentor Graphics ModelSim(2)			
Synthesis	XPS 13.2			
Support				
Provided by Xilinx, Inc.				

1. For a complete list of supported derivative devices, please see the IDS Embedded Edition Derivative Device Support.
2. For the supported versions of the tools, see the ISE Design Suite 13: Release Notes Guide.

© Copyright 2010-2011 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners. ARM is a registered trademark of ARM in the EU and other countries. The AMBA trademark is a registered trademark of ARM Limited. All other trademarks are the property of their respective owners.

Functional Description

The AXI SYSACE is composed of the AXI4-Lite Interface module and the System ACE Interface Controller. The connections between the AXI System ACE Interface Controller, the AXI4-Lite Interface module, and the Xilinx System ACE Controller device are shown in Figure 1.

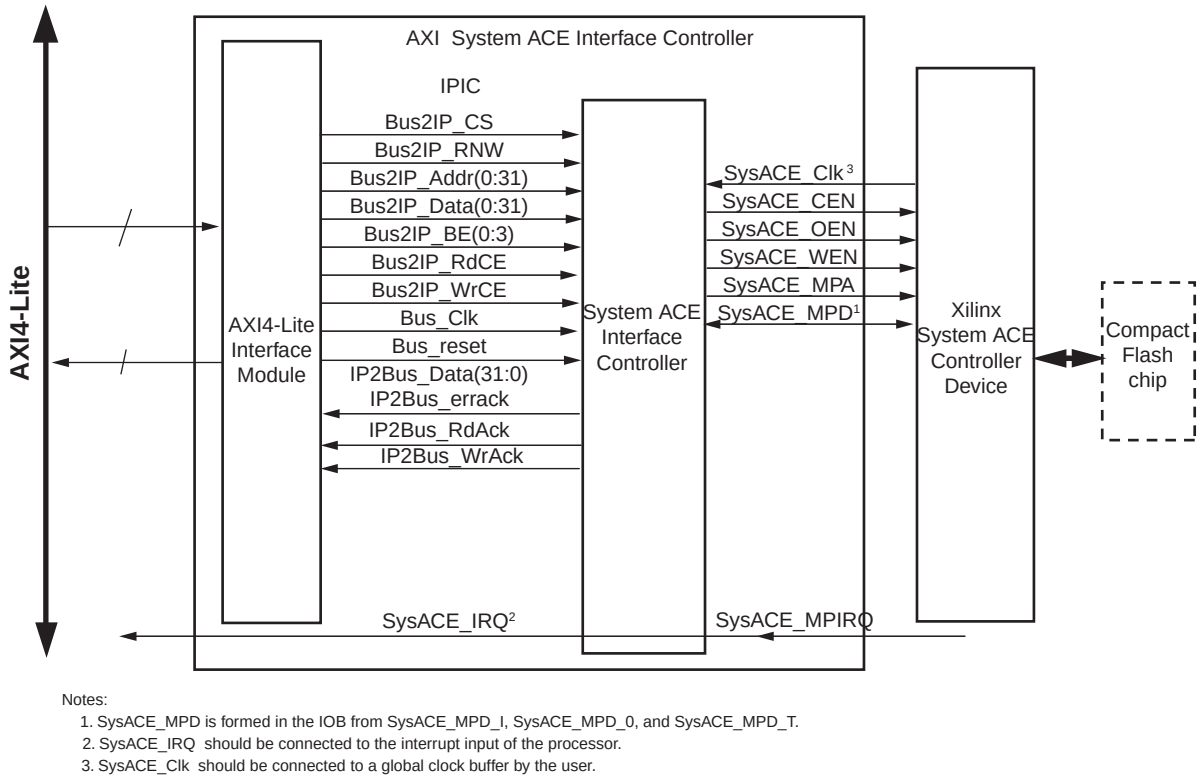
The AXI SYSACE provides the MPU interface to the Xilinx System ACE Controller Device. The Xilinx System ACE Controller device has multiple interfaces, including CompactFlash, MPU and JTAG. This allow for a highly flexible configuration solution. The MPU interface of the Xilinx System ACE Controller device is composed of a set of registers that provide a means for communicating with CompactFlash control logic, configuration control logic, and other resources in the Xilinx System ACE Controller device. Specifically, this interface can be used to read the identity of a CompactFlash device and read/write sectors. The AXI System ACE Interface Controller provides a means of communicating with the registers and data buffers that correspond to the CompactFlash device in the Xilinx System ACE Controller device, via the AXI. See the System ACE Interface Controller Flash chip document mentioned in the Reference Documents section for detailed information on the operation of the MPU interface, the MPU interface register definitions, and the MPU interface register address map.

The AXI System ACE Interface Controller allows for the registers and data buffers of the Xilinx System ACE Controller device, to be accessed in an 8-bit and 16-bit data bus access mode. The two modes are differentiated by the means of the parameter C_MEM_WIDTH, as follows:

- 8-bit mode (C_MEM _WIDTH = 8): The registers are accessed in a 8-bit data bus access mode. In this mode, the registers of the Xilinx System ACE Controller device should be accessed via byte accesses only.
- 16-bit mode (C_MEM _WIDTH = 16): The registers are accessed in a 16-bit data bus access mode. In this mode, the registers of the Xilinx System ACE Controller device should be accessed via halfword accesses only.

For example, a typical register like the Bus Mode register, is accessed by addresses “00h” and “01h” in the 8-bit access mode. It would be accessed by address “00h” in the 16-bit access mode.

The software drivers use the C_MEM_WIDTH parameter to configure the Xilinx System ACE Bus Mode register (setting the Xilinx System ACE MPU data bus access width to the desired mode) and to access the registers with the proper type of transaction.



Notes:
1. SysACE_MPD is formed in the IOB from SysACE_MPD_I, SysACE_MPD_0, and SysACE_MPD_T.
2. SysACE_IRQ should be connected to the interrupt input of the processor.
3. SysACE_Clk should be connected to a global clock buffer by the user.

Figure 1: **AXI System ACE Interface Controller Block Diagram**

AXI4-Lite Interface Module

AXI4-Lite Interface Module provides an interface between AXI System ACE Interface Controller and the AXI. The AXI4-Lite Interface Module implements the basic functionality of an AXI slave and does the necessary protocol and timing translation between the AXI and the IPIC interface. The AXI4-Lite Interface Module supports only single beat transactions.

System ACE Interface Controller

The System ACE interface controller contains a controller state machine and logic to synchronize signals across the `S_AXI_ACLK` and `SysACE_Clk` domains as shown in Figure 2.

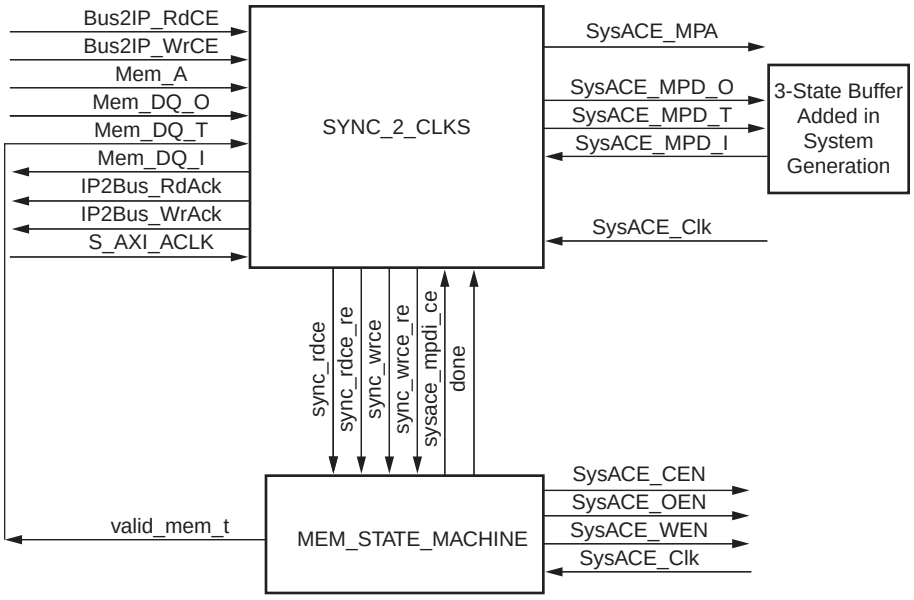


Figure 2: **System ACE Interface Controller Diagram**

The AXI System ACE Interface Controller core does not contain any internal registers or addressable memory space; therefore, the mapping of the AXI address bus is one-to-one with the System ACE address bus (SysACE_MPA) as shown in Table 1.

Table 1: **AXI Address Bus to System ACE Address Bus Mapping (Done in IP Core)**

AXI Address Bus	System ACE Address Bus
AXI_ABus[6 : 0]	SysACE_MPA[6 : 0]

The Xilinx System ACE Compact Flash chip and AXI are both little-endian, therefore the AXI System ACE Interface Controller has one-to-one mapping as shown in Table 2.

Table 2: **AXI Data Bus to System ACE Data Bus Mapping (Done in IP Core)**

AXI Data Bus	System ACE Data Bus
AXI_DBus[15 : 8]	SysACE_MPD[15 : 8]
AXI_DBus[7 : 0]	SysACE_MPD[7 : 0]

Clocking - SYNC_2_CLKS Module

The controller state machine runs on the SysACE_Clk. The IPIC signals indicating the start of a transaction are synchronized to the System ACE clock and used to start the state machine. All address, data and control signals that are output to the System ACE Compact Flash chip are synchronized to the SysACE_Clk and registered in the FPGA IO registers using SysACE_Clk to ensure a clean interface between this chip and the FPGA. Data from the System ACE Compact Flash chip is also registered in FPGA IO registers using SysACE_Clk. It is then synchronized to the S_AXI_ACLK for transmission on the bus. The frequency of the SysACE_Clk must be less than the frequency of the S_AXI_ACLK.

Note that the address and data (if a write transaction) from the AXI will stay stable during the entire bus transaction and therefore would not have to be synchronized and output using the SysACE_Clk. This was done to provide a robust design, however, if the overall FPGA design is limited on resources, these synchronization registers could possibly be removed. The user is cautioned to analyze timing before removing these registers.

Also note that this core does not instantiate a global clock buffer for SysACE_Clk. This is left for the user to instantiate based on the resource requirements of their system.

System ACE Control State Machine - MEM_STATE_MACHINE Module

The state machine in the System ACE Interface controller performs the specified transaction to the MPU interface of the System ACE Compact Flash chip and is shown in Figure 3. This state machine is clocked by SysACE_Clk and therefore outputs all System ACE control signals synchronous to this clock. The input control signals from the AXI4-Lite Interface Module have been synchronized to the SysACE_Clk in the sync_2_clocks module.

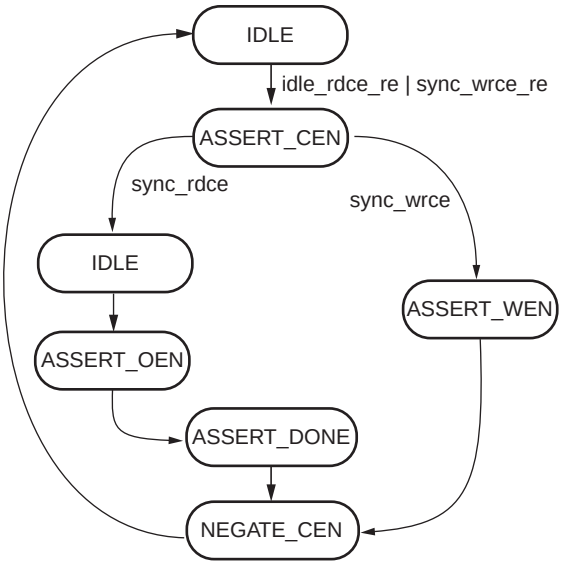


Figure 3: **System ACE Interface Control State Machine**

AXI System ACE Interface Controller I/O Signals

The I/O signals for the AXI System ACE Interface Controller are listed and described in Table 3.

Table 3: **AXI System ACE Interface Controller I/O Signals**

Port	Signal Name	Interface	I/O	Initial State	Description
System Signals					
P1	S_AXI_ACLK	AXI	I	-	AXI clock
P2	S_AXI_ARESETn	AXI	I	-	AXI reset, active low
AXI Write Address Channel Signals					
P3	S_AXI_AWADDR [C_S_AXI_ADDR_WIDTH - 1:0]	AXI	I	-	AXI Write address. The write address bus gives the address of the write transaction.
P4	S_AXI_AWVALID	AXI	I	-	Write address valid. This signal indicates that valid write address is available.
P5	S_AXI_AWREADY	AXI	O	0x0	Write address ready. This signal indicates that the slave is ready to accept an address.
AXI Write Data Channel Signals					
P6	S_AXI_WDATA [C_S_AXI_DATA_WIDTH - 1:0]	AXI	I	-	Write Data
P7	S_AXI_WSTB [C_S_AXI_DATA_WIDTH/8-1:0]	AXI	I	-	Write strobes. This signal indicates which byte lanes to update in memory
P8	S_AXI_WVALID	AXI	I	-	Write valid. This signal indicates that valid write data and strobes are available.
P9	S_AXI_WREADY	AXI	O	1'b0	Write ready. This signal indicates that the slave can accept the write data.
AXI Write Response Channel Signals					
P10	S_AXI_BRESP[1:0]	AXI	O	0x0	Write response. This signal indicates the status of the write transaction. "00" - OKAY "10" - SLVERR
P11	S_AXI_BVALID	AXI	O	1'b0	Write response valid. This signal indicates that a valid write response is available.
P12	S_AXI_BREADY	AXI	I	-	Response ready. This signal indicates that the master can accept the response information.
AXI Read Address Channel Signals					
P13	S_AXI_ARADDR [C_S_AXI_ADDR_WIDTH-1:0]	AXI	I	-	Read Address. The read address bus gives the address of a read transaction.
P14	S_AXI_ARVALID	AXI	I	-	Read address valid. This signal indicates, when HIGH, that the read address is valid and will remain stable until the address acknowledgement signal, S_AXI_AREADY, is high.
P15	S_AXI_ARREADY	AXI	O	1'b1	Read address ready. This signal indicates that the slave is ready to accept an address

Table 3: **AXI System ACE Interface Controller I/O Signals *(Cont'd)***

Port	Signal Name	Interface	I/O	Initial State	Description
AXI Read Data Channel Signals					
P16	S_AXI_RDATA[C_S_AXI_DATA_WIDTH-1:0]	AXI	O	0x0	Read Data
P17	S_AXI_RRESP[1:0]	AXI	O	0x0	Read response. This signal indicates the status of the read transfer: "00" - OKAY "10" - SLVERR
P18	S_AXI_RVALID	AXI	O	1'b0	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
P19	S_AXI_RREADY	AXI	I	-	Read ready. This signal indicates that the master can accept the read data and response information.
Signals					
P20	SysACE_Clk(1)	System Ace Core	I	-	System ACE Clock
P21	SysACE_MPIRQ	System Ace Core	I	-	System ACE Active high Interrupt Input
P22	SysACE_CEN	System Ace Core	O	1	System ACE Chip Enable
P23	SysACE_OEN	System Ace Core	O	1	System ACE Output Enable
P24	SysACE_WEN	System Ace Core	O	1	System ACE Write Enable
P25	SysACE_MPA[6 : 0]	System Ace Core	O	0	System ACE Address
P26	SysACE_MPD_I[C_MEM_WIDTH-1 : 0]	System Ace Core	I	-	System ACE Data Input
P27	SysACE_MPD_O[C_MEM_WIDTH-1 : 0]	System Ace Core	O	0	System ACE Data Output
P28	SysACE_MPD_T[C_MEM_WIDTH-1 : 0]	System Ace Core	O	1	System ACE Data Output enable
P29	SysACE_IRQ(2)	System Ace Core	O	0	System ACE Active High Interrupt Output

Note:

1. S_AXI_ACLK frequency must be greater than or equal to SysACE_Clk Frequency.
2. This interrupt output is just a pass-through of the System ACE interrupt (SysACE_MPIRQ) and should be connected to an interrupt controller or directly to the processor's interrupt input.

AXI System ACE Interface Controller Design Parameters

To allow the designer to obtain a AXI SYSACE core that is uniquely tailored for the designer's system, certain features can be parameterized. Some of these parameters control the interface to the AXI interface module while others provide information to minimize resource utilization. The features that can be parameterized in the AXI SYSACE are shown in Table 4.

In addition to the parameters listed in this table, there are also parameters that are inferred for each AXI interface in the EDK tools. Through the design, these EDK-inferred parameters control the behavior of the AXI Interconnect. For a complete list of the interconnect settings related to the AXI interface, see DS768, *AXI Interconnect IP Data Sheet*.

Table 4: **AXI System ACE Interface Controller Parameters**

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
System Parameter					
G1	Target FPGA family	C_FAMILY	virtex6, spartan6	virtex6	string
AXI Parameters					
G2	AXI System ACE Base Address	C_BASEADDR	Valid Address[1]	0xFFFFFFFF[1]	std_logic_vector
G3	AXI System ACE High Address	C_HIGHADDR	Valid Address[1]	0x00000000[1]	std_logic_vector
G4	AXI address bus width	C_S_AXI_ADDR_WIDTH	32	32	integer
G5	AXI data bus width	C_S_AXI_DATA_WIDTH	32	32	integer
System ACE Parameters					
G6	System ACE MPU Data Bus Access Mode[2]	C_MEM_WIDTH	8, 16	16	integer

Note:

1. The range specified by C_BASEADDR and C_HIGHADDR must be sized and aligned to some power of 2, 2^n. Then, the n least significant bits of C_BASEADDR is zero. This range needs to encompass the addresses needed by the AXI SYSACE registers
2. See Xilinx DS080, System ACE Compact Flash Solution, for more information.

Allowable Parameter Combinations

The address-range size of the AXI System ACE Interface Controller must be a power of 2. If the desired address-range size is represented by 2^n, then the n least significant bits of the base address must be 0. C_BASEADDR and C_HIGHADDR must specify an address range whose size is at least 0x80 bytes, to cover the addressable registers and data buffer available in the Xilinx System Ace Compact Flash chip.

AXI System ACE Interface Controller Parameter-Port Dependencies

The dependencies between the AXI System ACE Interface Controller 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: **AXI System ACE Interface Controller Parameter-Port Dependencies**

Generic or Port	Parameter	Affects	Depends	Relationship Description
Design Parameters				
G4	C_S_AXI_ADDR_WIDTH	P3,P13	-	Width of the AXI Address Bus
G5	C_S_AXI_DATA_WIDTH	P6,P7,P16	-	Width of the AXI Data Bus
I/O Signals				
P3	S_AXI_AWADDR[C_S_AXI_ADDR_WIDTH-1:0]	-	G4	Width varies with the width of the AXI Address Bus width
P6	S_AXI_WDATA[C_S_AXI_DATA_WIDTH-1:0]	-	G5	Width varies with the AXI data bus width
P7	S_AXI_WSTB[C_S_AXI_DATA_WIDTH/8-1:0]	-	G5	Width varies with the width of the AXI Data Bus width
P13	S_AXI_ARADDR[C_S_AXI_ADDR_WIDTH-1:0]	-	G4	Width varies with the width of the AXI Address Bus width
P16	S_AXI_RDATA[C_S_AXI_DATA_WIDTH-1:0]	-	G5	Width varies with the width of the AXI Data Bus width

Design Implementation

Target Technology

The intended target technology is the Artix™-7, Virtex®-7, Kintex™-7, Virtex-6 and Spartan®-6 FPGAs.

Device Utilization and Performance Benchmarks

Core Performance

Since the AXI System ACE Controller will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are just estimates. When the AXI System ACE Interface Controller is combined with other designs in the system, the utilization of FPGA resources and timing will vary from the results reported here.

The AXI System ACE Interface Controller benchmarks are shown in Table 6 through Table 10.

Table 6: **Performance and Resource Utilization Benchmarks for Artix-7 FPGAs (XC7A355TDIE)**

Parameter Values			Device Resources			Performance
C_MEM_WIDTH	C_BASEADDR	C_HIGHADDR	Slices	Slice Flip-Flops	LUTs	F_{max} (in MHz)
8	0x30000000	0x3FFFFFFF	49	90	81	247.709
16	0x30000000	0x3FFFFFFF	53	114	75	211.954

Table 7: **Performance and Resource Utilization Benchmarks for Virtex-7 FPGAs (XC7V855T-FFG1157-3)**

Parameter Values			Device Resources			Performance
C_MEM_WIDTH	C_BASEADDR	C_HIGHADDR	Slices	Slice Flip-Flops	LUTs	F_{max} (in MHz)
8	0x30000000	0x3FFFFFFF	62	114	74	202.102
16	0x30000000	0x3FFFFFFF	62	114	74	202.102

Table 8: **Performance and Resource Utilization Benchmarks for Kintex-7 FPGAs (XC7K410T-FFG676-3)**

Parameter Values			Device Resources			Performance
C_MEM_WIDTH	C_BASEADDR	C_HIGHADDR	Slices	Slice Flip-Flops	LUTs	F_{max}(in MHz)
8	0x30000000	0x3FFFFFFF	54	90	81	211.282
16	0x30000000	0x3FFFFFFF	60	114	75	207.297

Table 9: **Performance and Resource Utilization Benchmarks for Virtex-6 FPGAs (XC6VLX195T-1-FF1156)**

Parameter Values			Device Resources			Performance
C_MEM_WIDTH	C_BASEADDR	C_HIGHADDR	Slices	Slice Flip-Flops	LUTs	F_{max} (in MHz)
8	0x30000000	0x3FFFFFFF	44	136	64	243.962
16	0x30000000	0x3FFFFFFF	46	160	75	235.46

Table 10: **Performance and Resource Utilization Benchmarks for Spartan-6 FPGAs (XC6SLX45-2-FGG484)**

Parameter Values			Device Resources			Performance
C_MEM_WIDTH	C_BASEADDR	C_HIGHADDR	Slices	Slice Flip-Flops	LUTs	F_{max} (in MHz)
8	0x30000000	0x3FFFFFFF	58	183	72	156.986
16	0x30000000	0x3FFFFFFF	70	207	67	161.734

System Performance

To measure the system performance (Fmax) of this core, this core was added to a Virtex-6 FPGA system, and a Spartan-6 FPGA system as the Device Under Test (DUT).

Because the AXI SYSACE Controller 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.

The target FPGA was then filled with logic to drive the LUT and block RAM 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 FMax numbers are shown in Table 11.

Table 11: **AXI SYSACE Controller Core System Performance**

Target FPGA	Target f_{MAX} (MHz)
Artix-7	110
Virtex-7	180
Kintex-7	180
Virtex-6	180
Spartan-6	110

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

Ordering Information

This Xilinx LogiCORE IP module is provided at no additional cost with the Xilinx ISE® Design Suite Embedded Edition software under the terms of the Xilinx End User License. The core is generated using the Xilinx ISE Embedded Edition software (EDK).

Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE modules and software, please contact your local Xilinx sales representative.

Reference Documents

The following documents contain reference information important to understanding the AXI SYSACE Controller design:

- *AMBA AXI Protocol Version: 2.0 Specification* (ARM® IHI 0022C)
- WP151, *System ACE Configuration Solution for Xilinx FPGAs*
- DS080, *System ACE Compact Flash Solution*
- DS768, *AXI Interconnect IP Data Sheet*

List of Acronyms

Acronym	Meaning
ACE	Advanced Configuration Environment
AMBA	Advanced Microcontroller Bus Architecture
ARM	Advanced RISC Machine
AXI	Advanced eXtensible Interface
DUT	Device Under Test
FF	Flip-Flop
FPGA	Field Programmable Gate Array
IO	Input/Output

Acronym	Meaning
IP	Intellectual Property
IPIC	IP Interconnect
JTAG	Joint Test Action Group
LUT	Lookup Table
MHz	Mega Hertz
MPU	Microprocessor Unit
RAM	Random Access Memory
VHDL	VHSIC Hardware Description Language (VHSIC an acronym for Very High-Speed Integrated Circuits)

Revision History

Date	Version	Revision
09/23/10	1.0	Initial Release of the core
12/14/10	1.1	Updated to v1.01.a; updated tools to 12.4.
06/22/11	2.0	Updated for XPS v13.2. Added support for Artix-7, Virtex-7, and Kintex-7 devices.

Notice of Disclaimer

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at http://www.xilinx.com/warranty.htm; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications: http://www.xilinx.com/warranty.htm#critapps.