LogiCORE IP AXI4-Stream Interconnect v1.0

Product Guide

PG035 April 24, 2012

XILINX®

Table of Contents

LogiCORE IP AXI4-Stream Interconnect v1.0

Introduction

The AXI4-Stream Interconnect is a key interconnect infrastructure IP that enables connection of heterogenous master/slave AMBA® AXI4-Stream protocol compliant endpoint IP. The AXI4-Stream Interconnect routes connection from one or more AXI4-Stream master channels to one or more AXI4-Stream slave channels.

Features

The AXI4-Stream Interconnect IP provides the following capabilities:

- Configurable multiple master to multiple slave (up to 16x16) capable cross-point switch
- Arbitrary TDATA byte width conversion
- Synchronous and asynchronous clock rate conversion
- Configurable datapath FIFO buffers including store and forward (packet) capable FIFOs
- Optional register slice at boundaries to ease timing closure
- Support for multiple clock domains

LogiCORE™ IP Facts Table	
Core Specifics	
Supported Devices(1)	Zynq™-7000, Artix™-7, Kintex™-7, Virtex®-7, Virtex-6, Spartan®-6
Supported User Interfaces	AXI4-Stream
Resources	To Be Supplied
Provided with Core	
Design Files	Verilog
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	Xilinx Constraints File
Simulation Model	Behavioral Verilog
Supported S/W Driver	N/A
Tested Design Tools	
Design Entry Tools(2)	CORE Generator™ tool
Simulation(2)	Questa (R)
Synthesis Tools(2)	Xilinx Synthesis Technology (XST)
Support	
Provided by Xilinx @ www.xilinx.com/support	

Notes:

1. For a complete list of supported derivative devices, see the IDS Embedded Edition Derivative Device Support.
2. For the supported versions of the tools, see the ISE Design Suite 14: Release Notes Guide.

Overview

The AXI4-Stream Interconnect is a collection of submodules centered around the AXI-S Switch. There are submodules grouped together in datapaths before and after the switch that allow for minor data manipulation and flow control. Each individual submodule consists of AXI4-Stream protocol compliant master and slave interfaces. A block diagram of the AXI-S Interconnect is shown in Figure 1-1.

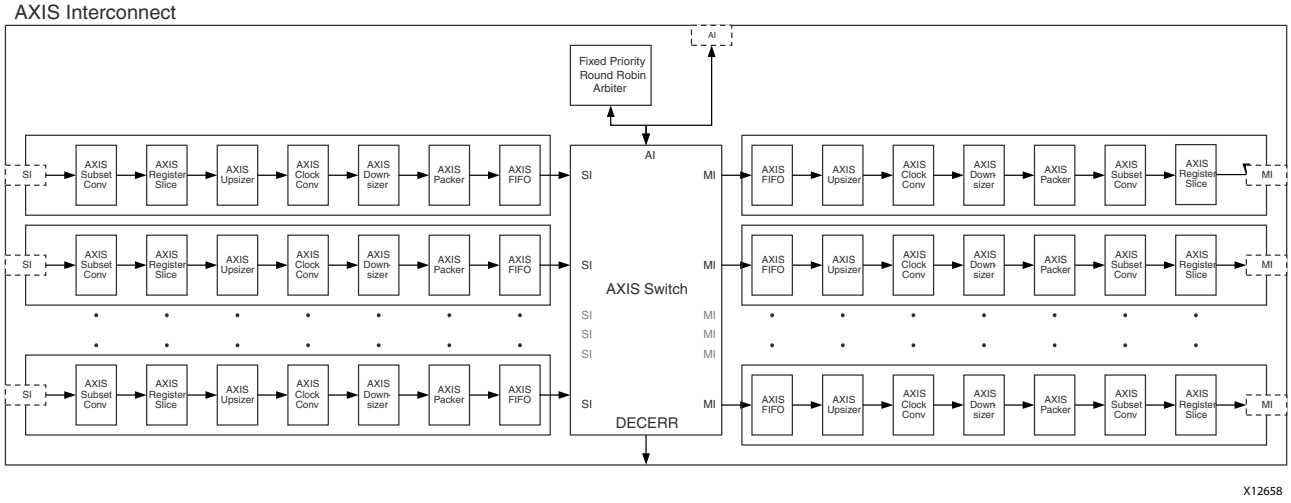

Figure 1-1: **AXI-S Interconnect Block Diagram**

The AXI-S Switch supports up to 16 Masters to 16 Slaves in a full or sparse crossbar configuration using the AXI4-Stream signal `TDEST` as the routing designator.

In Figure 1-1 an AXI-S Master can connect to what is designated as the Slave Interface (SI) bus of the AXI-S Interconnect. Similarly, an AXI-S Slave can connect to the Master Interface (MI) bus of the AXI-S Interconnect. The AXI-S Switch supports sparse configurations, SI side decoding, MI side arbitration (one arbiter per MI port) and various arbitration modes.

The datapath consists of submodules that contain one SI port and one MI port chained together between the external SI bus to the switch or from the switch to the external MI bus. The datapath allows for data width conversion, buffering, and clock conversion to and from the AXI-S Switch. All submodules including the AXI-S Switch are optionally instantiated based on parameters to allow full flexibility in design.

The submodules within the AXI-S Interconnect are presented in Table 1-1.

Table 1-1: **Submodules within the AXI-S Interconnect**

Submodule Name	Description	Inferred or Explicit instantiation
Subset Converter	Adds or removes optional signals from an AXI-S bus interface.	Limited inference. Infers a TKEEP internally to the interconnect if there is a data width converter and TID or TDEST or TLAST signals are present.
Register Slice	Provides a mechanism to bridge connections between Masters/Slaves within different ACLK domains.	Inferred
Data Width Converter	Allows expansion of the AXI-S TDATA width by aggregating multiple transfers into one transfer or allows reduction of the AXI-S TDATA width by splitting a transfer into multiple transfers of smaller TDATA width.	Inferred
Packer	Removes NULL bytes from a byte stream by merging multiple transfers together in place of the NULL bytes.	Not implemented
Data FIFO	Provides AXI-S data storage.	Explicit
Switch	Allows routing from multiple masters to multiple slaves.	Inferred when there is more than one master or more than one slave in the system.

Feature Summary

AXI-S Switch and Arbiter

The AXI-S Switch supports 1:N, M:1 and M:N configurations. It connects up to 16 masters to 16 slaves. The AXI-S TDEST signaling is required for 1:N and M:N configurations.

The AXI-S Switch only supports static routing through fixed base-high TDEST ranges for each MI. A single TDEST map applies to all MI. Each MI is arbitrated independently (slave side arbitration). The AXI-S Switch performs SI-side parallel decoding. Unmapped TDEST transfers will signal a decode error and drop the transfer.

Internal arbiter can perform fixed priority arbitration or round robin arbitration. The Arbiter/Switch can arbitrate on a per transfer basis or at packet boundaries (signaled by TLAST or a configurable number of transfers.)

AXI-S Clock Converter

Clock converters are necessary in the AXI-S protocol for converting masters at different clock rates to slaves. Typically, the AXI-S Interconnect wants to be clocked at the same rate as the fastest slave and devices not running at the same rate need to be converted. Synchronous clock converters are ideal because they have the lowest latency and smaller area. However, they are only viable if both clocks are phase-aligned and the fmax requirements are able to be met. Asynchronous clock converters are a generic solution able to handle both synchronous/asynchronous clocks with arbitrary phase alignment. The trade-off is that there is a significant increase in area and latency associated with asynchronous clock converters. Clock converter support will be a module block shared with Memory Mapped Interconnect. There is a Clock Converter module available in every datapath and it is instantiated if either the clocks are specified as asynchronous or have different synchronous clock ratios. The clock converter module performs the following functions:

- A clock-rate reduction module performs integer (N:1) division of the clock rate from its input (SI) side to its output (MI) side.
- A clock-rate acceleration module performs integer (1:N) multiplication of clock rate from its input (SI) to output (MI) side.
- Asynchronous clock rate conversion between the input and output using a FIFO Generator instantiated module.

Figure 1-2 shows the clock converter with support for independent `ACLKEN` signals on its SI and MI.

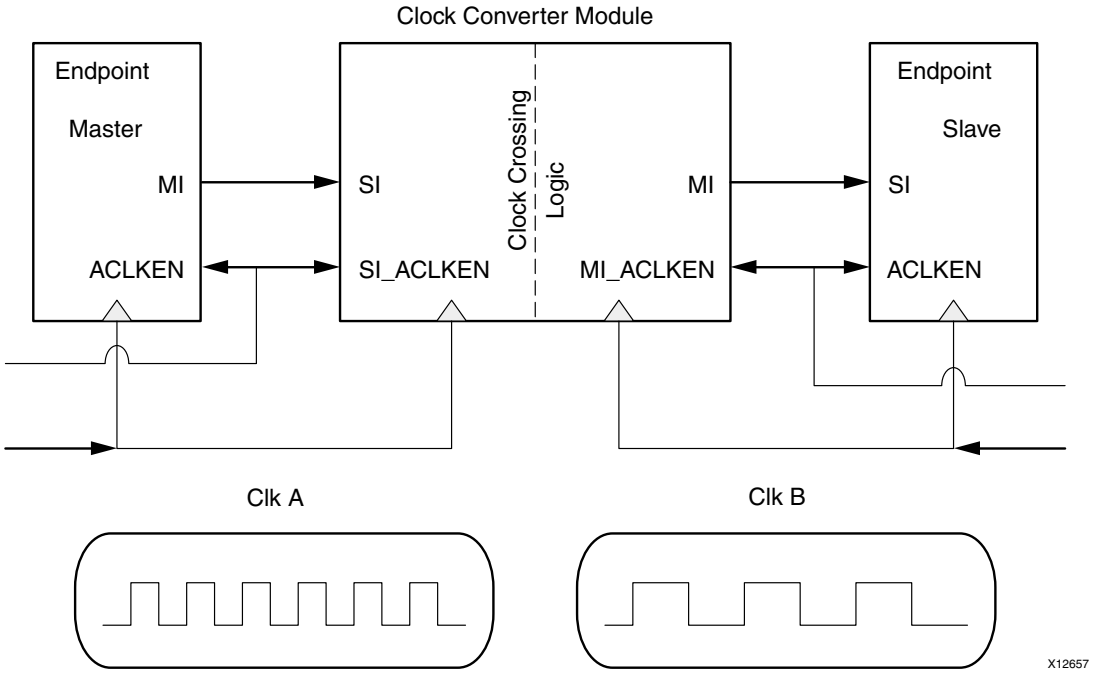

Figure 1-2: **Clock Converter Module Block Diagram**

AXI-S Data Width Converter

Data width converters (upsizer/downsizer) are required when interfacing different data width cores with each other. One data width conversion module is available to handle all supported combinations of data widths.

The conversion follows the AMBA® AXI4-Stream Protocol Specification with regards to ordering and expansion of TUSER bits. The width converter does not process any TUSER signaling of straddled packets. The number of TUSER bits per TDATA bytes must remain constant between input and output.

Up-conversion requires that each incoming beat that is composed of the new larger beat will consist of identical TID and TDEST bits and no intermediate TLAST assertions. Partial data might be flushed when either the TLAST bit is received or TID/TDEST changes before enough data is accumulated to send out a complete beat. Unassigned bytes are flushed out as null bytes. It is recommended that the `TKEEP` signal output is monitored if `TID`/`TDEST`/`TLAST` signal is present.

Any non-integer multiple byte ratio conversion (N:M) is accomplished by calculating the lowest common multiple (LCM) of N and M and then up-converting from N:LCM then down-converting from LCM:M.

Up-conversion features:

- Range: Input 1-256 Bytes, Output 2-512 Bytes
- Supports full range of 1:N byte ratio conversions
- Minimum latency of 2 + N clock cycles in 1:N byte ratio up-conversion.

Down-conversion features:

- Range: Input 2-512 bytes, output 256-1 bytes
- Supports full range of N:1 byte ratio conversions
- Minimum Latency: 2 clock cycles

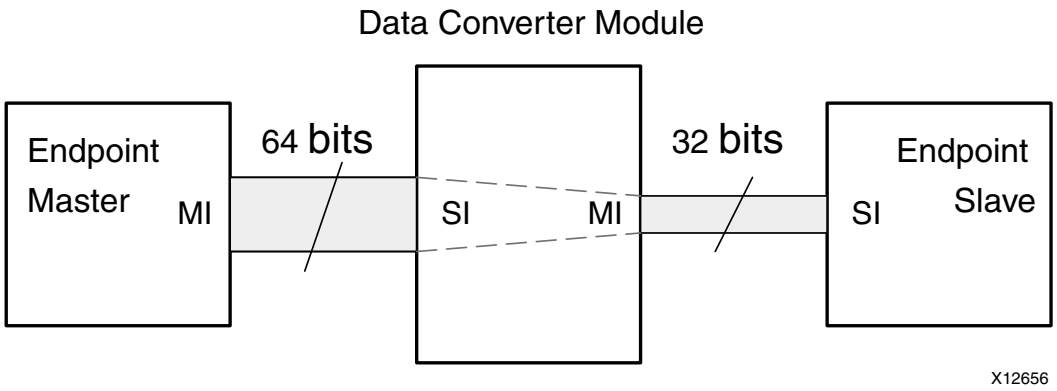

Figure 1-3: **Data Width Converter (Down Conversion) Module Block Diagram**

AXI-S Register Slice

The register slice is a multipurpose pipeline register that is able to isolate timing paths between master and slave while maintaining a minimum amount of buffer to not largely impact the area or latency of the system. Implemented as a two-deep FIFO buffer, this mode supports throttling by the master (channel source) and/or slave (channel destination) as well as back-to-back transfers without incurring dead cycles. The module can be independently instantiated at all port boundaries.

AXI-S Data FIFO Buffer

The FIFO module is capable of providing temporary storage (a buffer) of the AXI4-S stream. The FIFO Buffer module should be used in between two endpoints when:

- More buffering than a register slice is desired.
- Asynchronous clock conversions required. This functionality is inferred in the clock converter module in a fixed FIFO configuration if no explicit FIFO is specified.
- Store and forward: to accumulate a certain number of bytes from the master before forwarding them to the slave (packet mode.)

The FIFO module uses the Xilinx LogiCORE™ IP FIFO Generator module. This supports native AXI4-Stream with the following features:

- Variable FIFO depths
- FIFO data widths from 8 to 1024 bits

  1024-bit FIFO data width limit is a FIFOGen restriction that limits the payload width of the transfer being buffered in the FIFO. This limit might be removed in a future generation of FIFOGen, at which point the AXIS FIFO also supports FIFO buffering for transfers with payloads including TDATA widths of up to 4096 bits.
- Independent or common clock domains
- Symmetric aspect ratios
- Asynchronous active-Low reset
- Selectable memory type. The memory type will be inferred as distributed block RAM for depths of 32 or less and block RAM for all others.
- Operates in First-Word Fall-Through mode (FWFT)
- Occupancy interface
- Both FIFOGen rd_data_count and wr_data_count are passed as separate outputs synchronized to the read side and write side clock domains.

Licensing

This Xilinx LogiCORE™ IP module is included at no additional charge with the Xilinx Integrates Software Environment (ISE®) Design Suite and is provided under the terms of the Xilinx End User License Agreement. The core is generated using the ISE Design Suite CORE Generator™ tools.

For more information, visit the AXI4-Stream Interconnect product page. Information about additional LogiCORE IP modules can be found on the Xilinx.com Intellectual Property page.

Product Specification

Standards Compliance

This core has bus interfaces that comply with the ARM® AMBA® AXI4-Stream Protocol Specification Version 1.0.

Port Descriptions

Global Signals

These signals are always present on every configuration.

Table 2-1: **Global Signals**

Signal	Direction	Description
ACLK	Input	Global Clock Signal. Drives the clocks on the AXI-S Switch and is the primary clock to the system.
ARESETN	Input	Global Reset Signal. This active-Low signal drives the reset pins on the AXI-S Switch and is the primary reset of the system.
ACLKEN	Input	Global ACLK Enable signals. Drives the ACLKEN pins on the AXI-S Switch and is the primary ACLKEN of the system.

Slave Interface Signals

The following table lists the signals associated with the Slave Interface buses. The number of buses is configurable and the signals in Table 2-2 will be replicated for each port. The nn denoted for the signals starts at 00 and increments by one up to 15 for each Slave Interface bus instantiated.

Table 2-2: **Signals Associated with the Slave Interface Buses**

Signal	Direction	Description
Snn_AXIS_ACLK	Input	Clock signal. All inputs/outputs of this bus interface are rising edge aligned with this clock.
Snn_AXIS_ARESETN	Input	Active-Low synchronous reset signal
Snn_AXIS_ACLKEN	Input	Clock enable signal
Snn_AXIS_TVALID(1)	Input	TVALID indicates that the master is driving a valid transfer. A transfer takes place when both TVALID and TREADY are asserted.
Snn_AXIS_TREADY(1)	Output	TREADY indicates that the slave can accept a transfer in the current cycle.
Snn_AXIS_TDATA [(C_Mnn_AXIS_TDATA_WIDTH-1):0](1)	Input	TDATA is the primary payload that is used to provide the data that is passing across the interface. The width of the data payload is an integer number of bytes.
Snn_AXIS_TSTRB [((C_Mnn_AXIS_TDATA_WIDTH/8)-1):0](1)	Input	TSTRB is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as a data byte or a position byte.
Snn_AXIS_TKEEP [((C_Mnn_AXIS_TDATA_WIDTH/8)-1):0](1)	Input	TKEEP is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as part of the data stream. Associated bytes that have the TKEEP byte qualifier deasserted are null bytes and can be removed from the data stream.
Snn_AXIS_TLAST(1)	Input	TLAST indicates the boundary of a packet.
Snn_AXIS_TID [C_NATIVE_TID_WIDTH-1:0](1)	Input	TID is the data stream identifier that indicates different streams of data.
Snn_AXIS_TDEST [(C_NATIVE_TDATA_WIDTH-1):0](1)	Input	TDEST provides routing information for the data stream.
Snn_AXIS_TUSER [(C_Snn_AXIS_TUSER_WIDTH-1):0](1)	Input	TUSER is user-defined sideband information that can be transmitted alongside the data stream.
Snn_ARB_REQ_SUPPRESS	Input	Active-High signal to skip this bus on the next arbitration cycle. While the signal is asserted, this bus does not receive the next arbitration. If this bus already has arbitration granted, it remains granted until the arbitration cycle is completely normally.

Table 2-2: **Signals Associated with the Slave Interface Buses *(Cont'd)***

Signal	Direction	Description
Snn_DECODE_ERR	Output	One-hot output indicates that a incoming transfer has a TDEST value that not map to a valid Master Interface. Invalid TDEST transfer are dropped.
Snn_SPARSE_TKEEP_REMOVED	Output	Not Implemented
Snn_PACKER_ERR	Output	Not Implemented
Snn_FIFO_DATA_COUNT[31:0]	Output	Indicates the write count inside the DATA FIFO. Does not produce valid output when using Packet Mode FIFO.

1. This signal description is taken from the ARM AMBA Protocol Specification.

Master Interface Signals

Table 2-3 lists the signals associated with the Master Interface buses. The number of buses is configurable and the signals will be replicated for each port. The nn denoted for the signals starts at 00 and increments by one up to 15 for each Master Interface bus instantiated.

Table 2-3: **Signals Associated with the Master Interface Buses**

Signal	Direction	Description
Mnn_AXIS_ACLK	Input	Clock signal. All inputs/outputs of this bus interface are rising edge aligned with this clock.
Mnn_AXIS_ARESETN	Input	Active-Low synchronous reset signal
Mnn_AXIS_ACLKEN	Input	Clock enable signal
Mnn_AXIS_TVALID(1)	Input	TVALID indicates that the master is driving a valid transfer. A transfer takes place when both TVALID and TREADY are asserted.
Mnn_AXIS_TREADY(1)	Output	TREADY indicates that the slave can accept a transfer in the current cycle.
Mnn_AXIS_TDATA [(C_Mnn_AXIS_TDATA_WIDTH-1):0](1)	Input	TDATA is the primary payload that is used to provide the data that is passing across the interface. The width of the data payload is an integer number of bytes.
Mnn_AXIS_TSTRB [((C_Mnn_AXIS_TDATA_WIDTH/8)-1):0](1)	Input	TSTRB is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as a data byte or a position byte.

Table 2-3: **Signals Associated with the Master Interface Buses *(Cont'd)***

Signal	Direction	Description
Mnn_AXIS_TKEEP [((C_Mnn_AXIS_TDATA_WIDTH/8)-1):0][(1)]	Input	TKEEP is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as part of the data stream. Associated bytes that have the TKEEP byte qualifier deasserted are null bytes and can be removed from the data stream.
Mnn_AXIS_TLAST[(1)]	Input	TLAST indicates the boundary of a packet.
Mnn_AXIS_TID [C_NATIVE_TID_WIDTH-1:0][(1)]	Input	TID is the data stream identifier that indicates different streams of data.
Mnn_AXIS_TDEST [(C_NATIVE_TDATA_WIDTH-1):0][(1)]	Input	TDEST provides routing information for the data stream.
Mnn_AXIS_TUSER [(C_Snn_AXIS_TUSER_WIDTH-1):0][(1)]	Input	TUSER is user-defined sideband information that can be transmitted alongside the data stream.
Mnn_SPARSE_TKEEP_REMOVED	Output	One-hot output indicates that at least one Null Byte was introduced into TDATA due to width conversion, but because the TKEEP signal is not present, the Null Byte will not be detected by the downstream slave.
Mnn_PACKER_ERR	Output	Not Implemented
Mnn_FIFO_DATA_COUNT[31:0]	Output	Indicates the write count inside the DATA FIFO. Does not produce valid output when using Packet Mode FIFO.

1. This signal description is taken from the ARM AMBA Protocol Specification.

Appendix A

Additional Resources

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see the Xilinx Support website at:

www.xilinx.com/support.

For a glossary of technical terms used in Xilinx documentation, see:

www.xilinx.com/company/terms.htm.

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Technical Support

Xilinx provides technical support at www.xilinx.com/support for this LogiCORE™ IP 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.

Ordering Information

The core is provided under the Xilinx End User License Agreement and can be generated using the Xilinx® CORE Generator™ system. The CORE Generator system is shipped with Xilinx ISE® Design Suite software.

Contact your local Xilinx sales representative for pricing and availability of additional Xilinx LogiCORE IP modules and software. Information about additional Xilinx LogiCORE IP modules is available on the Xilinx IP Center.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/24/12	1.0	Initial Release

Notice of Disclaimer