XAUI v6.2

DS265 July 13, 2006 **Product Specification**

Introduction

The LogiCORE™ XAUI core is a high-performance, low pin count 10-Gbps interface intended to allow physical separation between data-link layer and physical layer devices in a 10-Gigabit Ethernet system.

The XAUI core implements a single-speed full-duplex 10-Gbps Ethernet eXtended Attachment Unit Interface (XAUI) solution for the Xilinx Virtex™-II Pro and Virtex-4 families of FPGAs.

The Virtex™-II Pro and Virtex-4 FPGA families, in combination with the XAUI core, enable the design of XAUI-based interconnects whether chip-to-chip, over backplanes, or connected to 10 Gigabit optical modules.

Features

- Designed to 10-Gigabit Ethernet *IEEE 802.3ae-2002* specification
- Uses four RocketIO™ transceivers at 3.125 Gbps line rate to achieve 10-Gbps data rate
- Implements DTE XGXS, PHY XGXS, and 10GBASE-X PCS in a single netlist
- Uses Virtex-II Pro or Virtex-4 Digital Clock Management to implement optional XGMII interface clocking
- Uses Virtex-II Pro or Virtex-4 DDR I/O primitives for the optional XGMII interface
- Elastic buffering of inbound XGMII data (optional)
- Uses RocketIO transceivers for the XAUI interface
- 802.3ae-2002 Clause 45 MDIO interface (optional)
- 802.3ae-2002 Clause 48 State Machines (optional for Virtex-II Pro)
- Supports 10-Gigabit Fibre Channel (10-GFC) XAUI data rates and traffic
- Available under the SignOnce IP Site License program

LogiCORE Facts				
Core Specifics				
Supported Device Family	Virtex-II Pro -6, -7 (2VP4 or larger) Virtex-4[1] (4VFX)			
Resources Used[2]	**Slices**	**LUTs**	**FFs**	**Block RAMs**
	917	1327	700	0
Special Features	Delivered through CORE Generator™			
Provided with Core				
Documentation	Product Specification Getting Started Guide User Guide			
Design File Formats	NGC netlist			
Constraints File	UCF			
Verification	VHDL Test Bench Verilog Test Fixture			
Example design	VHDL and Verilog			
Additional Items	UniSim-based Simulation Models			
Design Tool Requirements				
Xilinx Implementation Tools	ISE™ 8.2i			
Simulation	Mentor ModelSim® v6.1e Cadence IUS			
Support				
Provided by Xilinx, Inc. www.xilinx.com/support/				

1. Virtex-4 FX solutions require the latest silicon stepping and are pending hardware validation.
2. Figures quoted are approximate for Virtex-II Pro default configuration. See "Device Utilization" on page 13 for details on device utilization by configuration.

Overview

XAUI is a four-lane, 3.125 Gbps-per-lane serial interface. Each lane is a differential pair, carrying current mode logic (CML) signalling and the data on each lane is 8B/10B encoded before transmission. Special code groups are used to allow each lane to synchronize at a word boundary and to de-skew all four lanes into alignment at the receiving end. The XAUI standard is fully specified in Clauses 47 and 48 of the 10-Gigabit Ethernet *IEEE 802.3ae-2002* specification.

The XAUI standard was initially developed as a means to extend the physical separation possible between MAC and PHY components in a 10-Gigabit Ethernet system distributed across a circuit board, and to reduce the number of interface signals in comparison with the XGMII (Ten Gigabit Ethernet Media Independent Interface).

Applications

Figure 1 shows the XAUI core being used to connect a Ten Gigabit Ethernet MAC to a 10-Gigabit XPAK optical module.

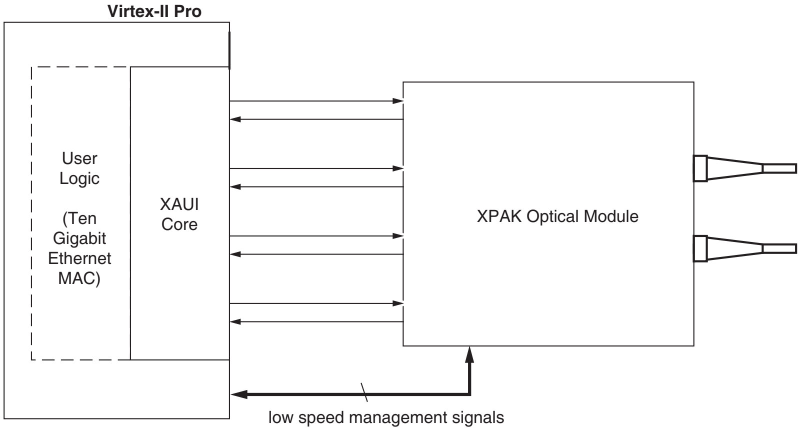

Figure 1: **Connecting XAUI to an Optical Module**

Since its publication, the applications of XAUI have extended beyond 10-Gigabit Ethernet to backplane and other general high-speed interconnect applications. Figure 2 shows a typical backplane application.

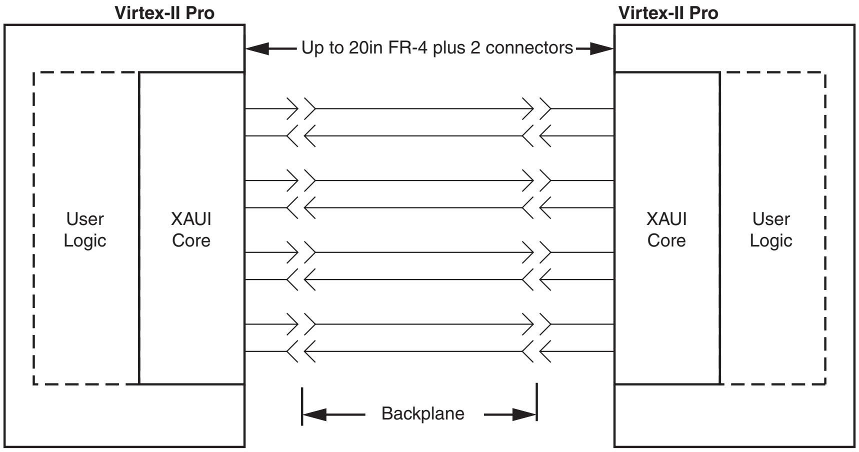

Figure 2: **Typical Backplane Application for XAUI**

Functional Description

Figure 3 shows a block diagram of the XAUI core implementation. The major functional blocks of the core include the following:

- **Client-side XGMII Interface**: If necessary, converts 32-bit DDR data into 64-bit SDR data and crosses clock domain for inbound XGMII data using an elastic buffer.
- **Transmit Idle Generation Logic**: Creates the code groups to allow synchronization and alignment at the receiver.
- **Synchronization State Machine** (one per lane): Identifies byte boundaries in incoming serial data.
- **De-skew State Machine**: De-skews the 4 received lanes into alignment.
- **Optional MDIO Interface**: A 2-wire low-speed serial interface used to manage the core.
- **Four RocketIO Transceivers (embedded in the Virtex-II Pro or Virtex-4 device)**: Provides the high-speed transceivers as well as 8B/10B encode and decode, and elastic buffering in the receive data path.

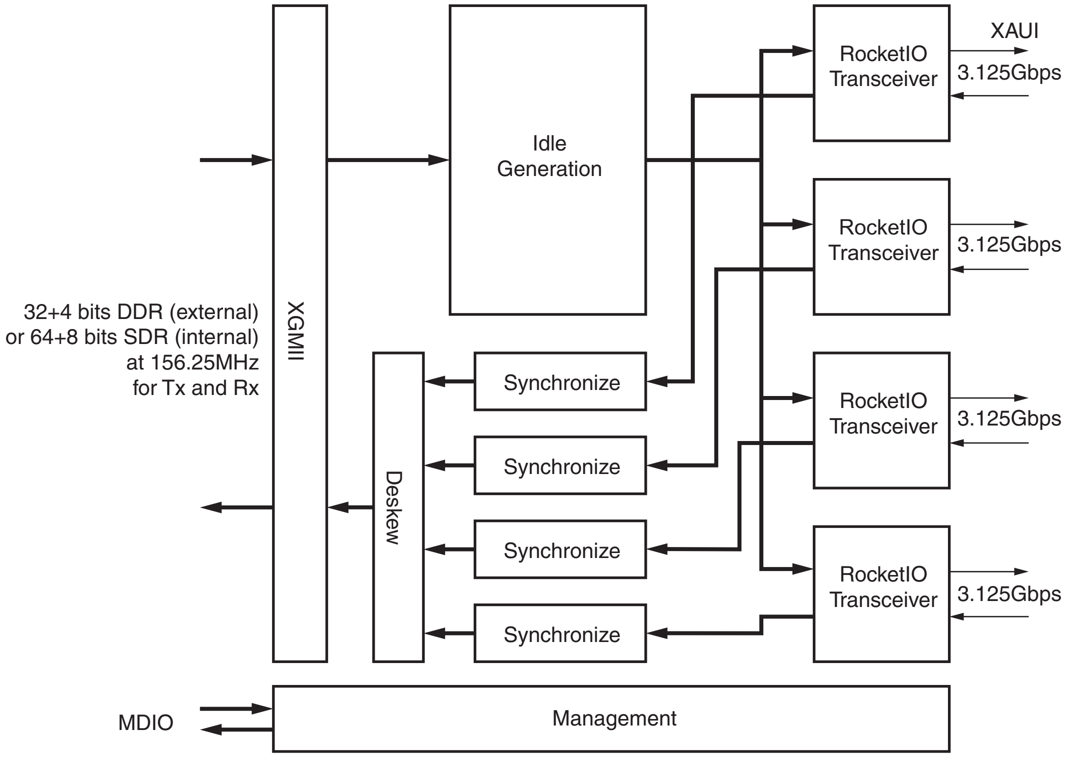

Figure 3: **Implementation of XAUI Core**

The core is implemented with the RocketIO instantiations in source code rather than in the netlist. This gives the user more flexibility in their particular application to use additional RocketIO features and resolve placement issues.

Core Interfaces

XAUI Interface

The XAUI interface consists of 4 differential transmit and receive pairs plus 4 low-speed control inputs to indicate the status of an attached optical module. The differential pairs are generated by the RocketIO. The control input signals are connected directly to the core. These signals are described in Table 1.

Table 1: **XAUI Interface Ports**

Signal Name	Direction	Description
SIGNAL_DETECT[3:0]	IN	Signals from a 10GBASE-LX4 optical module indicating the optical receivers are illuminated by a signal. If unused, tie this bus to “1111.”

MGT Interface

The MGT interface in the internal connection between the XAUI core netlist and the serial transceivers (MGT or GT11) which provide the XAUI interface. For detailed descriptions of the functions of these signals, see the Rocket IO (MGT or GT11) user guides. Table 2 describes these signals.

Table 2: **MGT Interface Ports**

Signal Name	Direction	Description
MGT_TXDATA	OUT	Data to MGTs
MGT_TXCHARISK	OUT	Control to MGTs
MGT_RXDATA	IN	Data from MGTs
MGT_RXCHARISK	IN	Control from MGTs
MGT_CODEVALID	IN	Code error signals from each MGT
MGT_CODECOMMA	IN	Code comma signals from each MGT
MGT_ENABLE_ALIGN	OUT	Enable comma align signals to each MGT
MGT_ENCHANSYNC	OUT	Enable channel sync signal to the MGTs
MGT_SYNCOK	IN	Sync OK signals from each MGT
MGT_RXLOCK	IN	Receive lock signals from each MGT
MGT_LOOPBACK	OUT	Loopback enable signal to each MGT
MGT_POWERDOWN	IN	Power down signal to each MGT

Client-side Interface

The client-side interface is a 72-bit (64 data bits and 8 control bits) interface running at 156.25 MHz based on the XGMII standard. It is designed to be easily connected to either user logic within the FPGA or, by using SelectIO DDR registers in the design top-level, to provide an external 32-bit 312 Mbps DDR XGMII defined in Clause 46 of *IEEE 802.3ae-2002*.

Table 3: **Client-side Interface Ports**

Name	Direction	Description
XGMII_TXD[63:0]	IN	Transmit data, 8 bytes wide
XGMII_TXC[7:0]	IN	Transmit control bits, one bit per transmit data byte.
XGMII_RXD[63:0]	OUT	Received data, 8 bytes wide
XGMII_RXC[7:0]	OUT	Receive control bits, one bit per received data byte.

Figure 4 illustrates transmitting a frame through the client-side interface.

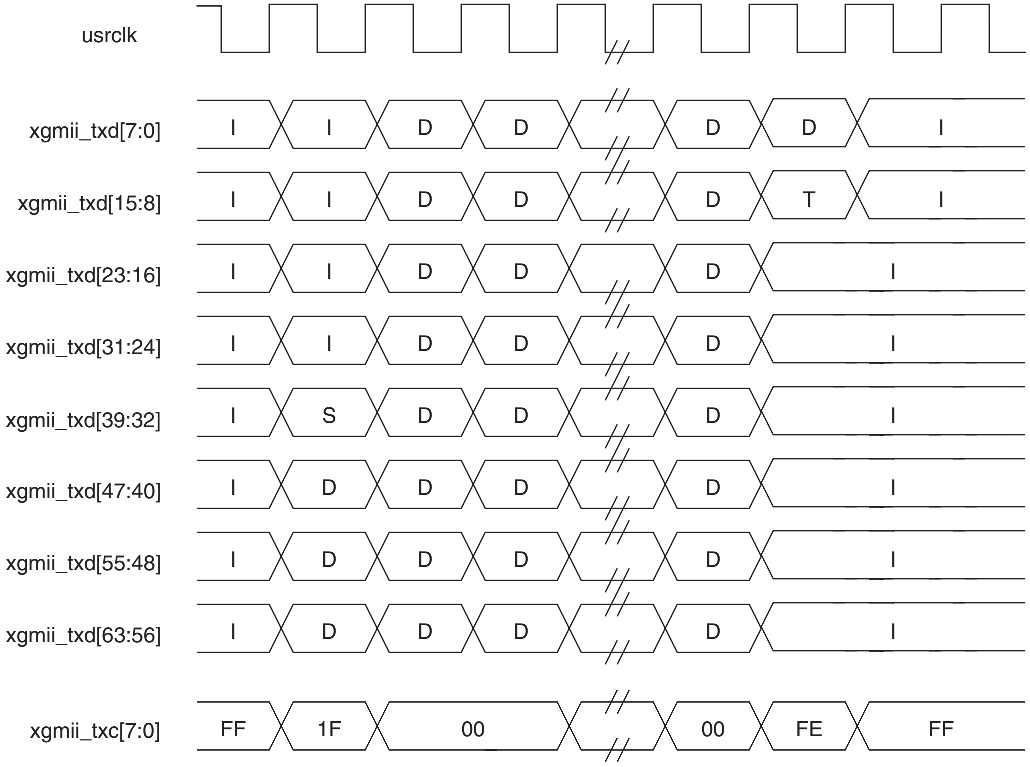

Figure 4: **Transmitting a Frame Through the Client-side Interface**

Figure 5 illustrates receiving a frame through the client-side interface.

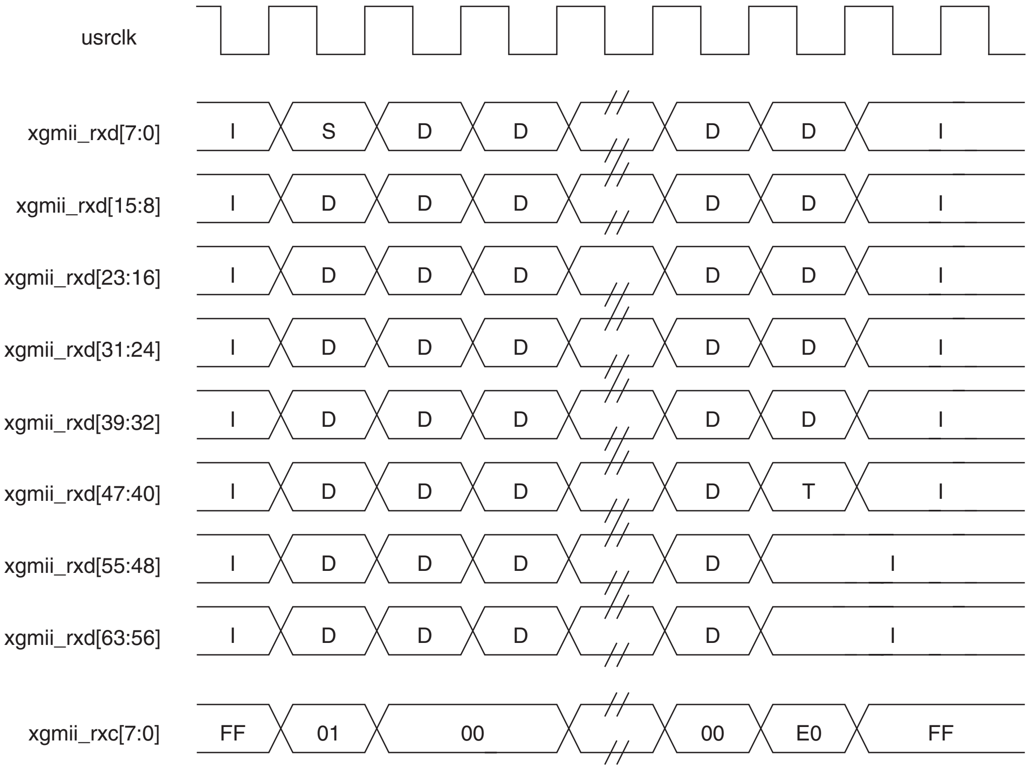

Figure 5: **Receiving a Frame Through the Client-side Interface**

Management Interface (MDIO)

The MDIO interface is a simple low-speed 2-wire interface for management of the XAUI core, consisting of a clock signal and a bi-directional data signal. It is defined in Clause 45 of *IEEE 802.3ae-2002* standard.

In the XAUI core, the MDIO interface is an optional block. If implemented, the bi-directional data signal MDIO is implemented as three unidirectional signals. These can be used to drive a tri-state buffer either in the FPGA IOB or in a separate device.

There are additional signals that control the behavior of the core's MDIO interface, specifically to set its position in the MDIO memory map.

Table 4: **MDIO Management Interface Ports**

Signal Name	Direction	Description
MDC	IN	Management clock
MDIO_IN	IN	MDIO input
MDIO_OUT	OUT	MDIO output

Table 4: **MDIO Management Interface Ports *(Continued)***

Signal Name	Direction	Description
MDIO_TRI	OUT	MDIO tri-state. ’1’ disconnects the output driver from the MDIO bus.

TYPE_SEL[1:0]	IN	Type select. Determines which MDIO Register addresses the core responds to.
		type_sel = "00" or "01"- 10GBASE-X PCS
		type_sel = "10" - DTE XS
		type_sel = "11" - PHY XS
		See the *XAUI User Guide* for the MDIO Register addresses responded to in each case.

PRTAD[4:0]	IN	MDIO port address. When multiple MDIO-managed ports appear on the same bus, this address can be used to address each one individually.

Configuration and Status Signals

In addition to the pollable MDIO interface, the XAUI core continuously indicates its status on ports, as defined in Table 5.

Table 5: **Configuration and Status Vector Ports**

Signal Name	Direction	Description
ALIGN_STATUS	OUT	’1’ when the XAUI receiver is aligned across four lanes.
SYNC_STATUS[3:0]	OUT	Each pin is ’1’ when the respective XAUI lane receiver is synchronized to byte boundaries.
CONFIGURATION_ VECTOR[6:0]	IN	Configuration signals for the core. The bits are: Bit 0 - Loopback Bit 1 - Power down transceiver Bit 2 - Reset local fault status Bit 3 - Reset RX link status Bit 4 - Test Enable - ’1’ transmits test patterns on XAUI TX Bits 6:5 - Test pattern select For a more comprehensive description of these signals, please consult the *XAUI User Guide*. This port only exists on the core if the MDIO interface is omitted.
STATUS_VECTOR[7:0]	OUT	Status indicators for the core. The bits are: Bit 0 - TX local fault Bit 1 - RX local fault Bit 5:2 - Synchronization - identical to SYNC_STATUS[3:0] Bit 6 - Alignment - identical to ALIGN_STATUS Bit 7 - RX link status For a more comprehensive description of these signals, please consult the *XAUI User Guide*. This port only exists on the core if the MDIO interface is omitted.

Clock and Reset

Table 6 describes the clock and reset ports present on the core.

Table 6: **Clock and Reset Ports**

Name	Direction	Description
TX_CLK	IN	This port is only present if the core has been generated including a transmit elastic buffer. This clock is the forwarded reference clock for the source-centred XGMII transmit data. It is used to clock data into the FPGA, then the data is crossed into the system clock domain by the elastic buffer.
RESET	IN	Synchronous reset for core. The reference clock must be running for the core to emerge from the reset state.
USRCLK	IN	FPGA fabric logic system clock.
MGT_TX_RST	IN	Connect this to the same signal used to drive the MGT TXRESET signal
MGT_RX_RST	IN	Connect this to the same signal used to drive the MGT RXRESET signal

MDIO Management Registers

The XAUI core, when generated with an MDIO interface, implements an MDIO Interface Register block. The core will respond to MDIO transactions as either a 10GBASE-X PCS, a DTE XS or a PHY XS depending on the setting of the type_sel port (see Table 4).

10GBASE-X PCS Registers

Table 7 shows the MDIO registers present when the XAUI core is configured as a 10GBASE-X PCS. For a more comprehensive description of the registers and their effect on core operation, see the *XAUI User Guide*.

Table 7: **10GBASE-X PCS/PMA MDIO Registers**

Register Address	Register Name
1.0	PMA/PMD Control 1
1.1	PMA/PMD Status 1
1.2,1.3	PMA/PMD Device Identifier
1.4	PMA/PMD Speed Ability
1.5, 1.6	PMA/PMD Devices in Package
1.7	10G PMA/PMD Control 2
1.8	10G PMA/PMD Status 2
1.9	Reserved
1.10	10G PMD Receive Signal OK
1.11 TO 1.13	Reserved
1.14, 1.15	PMA/PMD Package Identifier
1.16 to 1.65 535	Reserved
3.0	PCS Control 1
3.1	PCS Status 1
3.2, 3.3	PCS Device Identifier
3.4	PCS Speed Ability
3.5, 3.6	PCS Devices in Package
3.7	10G PCS Control 2
3.8	10G PCS Status 2
3.9 to 3.13	Reserved
3.14, 3.15	PCS Package Identifier
3.16 to 3.23	Reserved
3.24	10GBASE-X PCS Status
3.25	10GBASE-X Test Control
3.26 to 3.65 535	Reserved

DTE XS Registers

Table 8 shows the MDIO registers present when the XAUI core is configured as a DTE XS. For a more comprehensive description of the registers and their effect on core operation, see the *XAUI User Guide*.

Table 8: **DTE XS MDIO Registers**

Register Address	Register Name
5.0	DTE XS Control 1
5.1	DTE XS Status 1
5.2, 5.3	DTE XS Device Identifier
5.4	DTE XS Speed Ability
5.5, 5.6	DTE XS Devices in Package
5.7	Reserved
5.8	DTE XS Status 2
5.9 to 5.13	Reserved
5.14, 5.15	DTE XS Package Identifier
5.16 to 5.23	Reserved
5.24	10G DTE XGXS Lane Status
5.25	10G DTE XGXS Test Control

PHY XS Registers

Table 9 shows the MDIO registers present when the XAUI core is configured as a PHY XS. For a more comprehensive description of the registers and their effect on core operation, see the *XAUI User Guide*.

Table 9: **PHY XS MDIO Registers**

Register Address	Register Name
4.0	PHY XS Control 1
4.1	PHY XS Status 1
4.2, 4.3	PHY XS Device Identifier
4.4	PHY XS Speed Ability
4.5, 4.6	PHY XS Devices in Package
4.7	Reserved
4.8	PHY XS Status 2
4.9 to 4.13	Reserved
4.14, 4.15	PHY XS Package Identifier
4.16 to 4.23	Reserved
4.24	10G PHY XGXS Lane Status
4.25	10G PHY XGXS Test Control

10-Gigabit Fibre Channel Support

The 10-Gigabit Fibre Channel (10GFC) specification describes a XAUI interface similar to the 10-Gigabit Ethernet XAUI but operating at 2% higher line and data rates; this equates to a line rate on each RocketIO lane of 3.1875 Gbps. The XAUI core has been verified with 10-Gigabit Fibre Channel traffic at this higher rate on Virtex-II Pro device.

Virtex-4 Implementation

The following options are not available for the Virtex-4 architecture:

- Transmit-only
- Receive-only
- Non 802.3ae-2002 State Machines

Verification

The XAUI core has been verified using both simulation and hardware testing in Virtex-II Pro and by simulation in Virtex-4.

Simulation

A highly parameterizable transaction-based simulation test suite has been used to verify the core. Tests include

- Register access over MDIO
- Loss and re-gain of synchronization
- Loss and re-gain of alignment
- Frame transmission
- Frame reception
- Clock compensation
- Recovery from error conditions

Hardware Verification

The core has been used in a number of hardware test platforms within Xilinx. In particular, the core has been used in a Virtex-II Pro test platform design with the Xilinx 10-Gigabit Ethernet MAC. This design comprises the MAC, XAUI, a *ping* loopback FIFO and a test pattern generator all under embedded PowerPC™ processor control. This design has been used for conformance and interoperability testing at the University of New Hampshire Interoperability Lab.

Device Utilization

Virtex-II Pro

Tables 10, 11, and 12 provide approximate slice counts for the various core options on Virtex-II Pro FPGAs. Table 10 defines the resource usage for full duplex implementations, Table 11 defines the resource usage for transmit-only cores, and Table 12 shows the resource usage for receive-only implementations.

Table 10: **Device Utilization - Virtex-II Pro, Full Duplex Core**

Parameter Values				Device Resources		
External XGMII	802.3ae State Machines	TX Elastic Buffer	MDIO Interface	Slices	LUTs	FFs
No	No	N/A	No	569	743	556
No	No	N/A	Yes	712	952	646
No	Yes	N/A	No	793	1120	636
No	Yes	N/A	Yes	916	1327	700
Yes	No	No	No	698	752	736
Yes	No	No	Yes	837	961	827
Yes	No	Yes	No	1051	1034	1161
Yes	No	Yes	Yes	1199	1257	1257
Yes	Yes	No	No	921	1128	816
Yes	Yes	No	Yes	1043	1341	881
Yes	Yes	Yes	No	1276	1412	1240
Yes	Yes	Yes	Yes	1406	1615	1322

Table 11: **Device Utilization - Virtex-II Pro, Transmit-only Core**

Parameter Values				Device Resources		
External XGMII	802.3ae State Machines	TX Elastic Buffer	MDIO Interface	Slices	LUTs	FFs
No	N/A	N/A	No	308	461	333
Yes	N/A	No	No	401	470	442
Yes	N/A	Yes	No	760	762	865

Table 12: **Device Utilization - Virtex-II Pro, Receive-only Core**

Parameter Values				Device Resources		
External XGMII	802.3ae State Machines	TX Elastic Buffer	MDIO Interface	Slices	LUTs	FFs
No	No	N/A	No	289	320	268
No	Yes	N/A	No	505	691	342
Yes	No	N/A	No	324	328	342
Yes	Yes	N/A	No	557	709	424

Virtex-4

Table 13 provides approximate slice counts for the various core options on Virtex-4 FPGAs.

Table 13: **Device Utilization - Virtex-4**

Parameter Values			Device Resources		
External XGMII	TX Elastic Buffer	MDIO Interface	Slices	LUTs	FFs
No	N/A	No	835	1510	643
No	N/A	Yes	947	1451	715
Yes	No	No	835	1510	643
Yes	No	Yes	948	1451	715
Yes	Yes	No	1204	1595	1065
Yes	Yes	Yes	1316	1780	1138

References

[1] IEEE Std. 802.3-2002, Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications.

[2] IEEE Std. 802.3ae-2002, Media Access Control (MAC) Parameters, Physical Layers, and Management Parameters for 10-Gbps Operation.

Support

Please visit http://www.xilinx.com/support for technical support. Xilinx provides technical support for this LogiCORE product when used as described in product documentation.

Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not listed in the documentation or if customized beyond that allowed in the product documentation, or if any changes are made in sections of the design marked *DO NOT MODIFY*.

Ordering Information

The XAUI core is provided under the SignOnce IP Site License and can be generated using CORE Generator system v8.2i and higher. The CORE Generator system is shipped with Xilinx ISE Foundation Series Development software.

A simulation evaluation license for the core is shipped with the CORE Generator system. To access the full functionality of the core, including FPGA bitstream generation, a full license must be obtained from Xilinx. For more information, please visit the XAUI product page at
http://www.xilinx.com/systemio/xaui/index.htm.

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

Revision History

Date	Version	Revision
9/24/04	1.0	Initial Xilinx release in new data sheet format.
4/28/05	1.1	Document updated to support XAUI core v6.0 and Xilinx software v7.1i.
1/18/06	1.2	Updated dates, version number, and ISE tools 8.1i.
7/13/06	1.3	Updated core version to 6.2; Xilinx tools to 8.2i.