

XAUI IP Core - Lattice Radiant Software

User Guide

FPGA-IPUG-02160-1.0

October 2021



Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults and associated risk the responsibility entirely of the Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.



Contents

Acronyms in This Document	5
1. Introduction	6
1.1. Quick Facts	6
1.2. Features	6
1.3. Conventions	7
1.3.1. Nomenclature	7
1.3.2. Signal Names	7
1.3.3. Attribute Names	7
2. Functional Descriptions	8
2.1. Overview	8
2.2. Signal Description	9
2.3. Attribute Summary	11
2.4. Register Description	11
2.4.1. MDIO Interface Registers	11
2.4.2. APB Interface Registers	13
2.5. XGMII Interface and Slip Buffers	13
2.6. XAUI-to-XGMII Translation (Receive Interface)	15
2.6.1. MPCS Soft IP (Receive Mode)	15
2.6.2. XAUI Rx Core	15
2.6.3. Rx Slip Buffer	16
2.6.4. Timing of Data Frame Transfer to Client-side Interface	16
2.7. XGMII to XAUI Translation (Transmit Interface)	18
2.7.1. Tx Slip Buffer	
2.7.2. XAUI Tx Core	18
2.7.3. MPCS Soft IP (Transmit Mode)	19
2.7.4. Timing of Data Frame Transfer from Client-side Interface	19
2.8. MDIO Interface	20
3. IP Generation and Evaluation	22
3.1. Licensing the IP	22
3.2. Generation and Synthesis	22
3.3. Running Functional Simulation	
3.4. Hardware Evaluation	
4. Ordering Part Number	
Appendix A. Resource Utilization	
References	29
Technical Support Assistance	30
Revision History	31



Figures

Figure 2.1. XAUI and XGXS Locations in 10 GbE Protocol Stack	8
Figure 2.2. Functional Block Diagram	9
Figure 2.3. XGMII Timing Diagram	13
Figure 2.4. 64-bit Client-side Interface after de-multiplexing the 32-bit XGMII data	14
Figure 2.5. XAUI IP Receive Path	15
Figure 2.6. Timing of Data Frame Transfer to Client-side Interface	17
Figure 2.7. Timing of Data Frame Transfer with Error to Client-side Interface	
Figure 2.8. XAUI IP Transmit Path	18
Figure 2.9. Timing of Data Frame Transmission from Client-side Interface	19
Figure 2.10. Timing of Data Frame Transmission with Error from Client-side Interface	
Figure 2.11. A Typical MDIO-Managed System	20
Figure 2.12. Field of MDIO Protocol	
Figure 2.13. Indirect Address Example	21
Figure 3.1. Module/IP Block Wizard	22
Figure 3.2. Configure User Interface of selected XAUI IP Core	
Figure 3.3. Check Generating Result	23
Figure 3.4. Simulation Wizard	
Figure 3.5. Adding and Reordering Source	
Figure 3.6. Simulation Waveform	25
Tables	
Table 1.1. XAUI IP Quick Facts	
Table 2.1. XAUI IP Core Signal Description	
Table 2.2. Attributes Table	
Table 2.3. Attributes Description	
Table 2.4. PHY XGXS MDIO Registers	
Table 2.5. Summary of XAUI IP Core Registers Accessible through MDIO	
Table 2.6. XGMII Control Encoding	
Table 2.7. XAUI 8b10b Code Points	
Table 2.8. XAUI 8b10b to XGMII Code Mapping	
Table 2.9. XGMII to XAUI Code Mapping	
Table 3.1. Generated File List	
Table A.1. Resource Utilization	28



Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
AMBA	Advanced Microcontroller Bus Architecture
APB	Advanced Peripheral Bus
LMMI	Lattice Memory Mapped Interface
PCS	Physical Coding Sublayer
RTL	Register Transfer Language
XGMII	10-Gigabit Media Independent Interface
XGXS	10Gb Ethernet Extended Sublayer
XAUI	10Gb Attachment Unit Interface



1. Introduction

The Lattice Semiconductor 10Gb Ethernet Attachment Unit Interface (XAUI) provides a solution for bridging between XAUI and 10-Gigabit Media Independent Interface (XGMII) devices. The IP core implements the 10Gb Ethernet Extended Sublayer (XGXS) capabilities in soft logic that together with PCS and SerDes functions implemented in the FGPA provides a complete XAUI-to-XGMII solution.

This document describes the use of the XAUI Interface IP and Lattice FPGA technology for Ethernet interface applications. The design, which can be applied in multiple configurations, is implemented in Verilog HDL.

1.1. Quick Facts

Table 1.1 presents a summary of the XAUI IP.

Table 1.1. XAUI IP Quick Facts

IP Requirements	Supported FPGA Families	CertusPro™-NX		
	Targeted Devices	LFCPNX-100		
Resource Utilization	Supported User Interface	10Gb PHY serial to/from XGMII		
	Resources	See Table A.1.		
Design Tool Support	Lattice Implementation	IP Core v1.0.x – Lattice Radiant™ software 3.0 or later		
	Cunthosis	Lattice Synthesis Engine		
	Synthesis	Synopsys® Synplify Pro® for Lattice		
	Simulation	For a list of supported simulators, see the Lattice Radiant software user guide.		

1.2. Features

The key features of the XAUI IP include:

- XAUI compliant functionality supported by embedded PCS implemented in the CertusPro-NX device, including four channels of 3.125 Gbps serializer/deserializer with 8b10b encoding/decoding.
- Complete 10Gb Ethernet Extended Sublayer (XGXS) solution
- Implements XGXS functionality conforming to IEEE 802.3-2005, including:
 - 10 GbE Media Independent Interface (XGMII).
 - Optional slip buffers for clock domain transfer to/from the XGMII interface
 - Complete translation between XGMII and XAUI PCS layers, including 8b10b encoding and decoding of Idle,
 Start, Terminate, Error and Sequence code groups and sequences, and randomized Idle generation in the XAUI transmit direction
 - XAUI compliant lane-by-lane synchronization.
 - Lane de-skew functionality
 - Optional standard compliant MDIO/MDC interface
- Supports AMBA 3 APB Protocol version 1.0 for register access of the soft IP
 - Supports PREADY, a ready signal to indicate completion of an APB transfer
 - Does not support PSLVERR (error signal indicating failure of a transfer)



1.3. Conventions

1.3.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.3.2. Signal Names

Signal names that end with:

- _n are active low (asserted when value is logic 0)
- _i are input signals
- _o are output signals

1.3.3. Attribute Names

Attribute names in this document are formatted in title case and italicized (Attribute Name).



2. Functional Descriptions

2.1. Overview

XAUI is a high-speed interconnect that offers reduced pin count. Each XAUI comprises four self-timed 8b10b encoded serial lanes each operating at 3.125 Gbps and thus is capable of transferring data at an aggregate rate of 10 Gbps.

XGMII is a 156.25 MHz Double Data Rate (DDR), parallel, short-reach interconnect interface. It supports interfacing to 10 Gbps Ethernet Media Access Control (MAC) and PHY devices.

The locations of XAUI and XGXS in the 10GbE protocol stack is shown in Figure 2.1.

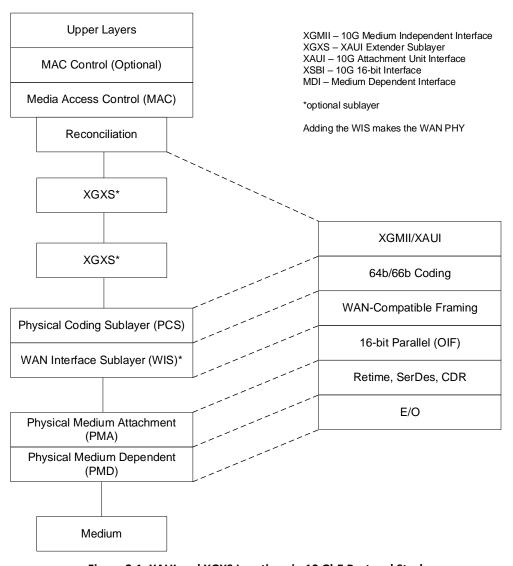


Figure 2.1. XAUI and XGXS Locations in 10 GbE Protocol Stack

The functional block diagram of XAUI IP Core is shown in Figure 2.2. The dashed lines in the figure are optional components/signals, which means they may not be available in the IP when disabled in the attribute.

8



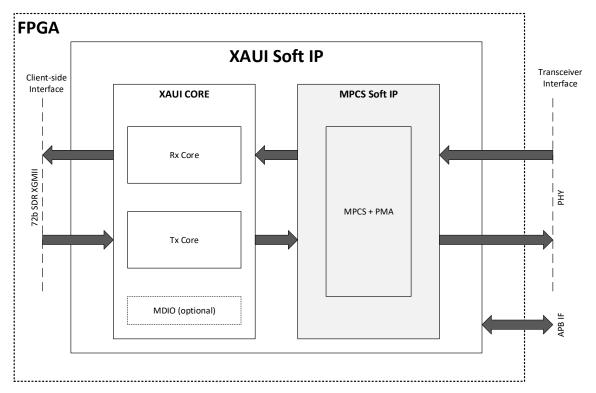


Figure 2.2. Functional Block Diagram

XAUI IP is consists of MPCS soft IP that instantiates MPCS hard IP configured as XAUI, XAUI IP Core with optional MDIO interface that is reused from LatticeECP3 and LatticeECP5, and APB interface that provides direct access to MPCS hard IP registers.

2.2. Signal Description

Table 2.1. XAUI IP Core Signal Description

Port Name	1/0	Width	Description	
System				
clk_xgmii_tx_o	out	1	Clock from XAUI IP core to the XGMII Tx.	
			This is just equal to clk_156_tx_i.	
clk_xgmii_rx_o	out	1	Clock from XAUI IP core to the XGMII Rx.	
			This is just equal to clk_156_rx_i.	
clk_156_rx_i	in	1	156.25 MHz XGMII RX clock. This is also used to clock the data coming from the MPCS.	
clk_156_tx_i	in	1	156.25 MHz XGMII TX clock.	
reset_n_i	in	1	Active-low asynchronous reset	
ready_o	out	1	Indicates that MPCS is ready	
Transceiver Interfac	е			
refclkp_i	in	1	Differential transceiver clock pair	
refclkn_i	in	1		
hdinp0_i	in	1	Differential receive input pair lane 0	
hdinn0_i	in	1		
hdinp1_i	in	1	Differential receive input pair lane 1	
hdinn1_i	in	1		
hdinp2_i	in	1	Differential receive input pair lane 2	
hdinn2_i	in	1		



Port Name	1/0	Width	Description	
hdinp3 i	in	1	Differential receive input pair lane 3	
hdinn3 i	in	1		
hdoutp0_o	out	1	Differential transmit output pair lane 0	
hdoutn0_o	out	1	_	
hdoutp1 o	out	1	Differential transmit output pair lane 1	
hdoutn1_o	out	1		
hdoutp2_o	out	1	Differential transmit output pair lane 2	
hdoutn2_o	out	1		
hdoutp3_o	out	1	Differential transmit output pair lane 3	
hdoutn3_o	out	1		
Client-side Interface		_		
xgmii_txd_i	in	64	Transmit data, 8 bytes wide; Data is from XGMII TX going to XAUI	
xgmii_txc_i	in	8	Transmit control bits, 1-bit per transmit data byte	
xgmii_rxd_o	out	64	Receive data, 8 bytes wide; Data goes to XGMII RX from XAUI	
xgmii rxc o	out	8	Receive control bits, 1-bit per receive data byte	
	L	<u> </u>	and/or Enable Tx Slip Buffer is/are checked)	
tx_fifo_empty_o	out	1	Tx FIFO Slip Buffer FIFO empty flag	
tx_fifo_full_o	out	1	Tx FIFO Slip Buffer FIFO full flag	
rx_fifo_empty_o	out	1	Rx FIFO Slip Buffer FIFO empty flag	
rx_fifo_full_o	out	1	Rx FIFO Slip Buffer FIFO full flag	
MDIO Interface ² (wher		<u> </u>		
mdin i	in	1	MDIO serial input data	
mdc i	in	1	MDIO input clock	
mdout o	out	1	MDIO serial output data	
mdtri o	out	1		
APB Interface	Out		Tristate control for MDIO port	
	lin	1	ADD clock. This should also be running at 156.35 MHz	
apb_pclk_i	in	1	APB clock. This should also be running at 156.25 MHz.	
apb_psel_i	in		Select signal. Indicates that the slave device is selected and a data transfer is required.	
apb_paddr_i	in	32	Address signal.	
apb_pwdata_i	in	32	Write data signal.	
apb_pwrite_i	in	1	Direction signal. Write = 1, Read = 0	
apb_penable_i	in	1	Enable signal. Indicates the second and subsequent cycles of an APB transfer.	
apb_pready_o	out	1	Ready signal. Indicates transfer completion. Slave uses this signal to extend an APB transfer.	
apb_prdata_o	out	32	Read data signal.	
apb_pslverr_o	out	1	APB error signal. Not supported and just tied to 0.	
Others ³		•		
mca_resync_i	in	1	Multi-channel alignment resynchronization request. This is in the	
_			clk_156_rx_i clock domain.	
			When not enabled in the gui, this is default to 0.	
mca_auto_resync_i	in	1	Multi-channel alignment auto resynchronization request. This is in the	
			clk_156_rx_i clock domain.	
			When not enabled in the gui, this is default to 1.	
mca_sync_status_o	out	1	XAUI multi-channel alignment status.	
tad :4	1	-	1 = All channels are aligned; 0 = XAUI channels are not aligned	
prtad_i ⁴	In	5	MDIO Port Address. (when MDIO Interface == enable)	
	1		When not enabled in the gui, this is default to 5'b00000.	



Notes:

- 1. Available only when Enable Rx Slip Buffer and/or Enable Tx Slip Buffer is/are checked.
- 2. Available only when Enable MDIO Interface is checked.
- 3. Available only when Enable Control signals is checked, except for prtad_i port.
- 4. Available only when both Enable Control signals and Enable MDIO Interface are checked.

2.3. Attribute Summary

The configurable attributes of the XAUI IP Core are shown in Table 2.2 and are described in Table 2.3. The attributes can be configured through the IP Catalog's Module/IP wizard of the Lattice Radiant software.

Table 2.2. Attributes Table

Attribute	Selectable Values	Default	Dependency on Other Attributes
Enable Tx Slip Buffer	Checked, Unchecked	Checked	_
Enable Rx Slip Buffer	Checked, Unchecked	Checked	_
Enable MDIO Interface	Checked, Unchecked	Unchecked	_
Enable Control signals	Checked, Unchecked	Unchecked	_

Table 2.3. Attributes Description

Attribute	Description
Enable Tx Slip Buffer	This option allows you to include a slip buffer in the XAUI transmit direction for clock tolerance compensation.
Enable Rx Slip Buffer	This option allows you to include a slip buffer in the XAUI receive direction for clock tolerance compensation.
Enable MDIO Interface	This option allows you to include a Management Data Input/Output Interface providing access to XAUI IP core internal registers.
Enable Control signals	This option allows you to control the following control signals. mca_resync_i mca_auto_resync_i mca_sync_status_o prtad_i (this is only available if <i>Enable MDIO Interface</i> is also checked)

2.4. Register Description

2.4.1. MDIO Interface Registers

When MDIO interface is enabled, registers in Table 2.4 can be accessed.

The XAUI Soft IP is configured as a PHY XGXS, thus, it occupies MDIO Device Address 4 according to IEEE 802.3-2005.

Table 2.4. PHY XGXS MDIO Registers

Register Address	Register Name
4.0	PHY XGXS control 1
4.1	PHY XGXS Status 1
4.2, 4.3	PHY XGXS Identifier
4.4	Reserved
4.5	PHY XGXS Status 2
4.6–4.23	Reserved
4.24	10G PHY XGXS Lane Status
4.25 – 4.32767	Reserved
4.32768 – 4.65535	Vendor Specific



Table 2.5. Summary of XAUI IP Core Registers Accessible through MDIO

Bit(s)	Name	Access	Default Value	Description
Control 1 Regi	ister			
4.0.15	Reset	R/W S/C	0	1 = PHY XS reset, 0 = Normal
				operation
4.0.14	Loopback (not supported)	R	0	The XAUI core does not provide loopback capability.
4.0.13	Speed Selection	R	0	Value always 0
4.0.12	Reserved	R	0	Value always 0
4.0.11	Low Power	R/W	0	1= Low Power Mode
				0= Normal operation
4.0.[10:7]	Reserved	R	0	Value always 0
4.0.6	Speed Selection	R	0	Value always 0
4.0.[5:2]	Speed Selection	R	0	Value always 0
4.0.[1:0]	Reserved	R	0	Value always 0
Status Registe	er 1			
4.1.[15:8]	Reserved	R	0	Value always 0
4.1.7	Fault (not supported)	R	0	0 =No Fault Condition
4.1.[6:3]	Reserved	R	0	Value always 0
4.1.2	PHY XS TX link status (not supported)	R	0	The Link status is available in the PCS core register
4.1.1	Low Power Ability	R	1	1=Low Power Mode support
4.1.0	Reserved	R	0	Value always 0
XGXS Identifie	er Registers			
4.2.[15:0]	PHY XS Identifier	R	0	MSB = 0x0000
4.3.[15:0]	PHY XS Identifier	R	4	LSB = 0x0004
XGXS Reserve	d Register			
4.4.[15:1]	Reserved	R	0	Value always 0
4.4.0	10G Capable	R	1	Value always 1
Status 1 Regis	ter			
4.5.[15:6]	Reserved	R	0	Value always 0
4.5.5	DTE XS Present	R	0	Value always 0
4.5.4	PHY XS Present	R	1	Value always 1
4.5.3	PCS Present	R	1	Value always 1
4.5.2	WIS Present	R	0	Value always 0
4.5.1	PMD/PMA Present	R	1	Value always 1
4.5.0	Clause 22 registers present	R	0	Value always 0
XGXS Reserve	d Register			
4.6.15	Vendor specific device present	R	0	Value always 0
4.6.[14:0]	Reserved	R	0	Value always 0
4.8.[15:14]	Device present	R	2	2=Device responding to this address
4.8.[13:12]	Reserved	R	0	Value always 0
4.8.11	Transmit Fault (not supported)	R	0	0=No fault of TX path
4.8.10	Receive Fault (not supported)	R	0	0=No fault of RX path
4.8.[9:0]	Reserved	R	0	Value always 0
4.14.[15:0]	Package Identifier	R	0	Value always 0

13



Bit(s)	Name	Access	Default Value	Description
4.15.[15:0]	Package Identifier	R	4	Value always 4
4.24.[15:13]	Reserved	R	0	Value always 0
4.24.12	PHY XGXS Lane Alignment (not supported)	R	0	TX alignment status
4.24.11	Pattern Testing Ability	R	0	0=Not able to generate pattern
4.24.10	PHY XGXS has loop back capability	R	0	0=Has no loop back capability
4.24.[9:4]	Reserved	R	0	Value always 0
4.24.3	Lane 3 Sync (not supported)	R	0	Not supported
4.24.2	Lane 2 Sync (not supported)	R	0	Not supported
4.24.1	Lane 1 Sync (not supported)	R	0	Not supported
4.24.0	Lane 0 Sync (not supported)	R	0	Not supported
4.25.[15:3]	Reserved	R	0	Value always 0
4.25.2	Receive test pattern enable	R	0	0=Receive test pattern not enabled
4.25.[1:0]	Test pattern select	R	0	Value always 00

2.4.2. APB Interface Registers

The MPCS Hard IP registers can be accessed directly through APB interface. Refer to Multi-Protocol PCS Module (FPGA-IPUG-02118) for the MPCS hard IP address mapping.

2.5. XGMII Interface and Slip Buffers

The 10-Gigabit Media Independent Interface (XGMII) supported by XAUI IP Core solution conforms to Clause 46 of IEEE 802.3-2005. The XGMII is composed of independent transmit and receive paths. Each direction uses 32-bit data signals, four (4) control signals and a clock. The 32-bit data signals in each direction are organized into four lanes of 8-bit signals each. Each lane is associated with a control signal. The XGMII supports Double Data Rate (DDR) transmission (that is the data and control input signals are sampled on both the rising and falling edge of the corresponding clock) as shown in Figure 2.3.

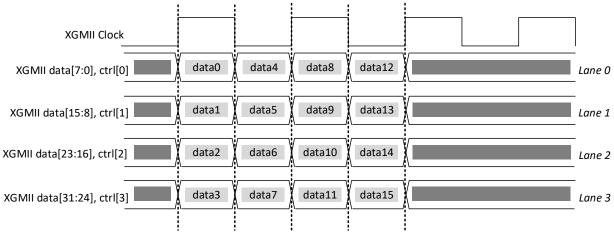


Figure 2.3. XGMII Timing Diagram

The client-side interface of the XAUI soft IP is a 64-bit single-data rate (SDR) and is based on the 32-bit XGMII-like interface. Before the XGMII data goes to the soft IP, the 32-bit wide DDR data should be de-multiplexed into 64-bits wide on a single clock edge. This de-multiplexing is done by extending the bus upwards so that there are now 8-lanes of data as shown in Figure 2.4.



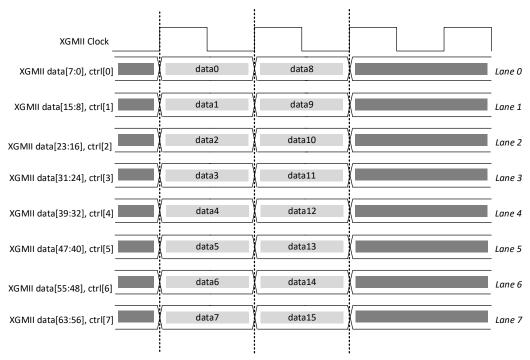


Figure 2.4. 64-bit Client-side Interface after de-multiplexing the 32-bit XGMII data

The control signal for each lane is de-asserted when a data octet is being sent on the corresponding lane and is asserted when a control character is being sent. Supported control octet encodings are shown in Table 2.6. All data and control signals are passed directly to/from the 8b10b encoding/decoding blocks.

Table 2.6. XGMII Control Encoding

Control Signal	Data	Description
0	0x00 – 0xFF	Normal data transmission (D)
1	0x00 - 0x06	Reserved
1	0x07	Idle (I)
1	0x08 - 0x9B	Reserved
1	0x9C	Sequence (only valid on lane 0)
1	0x9D – 0xFA	Reserved
1	0xFB	Start (S) – only valid on lane 0
1	0xFC	Reserved
1	0xFD	Terminate (T)
1	0xFE	Error (E)
1	0xFF	Reserved

The XAUI soft IP incorporates optional slip buffers that accommodate small differences between XGMII and XAUI timing by inserting or deleting idle characters. The slip buffer is implemented as a 256×72 FIFO. There are four flags from the FIFO: full, empty, partially full, and partially empty. The partially empty full is used as the watermark to start reading from the FIFO. If the difference between write and read pointers falls below the partially empty watermark and the entire packet has been transmitted, idle characters are inserted until the partially full watermark is reached. No idle is inserted during data transmission.



2.6. XAUI-to-XGMII Translation (Receive Interface)

The block diagram for XAUI Soft IP receive path is shown in Figure 2.5. The receive interface converts the incoming XAUI stream into XGMII-compatible signals. Rx Slip Buffer is optional and when disabled, the XGMII control and data directly comes from Rx Core.

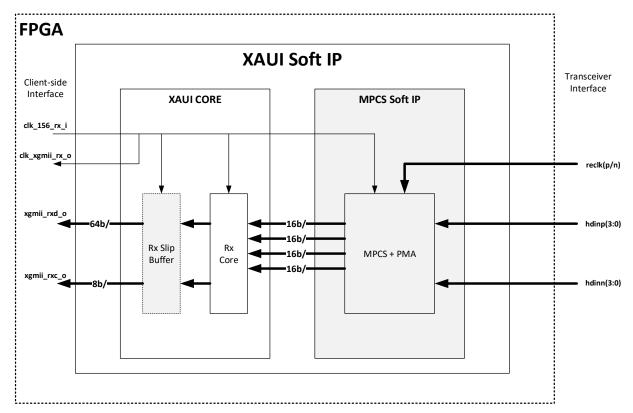


Figure 2.5. XAUI IP Receive Path

2.6.1. MPCS Soft IP (Receive Mode)

The MPCS soft IP is an IP that implements functionalities of PCS layer defined by SerDes protocols. The PMA and PCS settings in this IP is pre-configured for XAUI. This IP receives the serial data for each lane and performs mainly wordalignment, 8b/10b decoding, and lane-to-lane Rx deskew of the incoming data.

2.6.2. XAUI Rx Core

The four converted XAUI data lanes goes to the XAUI IP RX Core to de-skew, align and decode the XAUI code to its corresponding XGMII code. The XAUI IP RX Core consists of the multi-channel alignment block and RX decoder block.

The multi-channel alignment block consists of four 16-byte deep FIFOs to individually buffer each XAUI lane. The multi-channel alignment block searches for the presence of the alignment character /A/ in each XAUI lane, and begins storing the data in the FIFO when the /A/ character is detected in a lane. When the alignment character has been detected in all four XAUI lanes, the data is retrieved from each FIFO so that all of the alignment characters /A/ are aligned across all four XAUI lanes. Once synchronization is achieved, the block does not resynchronize until a resynchronization request is issued. The data from the RX multi-channel alignment is passed to the RX decoder.

The RX decoder block converts the XAUI code to the corresponding XGMII code. Table 2.7 shows the 8b10b code points. Table 2.8 shows the code mapping between the two domains in the receive direction. XAUI /A/, /R/, /K/ characters are translated into XGMII Idle (/I/) characters.



Table 2.7. XAUI 8b10b Code Points

Symbol	Name	Function	Code Group
/A/	Align	Lane alignment (XGMII idle)	K28.3
/K/	Sync	Code-Group Alignment (XGMII idle)	K28.5
/R/	Skip	Clock Tolerance Compensation (XGMII idle)	K28.0
/S/	Start	Start of Packet Delimiter (in lane 0 only)	K27.7
/T/	Terminate	End of Packet Delimiter	K29.7
/E/	Error	Error Propagation	K30.7
/Q/	Sequence	Link Status Message Indicator	K28.4
/d/	Data	Information Bytes	Dxx.x

Table 2.8. XAUI 8b10b to XGMII Code Mapping

8b10b Data from MPCS	XGMII	XGMII Control
K28.5 (0xBC)	0x07 (Idle)	1
K28.3 (0x7C)	0x07 (Idle)	1
K28.0 (0x1C)	0x07 (Idle)	1
K27.7 (0xFB)	0xFB (Start)	1
K29.7 (0xFD)	0xFD (Terminate)	1
K30.7 (0xFE)	OxFE (Error)	1
K28.4 (0x9C)	0x9C (Ordered Set)	1
Dxx.x	0x00-0xFF (Data)	0

2.6.3. Rx Slip Buffer

The decoded data is written to the Rx slip buffer (when *Enable Rx Slip Buffer* is checked). Slip buffers are required to compensate for differences in the write and read clocks derived from the XAUI and XGMII reference clocks, respectively.

2.6.4. Timing of Data Frame Transfer to Client-side Interface

The timing of a normal incoming transfer to the Client-side is shown in Figure 2.6. The frame is enclosed by a Start character (S) and by a Terminate character (T). The Start character in this implementation can occur in either lane 0 or lane 4. The Terminate character, T, can occur in any lane.

Figure 2.7 shows a frame of data propagating an error. In this example, the error is propagated in lanes 4 to 7, shown by letter E.



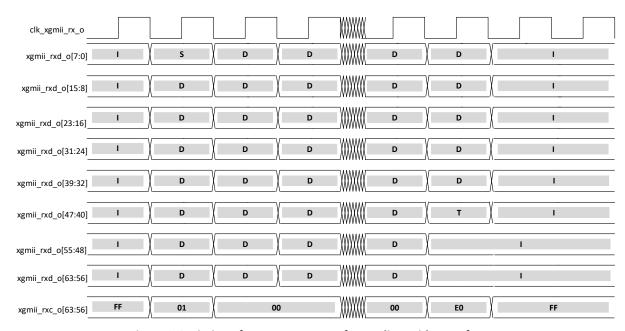


Figure 2.6. Timing of Data Frame Transfer to Client-side Interface

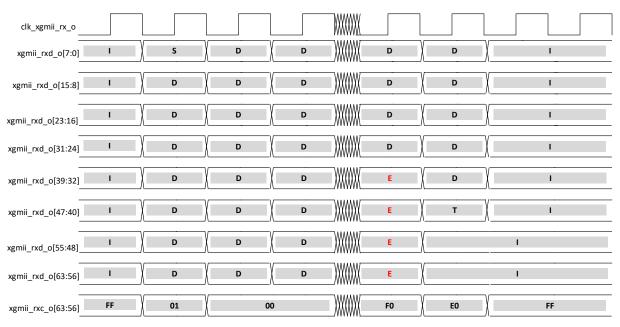


Figure 2.7. Timing of Data Frame Transfer with Error to Client-side Interface



2.7. XGMII to XAUI Translation (Transmit Interface)

The block diagram of the XAUI Soft IP transmit data path is shown in Figure 2.8. The TX interface converts the incoming XGMII data into XAUI-compatible characters. Tx Slip Buffer is optional and when disabled, the XGMII control and data directly goes to Tx Core.

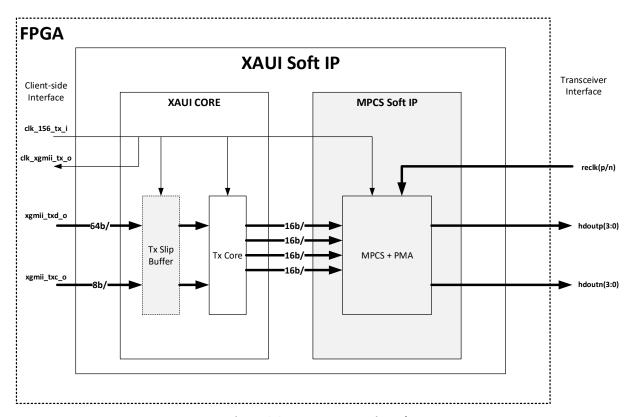


Figure 2.8. XAUI IP Transmit Path

2.7.1. Tx Slip Buffer

The 64-bit XGMII data and control signals at 156.25 MHz are passed to a Tx slip buffer when *Enable Tx Slip Buffer* is checked. This functionality is identical with the one used for the Rx interface.

2.7.2. XAUI Tx Core

After the Slip buffer, the XGMII transmit data and control goes to the Tx Encoder to encode XGMII characters into 8b10b format shown in Table 2.9. The idle generation state machine in the TX encoder converts XGMII /I/ characters to a random sequence of XAUI /A/, /K/ and /R/ characters as specified in IEEE 802.3-2005. XGMII idles are mapped to a random sequence of code groups to reduce radiated emissions. The /A/ code groups support XAUI lane alignment and have a guaranteed minimum spacing of 16 code-groups. The /R/ code groups are used for clock compensation. The /K/ code groups contain the 8b10b comma sequence.

Table 2.9. XGMII to XAUI Code Mapping

XGMII	XAUI	XGMII Control
/I/ = 0x07 (Idle)	Randomized /A/, /R/, /K/ Sequence /A/ = K28.3 = 0x7C	1
	/R/ = K28.0 = 0x1C	
	/K/ = K28.5 = 0xBC (Comma)	
/S/ = 0xFB (Start)	/S/ = 0xFB	1
/E/ = 0xFE (Error)	/E/ = 0xFE	1

18



XGMII	XAUI	XGMII Control
/T/ = 0xFD (Terminate)	/T/ = 0xFD	1
/Q/ = 0x9C (Ordered set)	/Q/ = 0x9C	1
0x00-0xFF (Data)	0x00-0xFF (Data)	0

The random /A/, /K/, sequence is generated as specified in section 48.2.4.2 of IEEE 802.3-2005. In addition to idle generation, the state machine also forwards sequences of |Q| ordered sets used for link status reporting. These sets have the XGMII sequence control character on lane 0 followed by three data characters in XGMII lanes 1 through 3. Sequence ordered-sets are only sent following an |A| ordered set.

The random selection of /A/, /K/, and /R/ characters is based on the generation of uniformly distributed random integers derived from a PRBS. Minimum ||A|| code group spacing is determined by the integer value generated by the PRBS. ||K|| and ||R|| selection is driven by the value of the least significant bit of the generated integer value. The idle generation state machine specified in IEEE 802.3-2005 transitions between states based on a 312 MHz system clock. The TX encoder implemented in the XAUI TX core runs at a system clock rate of 156.25 MHz. Thus, the XGXS state machine implementation performs the equivalent of two state transitions each clock cycle.

2.7.3. MPCS Soft IP (Transmit Mode)

This block receives the 20 bits of data of each XAUI lane. This block serializes the incoming XAUI data. Refer to Multi-Protocol PCS Module (FPGA-IPUG-02118) for the detailed discussion of this block.

2.7.4. Timing of Data Frame Transfer from Client-side Interface

The timing of a normal incoming transfer from the Client-side is shown in Figure 2.9. The beginning of the data frame is indicated by the presence of the Start character followed by data characters in lanes 5, 6, and 7. Alternatively, the start of the data frame can be marked by the occurrence of a Start character in lane 0, with the data characters in lanes 1 to 7. When the frame is complete, a Terminate character completes it. The Terminate character can occur in any lane. The remaining lanes are padded by XGMII idle characters.

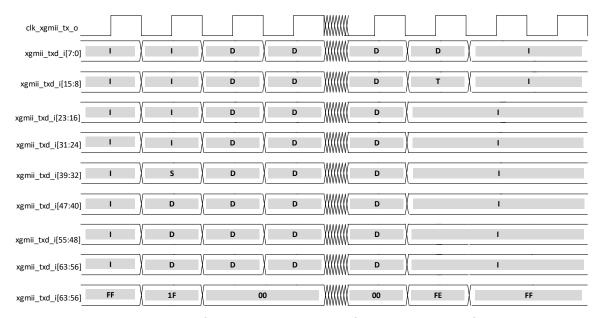


Figure 2.9. Timing of Data Frame Transmission from Client-side Interface



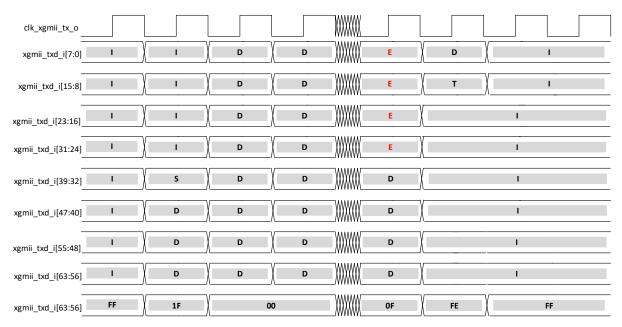


Figure 2.10. Timing of Data Frame Transmission with Error from Client-side Interface

Figure 2.10 depicts a similar frame to that of Figure 2.9 but with error in the frame. The letter E denotes the error, with the relevant control bit set.

2.8. MDIO Interface

The Management Data Input/Output (MDIO) interface provides access to the internal XAUI IP core registers. The register access mechanism corresponds to Clause 45 of IEEE 802.3-2005. This soft IP provides access to XGXS registers 0 to 25 as specified in IEEE 802.3-2005.

An MDIO bus in a system consists of a single Station Management (STA) master management entity and several MDIO Managed Device (MMD) slave entities. Figure 2.11 illustrates a typical system. All transactions are initiated by the STA entity. The XAUI soft IP implements an MMD.

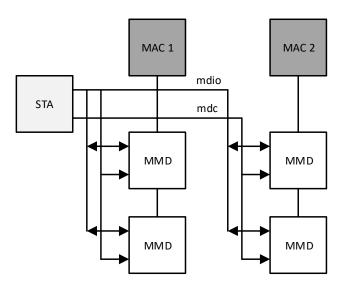


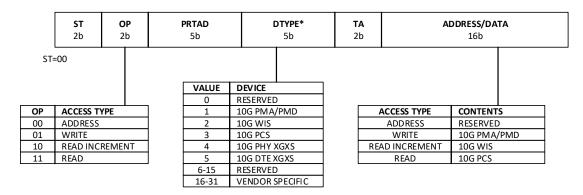
Figure 2.11. A Typical MDIO-Managed System

21



The physical interface consists of two signals: MDIO to transfer data/address/control to and from the device, and MDC, a clock up to 2.5 MHz sourced externally to provide the synchronization for MDIO.

The fields of the MDIO transfer are shown in Figure 2.12.



^{*} If ST=01, this field is REGAD (register address)

Figure 2.12. Field of MDIO Protocol

Each management data frame consists of 64 bits. The first 32 bits are preamble consisting of 32 contiguous 1's in the MDIO. Following the preamble is the start-of-frame field (ST) which is a 00 pattern. The next field is the operation code (OP).

The next two fields are the port address (PRTAD) and device type (DTYPE). Since the physical layer function in 10 GbE is partitioned into various logical (and possibly separate physical) blocks, two fields are used to access these blocks. The PRTAD provides the overall address to the PHY function. The PRTAD default value is 5'h00. Value of this address can be programmed through through prtad i[4:0]. The first port address bit transmitted and received is the MSB of the address. The DTYPE field addresses the specific block within the physical layer function. This soft IP only supports 10G PHY XGXS device type. Device address zero is reserved to ensure that there is not a long sequence of zeros.

If the ST field is 01 then the DTYPE field is replaced with REGAD (register address field of the original clause 22 specification). The Soft IP does not respond to any accesses with ST = 01.

The TA field (Turn Around) is a 2-bit turnaround time spacing between the device address field and the data field to avoid contention during a read transaction. The TA bits are treated as don't cares by the soft IP.

During a write or address operation, the address/data field transports 16 bits of write data or register address depending on the access type. The register is automatically incremented after a read increment. The address/data field is 16 bits.

For an address cycle, this field contains the address of the register to be accessed on the next cycle. For read/write/increment cycles, the field contains the data for the register. The first bit of data transmitted and received in the address/data field is the MSB (bit 15). An example access is shown in Figure 2.13.

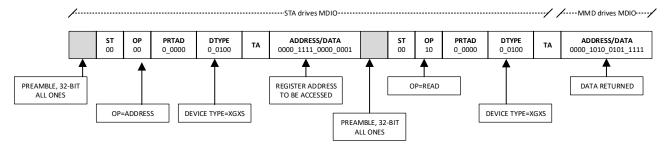


Figure 2.13. Indirect Address Example

FPGA-IPLIG-02160-1 0



3. IP Generation and Evaluation

This section provides information on how to generate the XAUI IP Core using the Lattice Radiant software and how to run simulation and synthesis. For more details on the Lattice Radiant software, refer to Lattice Radiant Software 2.1 User Guide.

3.1. Licensing the IP

An IP core-specific license string is required enable full use of the XAUI IP Core in a complete, top-level design.

When the IP Core is used in CertusPro-NX devices, you can fully evaluate the IP core through functional simulation and implementation (synthesis, map, place and route) without an IP license string. This IP core supports Lattice's IP hardware evaluation capability, which makes it possible to create versions of the IP core, which operate in hardware for a limited time (approximately four hours) without requiring an IP license string. See the Hardware Evaluation section for further details. However, a license string is required to enable timing simulation and to generate bitstream file that does not include the hardware evaluation timeout limitation.

3.2. Generation and Synthesis

The Lattice Radiant software allows you to customize and generate modules and IPs and integrate them into the device's architecture. The procedure for generating the XAUI IP Core in Lattice Radiant software is described below.

To generate the XAUI IP Core:

- 1. Create a new Lattice Radiant software project or open an existing project.
- 2. In the IP Catalog tab, double-click XAUI under the IP, Connectivity category. The Module/IP Block Wizard opens as shown in Figure 3.1.
- 3. Enter values in the Instance name and Create in fields. Click Next.

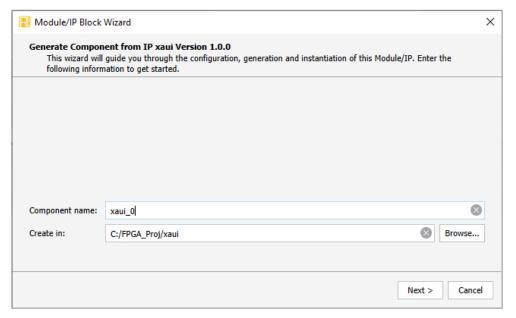


Figure 3.1. Module/IP Block Wizard

4. In the module's dialog box of the **Module/IP Block Wizard** window, customize the selected XAUI IP Core using drop-down menus and check boxes. As a sample configuration, see Figure 3.2. For configuration options, see the Attribute Summary section.



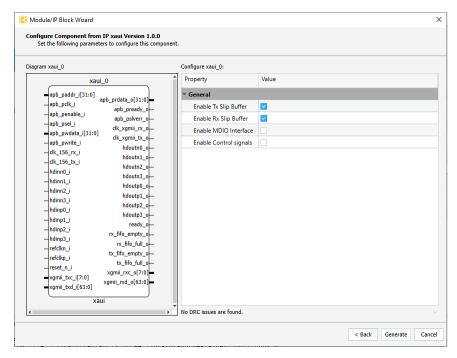


Figure 3.2. Configure User Interface of selected XAUI IP Core

5. Click **Generate**. The **Check Generating Result** dialog box opens, showing design block messages and result as shown in Figure 3.3.

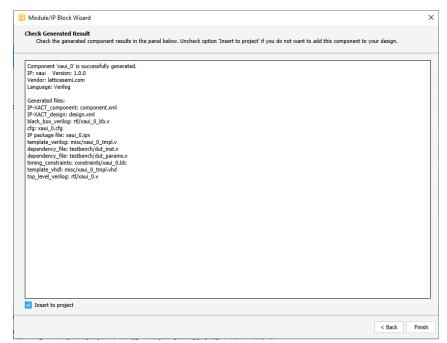


Figure 3.3. Check Generating Result

© 2021 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.

All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice



6. Click the **Finish** button. All the generated files are placed under the directory paths in the **Create in** and **Instance name** fields shown in Figure 3.1.

The generated XAUI IP Core package includes the black box (<Instance Name>_bb.v) and instance templates (<Instance Name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Instance Name>.v) that can be used as an instantiation template for the IP core is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in Table 3.1.

Table 3.1. Generated File List

Attribute	Description
<instance name="">.ipx</instance>	This file contains the information on the files associated to the generated IP.
<instance name="">.cfg</instance>	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/ <instance name="">.v</instance>	This file provides an example RTL top file that instantiates the IP core.
rtl/ <instance name="">_bb.v</instance>	This file provides the synthesis black box.
misc/ <instance name="">_tmpl.v misc /<instance name="">_tmpl.vhd</instance></instance>	These files provide instance templates for the IP core.

3.3. Running Functional Simulation

Running functional simulation can be performed after the IP is generated. The following steps can be performed.

- 1. To run simulation, add the top level testbench file, tb_top.v in the project as a simulation file. Click the **File** tab and select **Add** in the drop down menu.
- Click Existing Simulation File and select <Component name>/testbench/tb_top.v.
- 3. Click the button located on the **Toolbar** to initiate the **Simulation Wizard** shown in Figure 3.4.

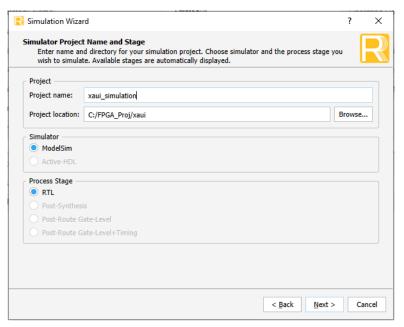


Figure 3.4. Simulation Wizard

24



4. Click **Next** to open the **Add and Reorder Source** window as shown in Figure 3.5.

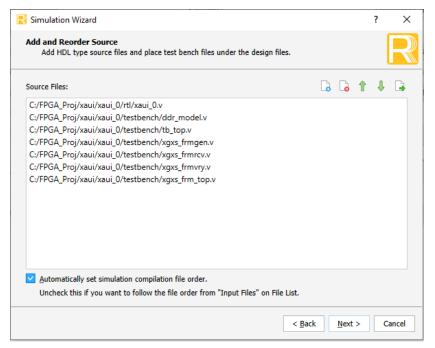


Figure 3.5. Adding and Reordering Source

5. Click **Next**. The Summary window is shown. Click **Finish** to run the simulation. The result of the simulation in our example is provided in Figure 3.6.

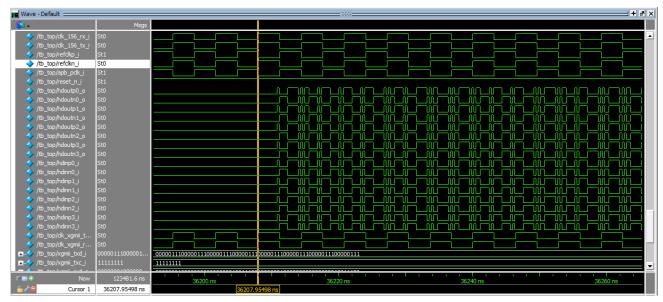


Figure 3.6. Simulation Waveform

Note: It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant software suite.



3.4. Hardware Evaluation

The XAUI IP Core supports Lattice's IP hardware evaluation capability when used with LFCPNX devices. This makes it possible to create versions of the IP core that operate in hardware for a limited period of time (approximately four hours) without requiring the purchase of an IP license. It may also be used to evaluate the core in hardware in user-defined designs. The hardware evaluation capability may be enabled/disabled in the Strategy dialog box. It is enabled by default. To change this setting, go to Project > Active Strategy > LSE/Synplify Pro Settings.



4. Ordering Part Number

The Ordering Part Number (OPN) for this IP Core are the following:

- XAUI-CPNX-U XAUI for CertusPro-NX Single Design License
- XAUI-CPNX-UT XAUI for CertusPro-NX Site License



Appendix A. Resource Utilization

Table A.1 shows resource utilization of XAUI default configuration for the LFCPNX using Lattice Synthesis Engine of Lattice Radiant software.

Table A.1. Resource Utilization

Device	Slice Registers	LUTs	EBRs
LFCPNX	2937	2066	4



References

- CertusPro-NX FPGA-Web Page in latticesemi.com
- Multi-Protocol PCS Module (FPGA-IPUG-02118)
- Lattice Memory Mapped Interface and Lattice Interrupt Interface User Guide (FPGA-UG-02039)



Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.



Revision History

Revision 1.0, October 2021

Section	Change Summary
All	Initial release.



www.latticesemi.com