Aurora 8B/10B v3.1 for Virtex-4 FX FPGA

DS128 April 24, 2009 **Product Specification**

Introduction

The LogiCORE™ IP Aurora 8B/10B core implements the Aurora protocol on the Virtex®-4 FX FPGA. The core can use up to 16 Virtex-4 FPGA RocketIO™ multi-gigabit transceivers (MGTs) running at any supported line rate to provide a low cost, general purpose, data channel with throughput from 1.26 Gbps to over 100 Gbps.

Aurora is a scalable, lightweight, link-layer protocol for high-speed serial communication. The protocol is open and can be implemented by Aurora protocol licensees using any technology. The protocol is typically used in applications requiring simple, low-cost, high-rate, data channels.

The CORE Generator™ software produces source code for Aurora cores with variable datapath width. The cores can be simplex or full-duplex, and feature one of two simple user interfaces and optional flow control.

Aurora cores are verified for protocol compliance using an array of automated hardware and simulation tests. The core comes with an example design.

Features

- General purpose data channels with 1.26 Gbps to over 100 Gbps data throughput
- Supports up to 16 MGTs on the Virtex-4 FX FPGA
- Aurora protocol specification 2.0 compliant (8B/10B encoding)
- Low resource cost ("Resource Utilization," page 7)
- Easy-to-use framing and flow control
- Automatically initializes and maintains the channel
- Full-duplex or simplex operation
- LocalLink (framing) or streaming user interface

LogiCORE IP Facts				
Core Specifics				
Supported Device Family	Virtex-4 FX FPGA (1)			
Resources Used	I/O	LUTs	FFs	Block RAMs
	Varies with channel size See "Resource Utilization," page 7			0
Special Features	Open source; Core is free			
Provided with Core				
Documentation	Product Specification User Guide Getting Started Guide			
Design File Formats	Verilog and VHDL			
Constraints File	`.ucf` (user constraints file)			
Verification	Example Design and Test Bench			
Design Tool Requirements				
Xilinx® Implementation Tools	ISE® 11.1			
Verification	Mentor Graphics® ModelSim® v6.4b			
Simulation	Mentor Graphics ModelSim v6.4b			
Synthesis	XST			
Support				
Provided by Xilinx, Inc.				

1. For more information on the Virtex-4 family, see DS112, *Virtex-4 Family Overview*
2. ISE Service Packs can be downloaded at www.xilinx.com/support/download.htm

Functional Overview

Aurora is a lightweight, serial communications protocol for multi-gigabit links. It is used to transfer data between devices using one or many MGTs. Connections can be *full-duplex* (data in both directions) or *simplex*.

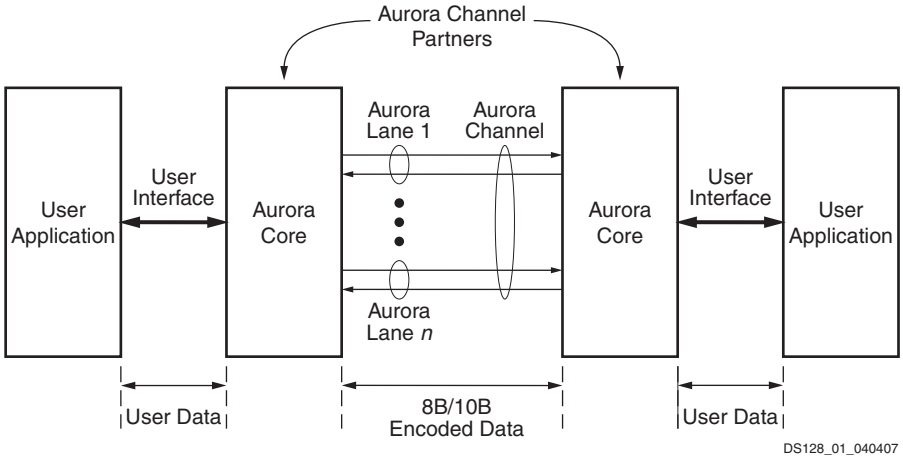

Figure 1: **Aurora Channel Overview**

Aurora cores automatically initialize a channel when they are connected to an Aurora channel partner. After initialization, applications can pass data freely across the channel as *frames* or *streams* of data. Aurora *frames* can be any size, and can be interrupted at any time. Gaps between valid data bytes are automatically filled with *idles* to maintain lock and prevent excessive electromagnetic interference. *Flow control* is optional in Aurora, and can be used to reduce the rate of incoming data, or to send brief, high-priority messages through the channel.

Streams are implemented in Aurora as a single, unending frame. Whenever data is not being transmitted, idles are transmitted to keep the link alive. The Aurora core detects single-bit, and most multi-bit errors using 8B/10B coding rules. Excessive bit errors, disconnections, or equipment failures cause the core to reset and attempt to initialize a new channel.

Applications

Aurora cores can be used in a wide variety of applications because of their low resource cost, scalable throughput, and flexible data interface. Examples of Aurora core applications include:

- Chip-to-chip links: Replacing parallel connections between chips with high-speed serial connections can significantly reduce the number of traces and layers required on a PCB. The Aurora core provides the logic needed to use MGTs, with minimal FPGA resource cost.
- Board-to-board and backplane links: Aurora uses standard 8B/10B encoding, making it compatible with many existing hardware standards for cables and backplanes. Aurora can be scaled, both in line rate and channel width, to allow inexpensive legacy hardware to be used in new, high-performance systems.
- Simplex connections (unidirectional): In some applications there is no need for a high-speed back channel. The Aurora simplex protocol provides several ways to perform unidirectional channel initialization, making it possible to use the MGTs when a back channel is not available, and to reduce costs due to unused full-duplex resources.

- ASIC applications: Aurora is not limited to FPGAs, and can be used to create scalable, high-performance links between programmable logic and high-performance ASICs. The simplicity of the Aurora protocol leads to low resource costs in ASICs as well as in FPGAs, and design resources like the Aurora bus functional model (BFM) with compliance testing make it easy to get an Aurora connection up and running. Contact Xilinx Sales or Auroramkt@xilinx.com for information on licensing Aurora for ASIC applications.

Functional Blocks

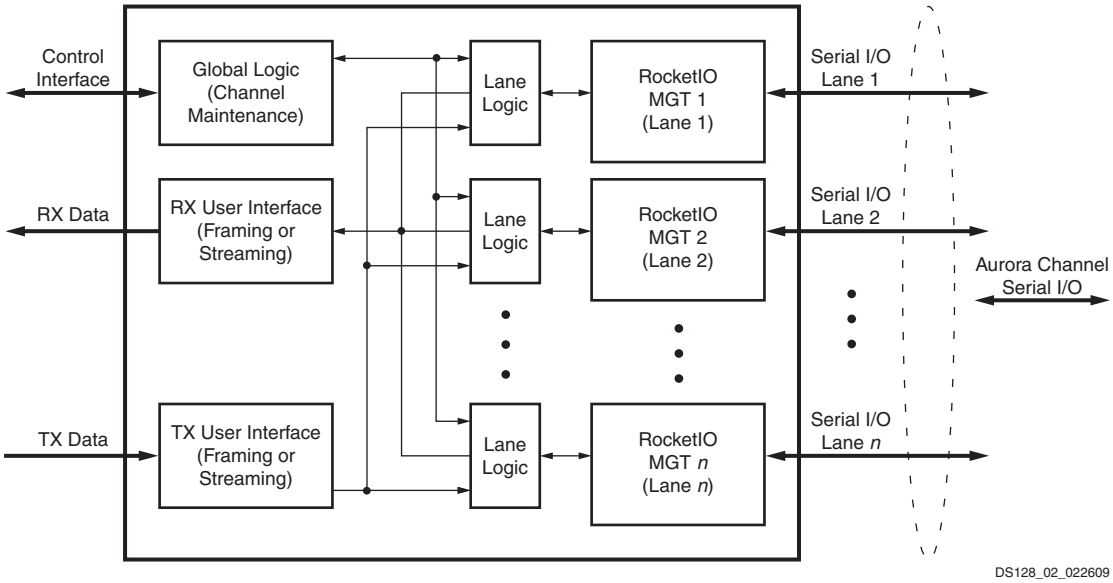

Figure 2: **Aurora Core Block Diagram**

Figure 2 shows a block diagram of the implementation of the Aurora core. The major functional modules of the Aurora core are:

- Lane logic: Each MGT is driven by an instance of the lane logic module, which initializes each individual MGT and handles the encoding and decoding of control characters and error detection.
- Global logic: The global logic module in each Aurora core performs the bonding and verification phases of channel initialization. While the channel is operating, the module generates the random idle characters required by the Aurora protocol and monitors all the lane logic modules for errors.
- RX user interface: The RX user interface moves data from the channel to the application. Streaming data is presented using a simple stream interface equipped with a data bus and a data valid signal. Frames are presented using a standard LocalLink interface. This module also performs flow control functions.
- TX user interface: The TX user interface moves data from the application to the channel. A stream interface with a data valid and a ready signal is used for streaming data. A standard LocalLink interface is used for data frames. The module also performs flow control TX functions. The module has an interface for controlling clock compensation (the periodic transmission of special characters to prevent errors due to small clock frequency differences between connected Aurora cores). Normally, this interface is driven by a standard clock compensation manager module provided with the Aurora core, but it can be turned off, or driven by custom logic to accommodate special needs.

Core Parameters

It is not advisable to change core settings once the core is generated using a set of parameters. The users can customize Aurora cores by setting the parameters for the core using the CORE Generator software. Table 1 describes the customizable parameters.

Table 1: **Core Parameters**

Parameter	Description	Values Supported by Aurora Core
Lanes	The number of MGTs used in the channel.	From 1 to 16 MGTs on the chosen device
Lane Width	The MGTs in the core can be set to use either a 2-byte SERDES or a 4-byte SERDES. For a given line rate, this has the effect of reducing the required clock rate of application logic connected to the Aurora core. The width of the Aurora core datapath is given by Lanes x Lane Width. The parallel clock rate (USER_CLK) for 2-byte lanes is twice the rate for 4-byte lanes for a given line rate.	2 bytes or 4 bytes
Direction	The type of channel the core generates. Can be full-duplex, simplex in the TX direction, simplex in the RX direction, or two separate simplex modules (one TX and one RX) sharing the same MGT.	Full-Duplex Simplex-TX Simplex-RX Simplex-Both
Flow Control	Enables optional Aurora flow control. There are two types of flow control: Native Flow Control (NFC): NFC allows full-duplex receivers to control the rate of incoming data. Completion mode NFC forces idles when frames are complete. Immediate mode NFC forces idles as soon as the flow control message arrives. User Flow Control (UFC): UFC allows applications to send each other brief high priority messages through the channel.	None NFC Immediate NFC Completion UFC UFC and NFC Immediate UFC and NFC Completion
Interface	The user can specify one of two types of interfaces: Framing: The framing user interface is LocalLink compliant. After initialization, it allows framed data to be sent across the Aurora channel. Framing interface cores tend to be larger because of their comprehensive word alignment and control character stripping logic. Streaming: The streaming user interface allows users to start a single, infinite frame. After initialization, the user writes words to the frame using a simple register style interface that has a data valid signal.	Framing (LocalLink) Streaming
Line Rate	The line rate for cores in Virtex-4 devices can be set from 1.26 Gbps to 6.5 Gbps using the CORE Generator software. Other devices use parameters in the source code to set their rate. The *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide* has detailed instructions.	1.26 Gbps to 6.5 Gbps
Reference Clock Frequency	The CORE Generator software accepts parameters to set the reference clock rate for modules in Virtex-4 devices. Other devices use parameters in the source code to set the reference clock rate. The *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide* has detailed instructions.	A selection of legal rates based on the selected line rate and available clock multipliers in the Virtex-4 FPGA MGTs
Reference Clock	MGTs can be fed a reference clock from a variety of dedicated and non-dedicated clock networks. The *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide* explains how to select the best reference clock network for a given application.	REFCLK1 REFCLK2
MGT Placement	The CORE Generator software provides a graphical interface that allows users to assign lanes to specific MGTs. The *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide* includes guidelines for MGT placement for best timing results.	Any combination of MGTs can be selected

Core Interfaces

The parameters used to generate each Aurora core determine the interfaces available (Figure 3) for that specific core. The Aurora cores have three to six interfaces:

- "User Interface," page 5
- "User Flow Control Interface," page 6
- "Native Flow Control Interface," page 6
- "MGT Interface," page 6
- "Clock Interface," page 6
- "Clock Compensation Interface," page 6

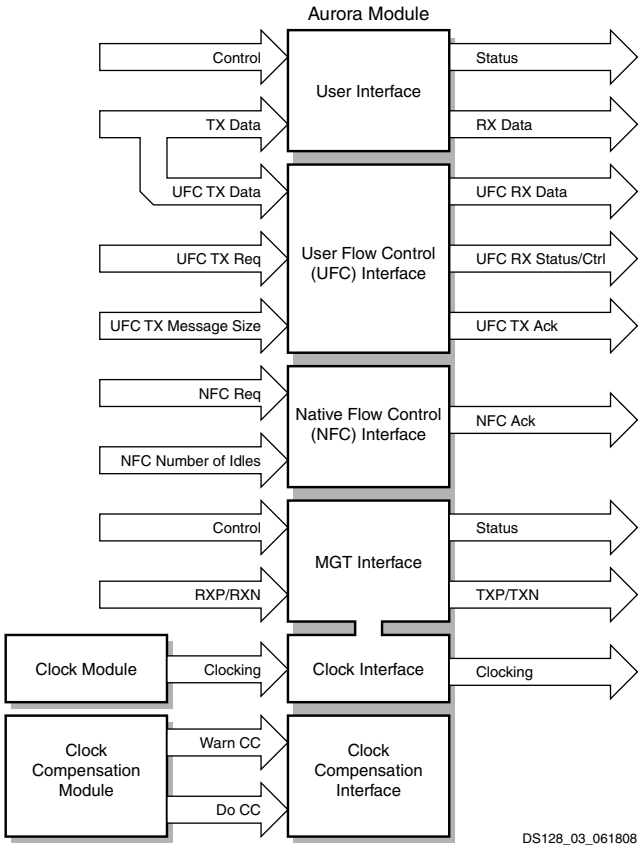

Figure 3: **Top-Level Interface**

User Interface

This interface includes all the ports needed to read and write *streaming* or *framed* data to and from the Aurora core. LocalLink ports are used if the Aurora core is generated with a framing interface; for streaming modules, the interface consists of a simple set of data ports and data_valid ports. Full-duplex cores include ports for both transmit and receive; simplex cores use only the ports they require to send data in the direction they support. The width of the data ports in all interfaces depends on the number of MGTs in the core, and on the width selected for the MGTs.

User Flow Control Interface

If the core is generated with user flow control (UFC) enabled, a UFC interface is created. The TX side of the UFC interface consists of a request and an acknowledge port that are used to start a UFC message, and a port to specify the length of the message. The user supplies the message data to the data port of the user interface; immediately after a UFC request is acknowledged, the user interface indicates it is no longer ready for normal data, thereby allowing UFC data to be written to the data port.

The RX side of the UFC interface consists of a set of LocalLink ports that allows the UFC message to be read as a frame. Full-duplex modules include both TX and RX UFC ports; simplex modules retain only the interface they need to send data in the direction they support.

Note: User flow control is not applicable for Streaming designs.

Native Flow Control Interface

If the core is generated with native flow control (NFC) enabled, an NFC interface is created. This interface includes a request and an acknowledge port that are used to send NFC messages, and a 4-bit port to specify the number of idle cycles requested.

Note: Native flow control is not applicable for Streaming designs.

MGT Interface

This interface includes the serial I/O ports of the MGTs, and the control and status ports of the Aurora core. This interface is the user's access to control functions such as reset, loopback, and powerdown. Status information about the state of the channel, and error information is also available here.

Clock Interface

This interface is most critical for correct Aurora core operation. The clock interface has ports for the reference clocks that drive the MGTs, and ports for the parallel clocks that the Aurora core shares with application logic. For more details on clocking interface, see the *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide*.

Clock Compensation Interface

This interface is included in modules that transmit data, and is used to manage clock compensation. Whenever the DO_CC port is driven High, the core stops the flow of data and flow control messages, then sends clock compensation sequences. For modules with UFC, the WARN_CC port prevents UFC messages and CC sequences from colliding. Each Aurora core is accompanied by a clock compensation management module that is used to drive the clock compensation interface in accordance with the *Aurora Protocol Specification*. When the same physical clock is used on both sides of the channel, DO_CC should be tied Low. Clock compensation interface is managed automatically in the core. For more details on clock compensation interface, see the *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide*.

Resource Utilization

Table 2 and Table 3 show the number of look-up tables (LUTs) and flip-flops (FFs) used in selected Aurora modules. The Aurora core is also available in configurations not shown in the tables. These tables do not include the additional resource costs for flow control or Virtex-4 FPGA calibration logic.

Table 2: **Virtex-4 FX FPGA Resource Usage for Streaming**

Virtex-4 FX FPGA			Streaming			
			Duplex	Simplex		
Lanes	Lane Width	Resource Type	Full-Duplex	TX Only	RX Only	Both
1	2	LUTs	209	102	102	199
		FFs	221	93	138	231
	4	LUTs	360	145	175	320
		FFs	345	140	222	365
2	2	LUTs	395	161	214	364
		FFs	430	163	281	443
	4	LUTs	689	265	376	641
		FFs	691	254	463	716
4	2	LUTs	717	267	390	643
		FFs	811	293	533	825
	4	LUTs	1322	434	718	1152
		FFs	1338	467	897	1364
8	2	LUTs	1382	518	755	1243
		FFs	1580	556	1045	1601
	4	LUTs	2614	822	1412	2231
		FFs	2635	916	1773	2685
16	2	LUTs	2718	983	1423	2402
		FFs	3112	1080	2061	3142

Table 3: **Virtex-4 FX FPGA Resource Usage for Framing**

Virtex-4 FX FPGA			Framing			
			Duplex	Simplex		
Lanes	Lane Width	Resource Type	Full-Duplex	TX Only	RX Only	Both
1	2	LUTs	244	130	124	237
		FFs	335	163	185	347
	4	LUTs	499	175	306	466
		FFs	670	249	441	687

Table 3: **Virtex-4 FX FPGA Resource Usage for Framing *(Cont'd)***

Virtex-4 FX FPGA			Framing			
			Duplex	Simplex		
Lanes	Lane Width	Resource Type	Full-Duplex	TX Only	RX Only	Both
2	2	LUTs	543	192	356	513
		FFs	749	265	498	763
	4	LUTs	1146	277	822	1070
		FFs	1378	505	898	1396
4	2	LUTs	1188	307	881	1101
		FFs	1502	551	960	1509
	4	LUTs	2963	489	2310	2784
		FFs	2717	946	1791	2730
8	2	LUTs	3008	533	2431	2812
		FFs	2953	1037	1934	2958
	4	LUTs	8805	TBD	TBD	TBD
		FFs	6425	TBD	TBD	TBD
16	2	LUTs	8909	989	7552	TBD
		FFs	6961	2006	4922	TBD

Performance

Virtex-4 FX family MGTs have fewer restrictions on line rate, so the speed of Aurora cores in those devices is typically limited by the f_{MAX} of the FPGA design. The configurations shown in Table 2, page 7 and Table 3, page 7 run at 156.25 MHz or higher. For more details on core performance, see *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide*.

Verification

The Aurora core is verified using the Aurora 8B/10B BFM and proprietary custom test benches. The Aurora 8B/10B BFM verifies the protocol compliance along with interface level checks and error scenarios. An automated test system runs a series of simulation tests on the most widely used set of design configurations chosen at random. Aurora cores are also tested in hardware for functionality, performance, and reliability using Xilinx MGT demonstration boards. Aurora verification test suites for all possible modules are continuously being modified to increase test coverage across the range of possible parameters for each individual module

.

Table 4: **Boards Used for Verification**

Test Boards
ML421
ML423

References

1. Xilinx Aurora Web site, www.xilinx.com/aurora:
   - SP002, *Aurora Protocol Specification*
   - UG058, *Aurora 8B/10B Bus Functional Model User Guide* - Contact Auroramkt@xilinx.com
   - UG061, *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide*
   - UG173, *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA Getting Started Guide*
2. Xilinx RocketIO Transceiver User Guides:
   - UG076, *Virtex-4 FPGA RocketIO Multi-Gigabit Transceiver User Guide*

Support

For technical support, go to www.xilinx.com/support. Xilinx provides technical support for this LogiCORE IP product when used as described in the product documentation. Questions are routed to a team of engineers with expertise using the Aurora core.

Xilinx will provide technical support for use of this product as described in the *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA User Guide* and the *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA Getting Started Guide*. Xilinx cannot guarantee timing, functionality, or support of this product for designs that do not follow the guidelines in these documents, or for modifications to the source code.

Any feedback, questions, or concerns about the design can be reported to the Xilinx Technical Support or by opening a WebCase.

Ordering Information

The Aurora core is provided free of charge to licensed users. The licence for the Aurora core is also free and can be obtained by visiting www.xilinx.com/aurora.

There are three steps required to obtain the core:

1. Install Xilinx ISE 11.1. See the ISE product page for instructions if ISE is not already installed.
2. Install Xilinx ISE 11.1 to add version 3.1 of the Aurora core to the list of cores available in the Core Selection window in the CORE Generator software. Instructions for this step are available in the *LogiCORE IP Aurora 8B/10B v3.1 for Virtex-4 FX FPGA Getting Started Guide*.
3. Electronically sign the Aurora Core License Agreement to obtain a license file for the Aurora core. Instructions for this step and the link to the page with the license and the CORE Generator software license file are also at www.xilinx.com/aurora. You must be a registered user on www.xilinx.com to sign the license.

Revision History

Date	Version	Revision
10/26/04	2.2	Initial Xilinx release. Title: Aurora Reference Design v2.2.
04/28/05	2.3	LogiCORE IP Aurora v2.3 release.
01/10/06	2.4	LogiCORE IP Aurora v2.4 release.
09/12/06	2.5	LogiCORE IP Aurora v2.5 release.
11/30/06	2.5.1	LogiCORE IP Aurora v2.5.1 release.
03/01/07	2.6	LogiCORE IP Aurora v2.6 release.
05/17/07	2.7	LogiCORE IP Aurora v2.7 release.
08/22/07	2.7.1	LogiCORE IP Aurora v2.7.1 release.
10/10/07	2.8	LogiCORE IP Aurora v2.8 release.
03/24/08	2.9	LogiCORE IP Aurora v2.9 release.
03/24/08	2.9.1	Post-release updates and corrections.
09/19/08	3.0	LogiCORE IP Aurora v3.0 release. Corrected the placement of the TXP/TXN and RXP/RXN buses in Figure 3, page 5. Added the number of lanes tested in "Verification," page 8.
04/24/09	3.1	LogiCORE IP Aurora v3.1 release. Updated tools to v11.1. Changed title of data sheet to be device specific.

Notice of Disclaimer