ETA Cluster Communications: Physical and Data Link Layers

I. Introduction

The Eight-meter-wavelength Transient Array (ETA) is a radio telescope designed to observe radio transients expected to be produced by a number of high-energy astrophysical phenomena. The ETA system consists of a front-end receiver followed by a digital backend. The front-end includes the Altera S25 board [1], which digitizes the antenna feeds using 125MSPS, 12-bit A/D converters. MICTOR cables transfer the data to the backend cluster of 16 Xilinx ML310 boards [2], described in Figures 1(a), 1(b) and 1(c). The outer ring of twelve ML310s connect to the S25 boards, while the inner four ML310s further process the data and send it to a cluster of four Dell SC430 dual-core servers. A single custom interface board, shown in Figure 2, is used to connect the ML310s to an S25 board, an SC430 server, or both. By using software RAID level 0 over three disks, each SC430 is capable of recording datasets of up to 300GB at a sustained rate of 432 Mbit/sec. An EDT PCI CDa card [3] installed in each SC430 receives the ML310's 16-bit parallel data.

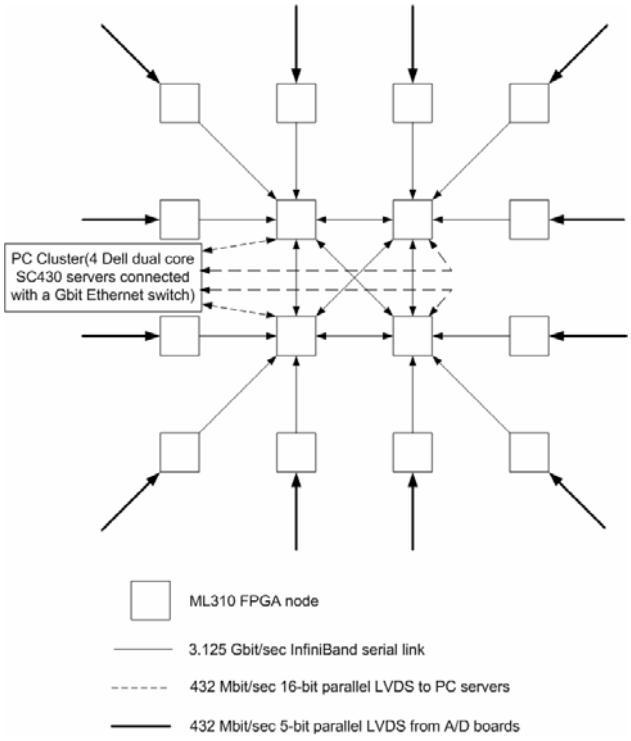

Figure 1(a). Digital backend of the ETA receiver.

Figure 1(b). ETA's ML310 cluster rack.

Figure 1(c). Inside one of the cluster nodes.

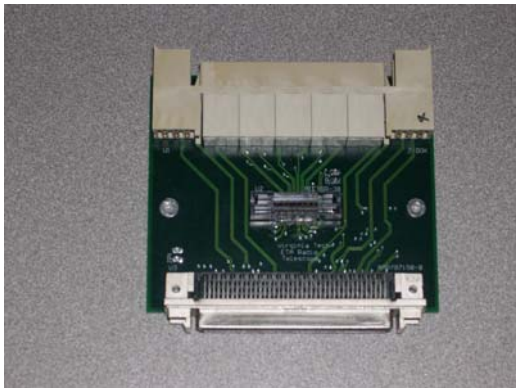

Figure 2. The ML310-SC430-S25 interface adapter board.

II. Design Considerations

A plethora of serial standards are available for high-speed data traffic. Some of these protocols are unnecessarily complicated for applications requiring simple, fixed connectivity. Aurora is a scalable and lightweight protocol at the OSI data link layer used to move data on point-to-point serial links [4]. It provides a transparent interface to the physical serial links and also allows the upper layers of industry standard protocols such as Ethernet, TCP/IP to easily use these links. Aurora is only a protocol and does not define physical interfaces; InfiniBand [5] connectors and cables were chosen to provide the physical layer for the ETA cluster's inter-node network.

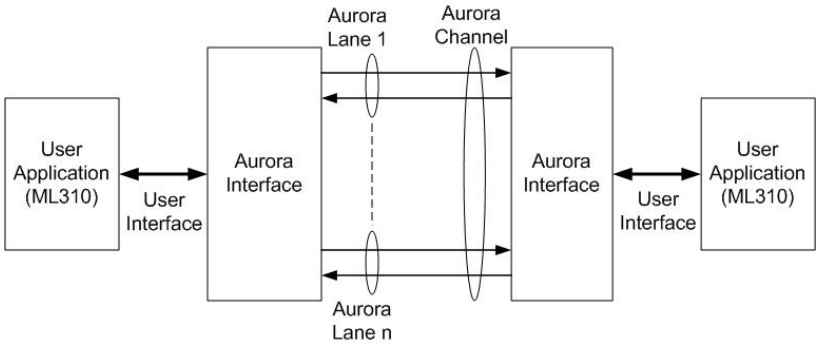

Figure 3. Aurora connection.

Figure 3 shows the setup for the Aurora connection between two ML310s. Each high-speed serial connection between MGTs is called a lane. Any number of lanes can be bonded to create an Aurora channel. A channel is filled with a random idle sequence when it is not used. Aurora uses 8B/10B encoding for DC balance, error detection and to allow control characters in the data stream.

The physical layer consists of InfiniBand cables for the internal and external connections. InfiniBand uses bidirectional point-to-point serial transceivers to avoid the signal skew problems of parallel busses when communicating over relatively long distances. Although InfiniBand is a serial connection, it is very fast, with 2.5 Gbit/sec links in each direction per connection. InfiniBand also supports double and quad data rates for 5 or 10 Gbit/sec respectively. Links use 8B/10B encoding so every 10 bits sent carry 8 bits of data, such that the actual data rate is 4/5ths the raw rate. Thus the single, double and quad data rates carry 2, 4 or 8 Gbit/sec respectively. Links can be aggregated in units of 4 or 12, called 4X or 12X. A quad-rate 12X link therefore carries 120

Gbit/sec raw, or 96 Gbit/sec of user data. Larger systems with 12x links are typically used for various cluster and supercomputer interconnects and for inter-switch connections. InfiniBand uses a switched fabric topology so that several devices can share the network at the same time. Unlike most InfiniBand networks such as System X [6], the ETA cluster does not require external switches.

The ML310 board's high-speed I/O is based on the Xilinx XC2VP30 FPGA's RocketIO Multi-Gigabit Transceiver (MGT) and LVDS capability [5] [7] [8]. The high-speed I/O signals are accessible through two Tyco Z-DOK connectors on the ML310 board named PM1 and PM2. The PM1 connector provides access to the input and output differential signals from the eight RocketIO MGTs. A custom interface board connecting to PM1 and containing eight InfiniBand connectors is shown in Figure 4. The ML310-SC430-S25 LVDS interface board shown in Figure 2 uses the PM2 connector.

Figure 4. The ML310 InfiniBand adapter board.

The steps involved in implementing the Aurora protocol are:

Instantiation of Aurora core: The Aurora core is instantiated using the Core Generator function in the Xilinx tool flow with a single MGT lane. The BREF CLK is selected in the clock option for the top row. ML310 board jumpers are used to connect pins 1-2 on J20 and pins 2-3 on J21 for enabling 156.25/125 MHz clock signals. Another jumper is used to connect pins 1-2 on J10 for LVDS 2.5v functionality.

HDL simulation and implementation of the serial link: The generated Aurora core is instantiated twice for a single I/O. One instantiation interfaces to the host and the other interfaces to the target on the MGT side, which enables a loopback test using an InfiniBand cable as shown in Figure 5.

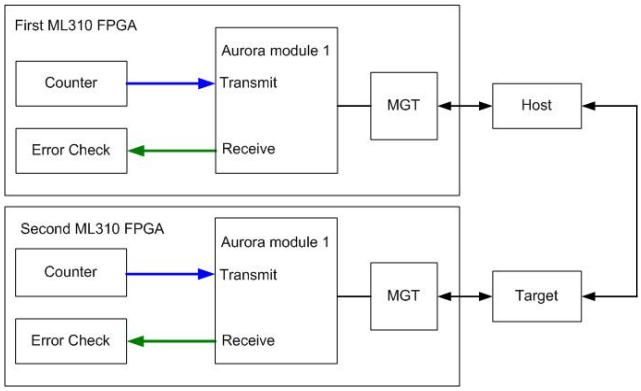

Figure 5. Design implementation.

The HDL code and user constraint file are modified for including the number of MGT lanes required by the design. The top-level module specifies the number of Aurora lanes, and the Aurora cores are instantiated accordingly. For a simple and efficient test of all eight MGT lanes, the counter is modified so that the counter increments by one for the first I/O link and by two for the second I/O link and so on. The design has a latency of 38 cycles and increases with the length of the cable.

Verification using ChipScope: The operation of the Aurora protocol is verified using ChipScope Pro [9], an in-circuit logic analysis tool provided by Xilinx. ChipScope cores are generated and pre-connected in the design, and the ChipScope analyzer interfaces directly to the ILA and ICON cores. The Aurora ChipScope project file is modified to instantiate a core for each MGT. Figure 6 shows the results of testing all eight MGTs.

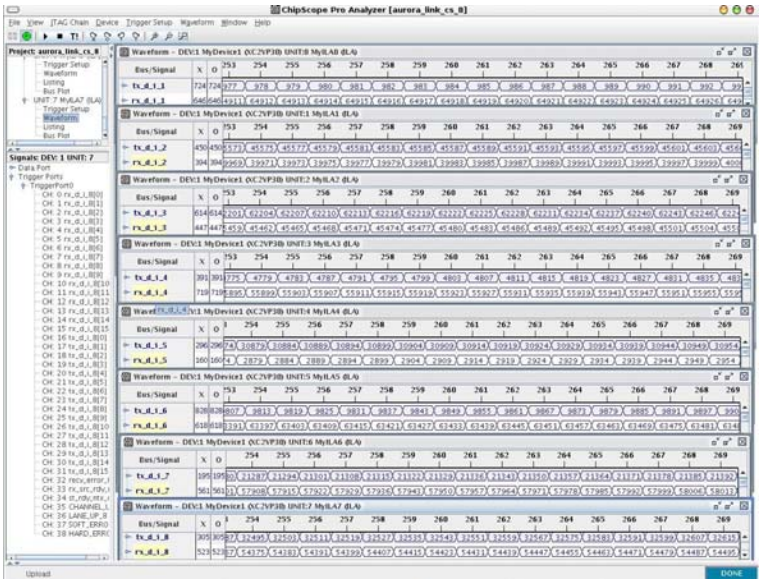

Figure 6. Chipscope Pro verification of all 8 MGTs.

The loopback configuration is implemented between two ML310s, where data is sent from one ML310 board to the second through one MGT lane and the same data is sent back from the first ML310 through another MGT lane as shown in Figure 7(a). The HDL code for the counter is modified and the receive stream of the first MGT lane is connected to the transmit stream of the second MGT lane. The initial test involves sending constant values through the Aurora links and the ChipScope observations are shown in Figure 7(b).

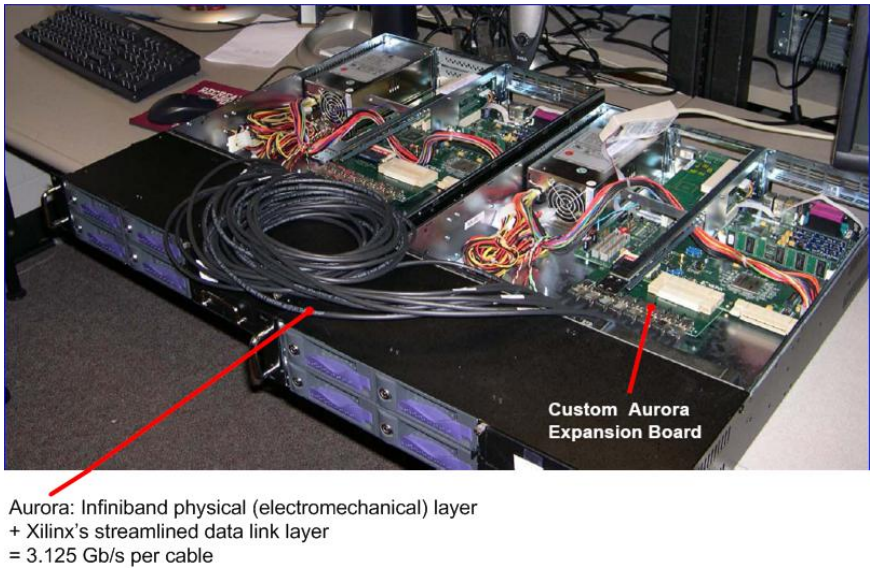

Figure 7(a). Interfacing two ML310s with InfiniBand cables in loopback mode.

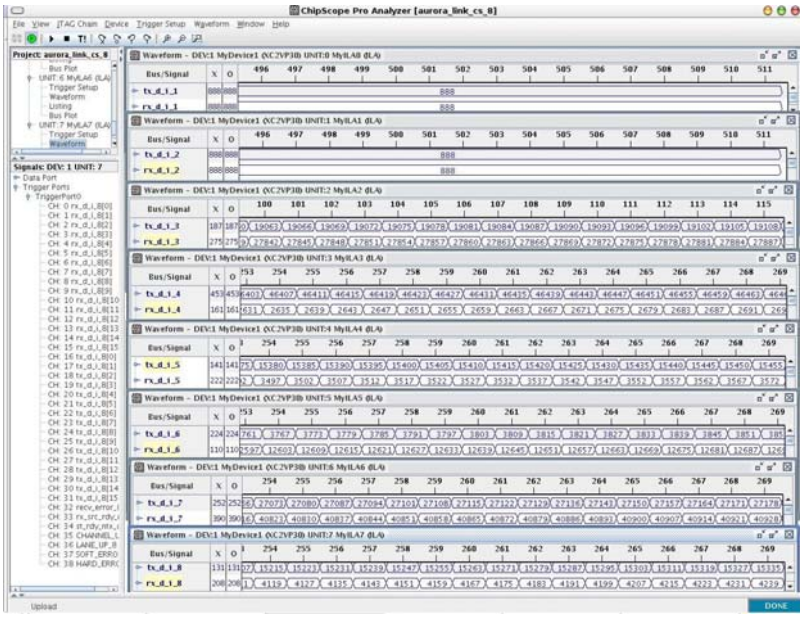

Figure 7(b). Loopback mode between 2 ML310s.

III. Fully Connected Network Test

Figures 8(a) and 8(b) show the connections between the inner four ML310s of the 16-node ETA cluster. Connections are verified by sending a 16-bit counter value to all three MGTs in one ML310. The counter increments by one for the first I/O link, by two for the second I/O link, and by three for the third I/O link.

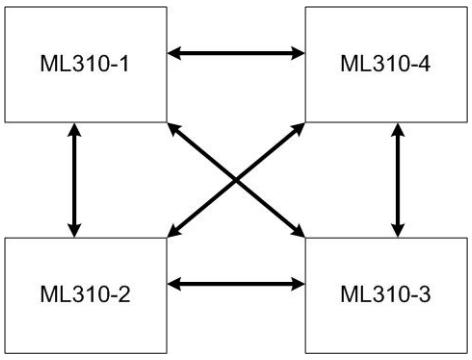

Figure 8(a). Fully connected network of 4 ML310 nodes.

Figure 8(b). Four ML310s in a fully connected network.

On ML310-1, the first MGT lane is connected to the first MGT lane of ML310-2. The second MGT lane is connected to the first MGT lane of ML310-2 and the third MGT lane is connected to the first MGT lane of ML310-4. After initializing ChipScope, it is observed that the Tx and Rx links for the first MGT lane increment by one. The Tx link of the second lane increments by two and the Rx link increments by one. Similarly, the Tx link of the third lane increments by three and the Rx link increments by one. The ChipScope observation of ML310-1 is shown in Figure 9. Data is transmitted and received over 3 meter cables without errors at 3.125 Gbit/sec per link, which exceeds InfiniBand's 2.5 Gbit/sec.

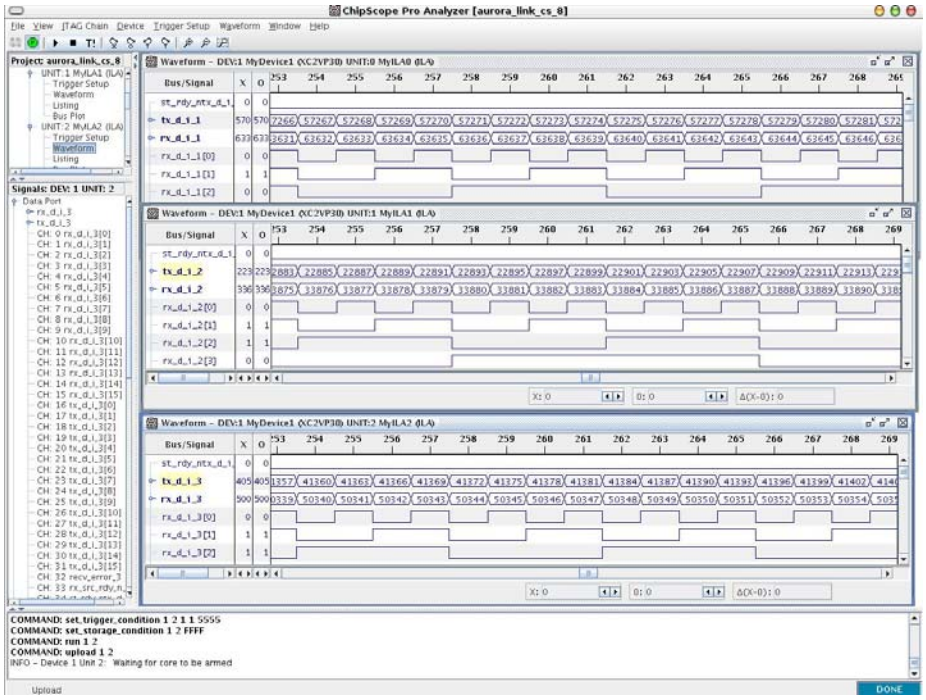

Figure 9. ChipScope Pro observation of ML310-1 in a fully connected network.

References

1. Stratix DSP development kit, www.altera.com/literature/ug/ug_stratix_dsp_kit.pdf
2. ML310 user guide, www.xilinx.com/bvdocs/userguides/ug068.pdf
3. Engineering Design Team, Inc., www.edt.com/pcicda.html
4. Quick start guide to Aurora, www.xilinx.com/aurora/aurora_protocol_member/aurora_protocol_spec_sp002.pdf
5. Using FPGAs to design Gigabit serial backplanes, www.xilinx.com/aurora/a1_xilinx_backplanes_v5.pdf
6. System X, www.tcf.vt.edu/systemX.html
7. RocketIO Transceiver user guide, www.xilinx.com/bvdocs/userguides/ug024.pdf
8. LogiCORE Aurora getting started guide, www.xilinx.com/aurora/aurora_member/aurora_gs_ug173.pdf
9. Chipscope Pro guide, www.xilinx.com/literature/literature-chipscope.htm

Glossary

FPGA: A Field Programmable Gate Array is a semiconductor device containing programmable logic components and programmable interconnects. The programmable logic components can provide the functionality of basic logic gates (such as AND, OR, XOR, NOT) or more complex combinatorial functions such as decoders or simple math functions. In most FPGAs, these programmable logic components (or logic blocks, in FPGA parlance) also include memory elements, which may range from simple flip-flops to large dual-ported memories. See www.xilinx.com

LVDS: Low voltage differential signaling, or LVDS, is an electrical signaling system that can run at very high speeds even over inexpensive, twisted-pair copper cables. LVDS uses the difference in voltage between two wires to signal information. The transmitter injects a small current into one wire or the other, depending on the logic level to be sent. The current passes through a resistor of about 100 to 120 ohms (matched to the characteristic impedance of the cable) at the receiving end and then returns in the opposite direction along the other wire. The receiver senses the polarity of this voltage to determine the logic level. The small amplitude of the signal and the tight electric- and magnetic-field coupling between the two wires reduces the amount of radiated electromagnetic noise. See www.national.com/appinfo/lvds/0,1798,100,00.html

MGT: Multi-Gigabit Transceiver lanes are provided on the top and bottom row of Xilinx XC2VP30 FPGAs. These lanes transmit and receive data using the RocketIO capability of the FPGAs. See www.xilinx.com/bvdocs/whitepapers/wp157.pdf