International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Implementation of Ethernet, Aurora and their Integrated module for High Speed Serial Data Transmission using Xilinx EDK on Virtex-5 FPGA

Chaitanya Kumar N.V.N.S[1], Mir Mohammed Ali[2]

[1, 2]Mahaveer Institute of Science and Technology, Department of Electronics and Communication Engineering, Hyderabad, Telangana, India
[1]chaitanya.nvns[at]gmail.com
[2]mirmohammedali21[at]gmail.com

Abstract: ***This paper implements and establishes serial data transmission techniques like high speed Rocket IO (SERDES/MGT) and Gigabit-Ethernet, employing the architectural features of Virtex-5 FPGA. Ethernet and custom Aurora IP core are integrated to the processor in EDK (Embedded Development Kit) and the registers which are enabled in the custom Aurora IP core are accessed by Ethernet whose assigned data is sent on the Ethernet cable. By using this, the high speed serial data is transmitted at the rate of 1 Gbps for Ethernet, 3.125 Gbps for aurora respectively and 1 Gbps for integrated module of Ethernet and Aurora.***

Keywords: FPGA, Integration of Ethernet and aurora, Aurora Protocol, Multi Gigabit Transceiver, EDK

1. Introduction

Parallel Data transmission technology has been one of the widely used techniques for transfer of digital data, in which parallel lines are used for transmitting and receiving data which would consume more resources and time when compared to serial transmission. So, Serial transmission technology is increasingly used for the transmission of digital data due to its improved signal integrity and high speed.

Field Programmable Gate Arrays (FPGAs) are the digital ICs that are reconfigurable as desired by the designer. That is, any portion of the system can be reconfigured at any time while the rest of the design is still working. Xilinx provides a lot of tools which will help the designers to configure the FPGAs more quickly and easily. With the advancement of FPGAs a new trend of implementing the microprocessors on the FPGAs has emerged in the design community. The ML507 board is such, which supports the embedded processor PowerPC. By configuring the embedded processor it is able to achieve the same transmission and reception speeds.

This paper describes how the data is transmitted serially at higher speeds using various protocols such as Aurora, Ethernet and the integration of Ethernet and Custom Aurora IP core to the processor so that the processor can access both the modules simultaneously and interconnects both Ethernet and Aurora to access each other.

2. Aurora Module

When transmitting and receiving data serially at higher speeds, a protocol has to be followed. Here, Aurora protocol is being utilized which is a physical link layer protocol that uses high speed serial lanes for transmission and reception. The protocol is open and can be implemented using XILINX FPGA technology which is used in applications requiring simple, low-cost, high-rate, data channels. It is used to move data from point to point between devices using one or many transceivers. Aurora utilizes 8b/10b encoding scheme developed by IBM hence called- Aurora 8b/10b protocol where 8-bit words are translated into 10-bit symbols.

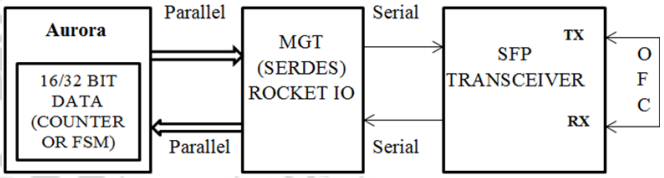

Figure 2.1 Block diagram of aurora module

The data transmission and reception in Rocket- IO uses Aurora protocol on Multi-Gigabit Transceivers (MGTs). A 32-bit data from a Virtex-5 evaluation board is transmitted to other board using Aurora protocol. Multi-Gigabit Transceivers present in the FPGA are configured using Aurora protocol and works at the clock rate of 125MHz (MGT clock). Aurora protocol converts the parallel data into serial data and vice versa (SEDERS). The data from Aurora module is transmitted serially at the rate of 3.125 Gbps through an OFC cable which is looped back as transmission medium for high speed serial data transmission. The receiver module receives data serially and converts it to parallel form.

3. Ethernet Module

Ethernet is the dominant wired connectivity standard. The Xilinx Virtex-5 Ethernet media access controller (Ethernet MAC) block provides dedicated Ethernet functionality. The Ethernet MAC block is integrated into the FPGA as a hard block in Virtex-5 devices and is available in the Xilinx design environment as a library primitive, named TEMAC. The primitive contains a pair of 10/100/1000 Mbps Ethernet MACs.

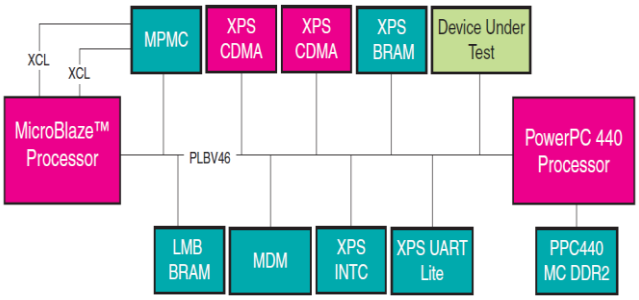

Figure 3.1 Virtex-5 FPGA System with the XPS LL TEMAC as the DUT

The block diagram for Ethernet applicaton requires Hard Ethernet MAC and DDR_SDRAM along with PowerPC 440 Processor for temporary storage purpose. And other peripherals to be selected are UARTlite, Block RAM and RS232 with suitable memory storage capacity inorder to execute the Gigabit Ethernet using Xilinx EDK (Embedded Development Kit).

4. Integration of Ethernet and Custom Aurora IP core

The procedure of binding the Ethernet MAC and Aurora core to the Processor is known as Integration of Ethernet and Aurora. This makes the processor to control both the components simultaneously and interconnects these two peripherals with each other so that, both of them can access one another through the processor. The data present with the custom IP core can be ported onto the Ethernet MAC and is sent on the Ethernet cable to analyze the results. A simple block diagram is shown in figure 4.1 to indicate the above operation.

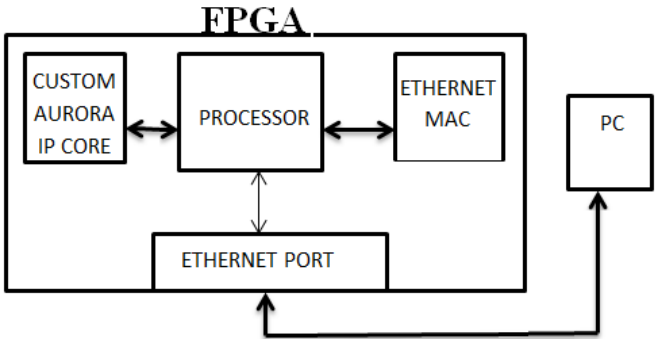

Figure 4.1: Block diagram of Integrated module

The process of integration is carried out in two steps. Initially Hard Ethernet MAC is configured with appropriate peripherals and then a custom IP core is created in EDK. After creating the core, it is instantiated according to the requirement and imported into EDK. After executing the task in EDK, the project is exported to the SDK (Software Development Kit) where the entire hardware portion can be operated with the help of software.

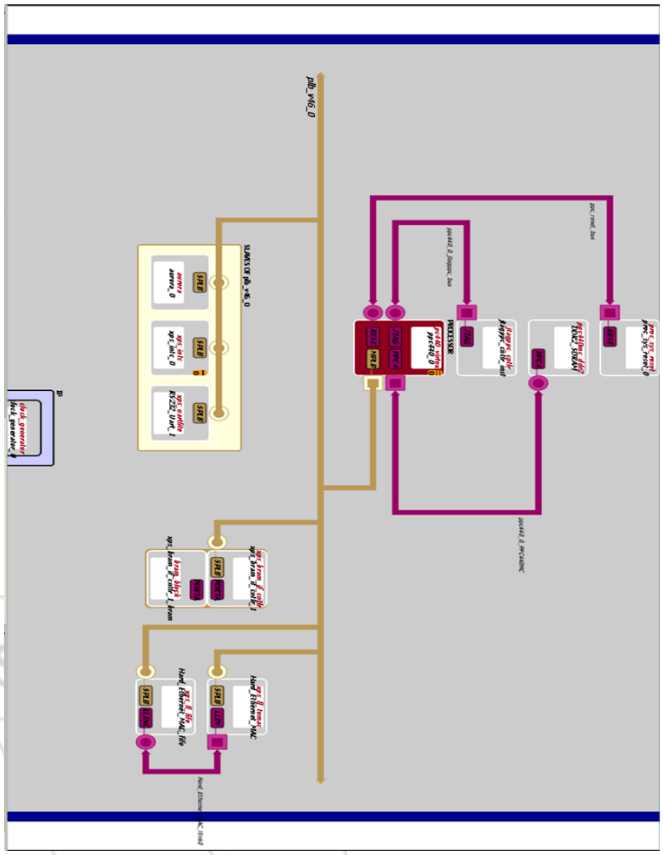

Figure 4.2 Exact block diagram of Ethernet and Aurora Integration

The precise block diagram of integration is given above which is created in the EDK. This depicts the accurate connections between all the necessary peripherals like PowerPC-Processor, DDR_SDRAM, Hard Ethernet MAC, Custom Aurora IP core, RS232- UARTlite and Block RAM. The yellow line is the Processor Local Bus (PLB) which connects the Processor with all other peripherals.

5. Implementation

Initially, Aurora and Ethernet are executed individually and then both are integrated to achieve the desired output. The implementation is classified in three sections as follows:

5.1 Aurora Implementation

Aurora core is generated in CORE Generator provided by Xilinx ISE. The Virtex-5 FPGA board can be configured for specific device, package and speed grade based on the requirements. The 32-bit data to be transmitted is effective if 4 lanes are used out of 16 lanes in the Aurora protocol for this configured FPGA board.

Main Parameters that are required to generate the core are Lane_up, Channel_up, Count signal along with Transmitting data (Tx_D) and Receiving data (Rx_D) signals. Once the 32-bit data is assigned for transmitting purpose, the same data flows continuously. A binary counter is designed with 32 bit width and it counts up to 32 and overflow occurs as VHDL code is written to generate the required output. Synthesis, simulation and implementation of the code are done using Xilinx ISim (ISim is an abbreviation for ISE

Simulator, an integrated HDL simulator used to simulate Xilinx FPGA and CPLD designs) and chip scope PRO analyzer. Tx_D and Rx_D signals are loop backed by using OFC with the help of Small Form factor Pluggable transceiver (SFP) which is a compact, hot-pluggable transceiver used for both telecommunication and data communications applications.

5.2 Ethernet Implementation

Ethernet is configured by enabling the Hard Ethernet MAC *xps_II_temac* as the main component to the single-processor system in the Xilinx EDK. The single-processor system contains two types of processors in which one is hardcore and the other is softcore processor. A hardcore processor type- PowerPC with clock frequency 125.0MHz should be used in synchronous with bus clock frequency of 125.0MHz for optimal communication as Processor Local Bus (PLB) is preferred over AXI bus to interconnect the embedded design efficiently. DDR_SDRAM for temporary storage purpose along with other peripherals such as UARTlite, Block RAM and RS232 with suitable memory storage capacity are utilized in-order to execute the Gigabit Ethernet using EDK.

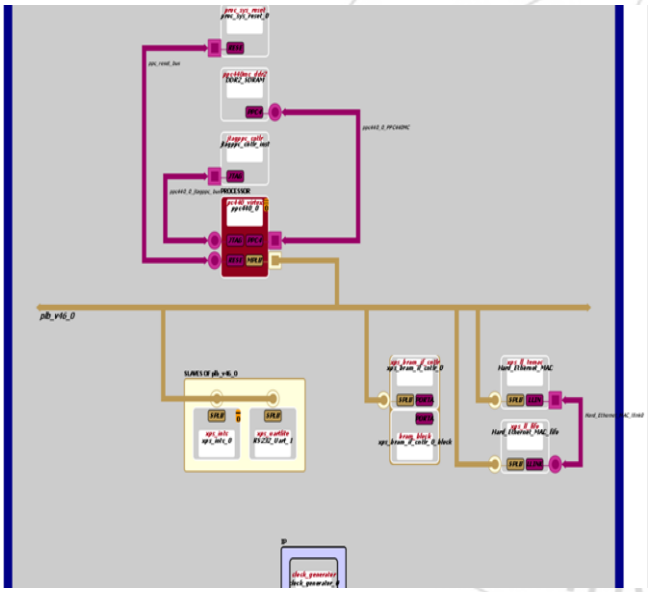

Figure 5.1 Block Diagram of Ethernet Module

After succesful generation of Ethernet Hardware in EDK, the entire project is exported and launched in Software Development Kit (SDK) in order to operate the hardware with the help of programming. In SDK, the desired code is written in the main.c file of the light weight internet protocol template and the FPGA is programmed.

5.3 Integration of Ethernet and Custom Aurora IP core Implementation

Initially for implementing the Integrated module, Ethernet module has to be generated as mentioned in the section 5.2. Now, as Hard Ethernet MAC is generated, custom IP core has to be created to integrate both of them to the processor. To create the custom IP core, a new peripheral is generated with XPS project. After generating the necessary files, the custom IP core named as Aurora will appear in the IP catalog under USER with an ice-cream symbol which denotes the custom core as shown in figure 5.2.

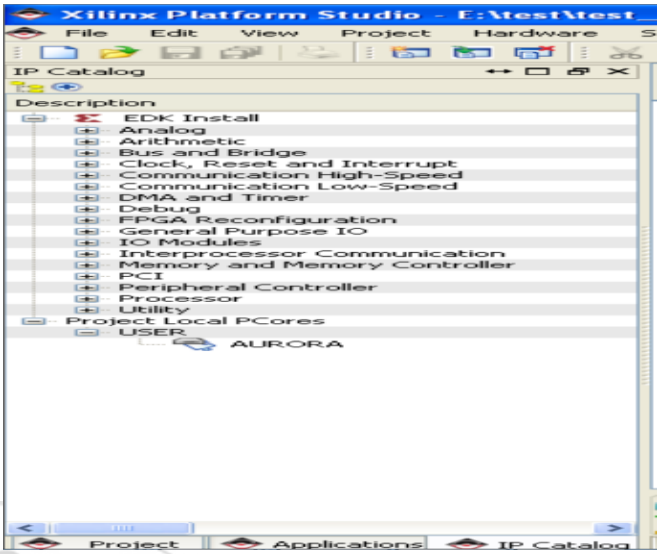

Figure 5.2: XPS IP catalog

And now, it is imported to the EDK after adding all the required files from the Aurora core. The imported peripheral has to be overwritten by HDL files. Once the importing part is completed, the custom aurora IP is added to the bus interface in the highlighted box of figure 5.3.

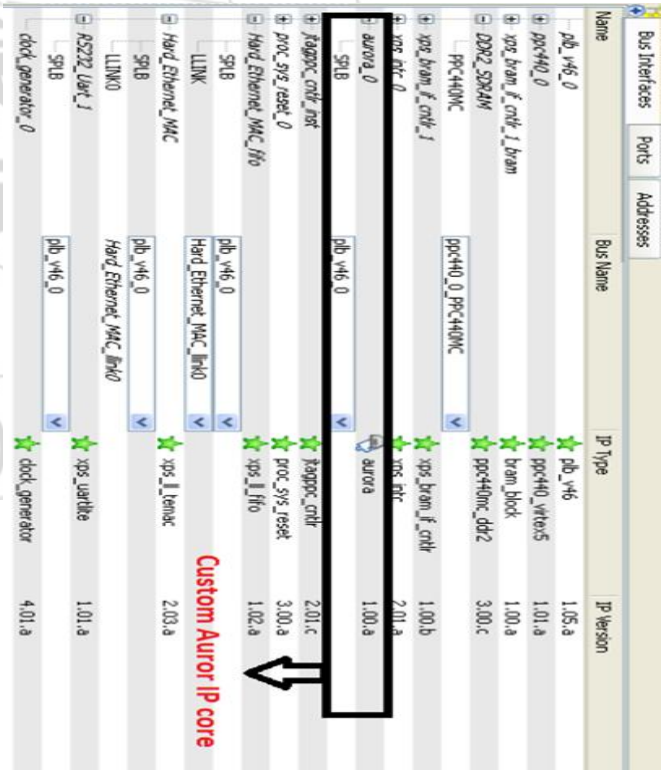

Figure 5.3: XPS wizard with integrated custom IP core

Once the custom IP has been added, connected with the PLB bus and addresses are assigned, then the entire module is arranged and interconnected as shown in the figure 4.2. Now, as the hardware is ready with both the modules integrated to the processor, the project has to be exported and launched on SDK for further execution. In SDK, the desired code is written in the main.c file of the light weight

internet protocol (lwIP) template which is of standalone and POSIX_thread of xilkernel OS to access both Ethernet and Custom Aurora IP core by the processor simultaneously and stored data in the registers of custom IP core are ported on Ethernet.

6. Results

Ethernet, Aurora and Integration of Ethernet and Custom Aurora IP core are performed on the FPGA board Virtex-5 ML507 and the results are as follows:

Figure 6.1 Virtex-5 ML507 board

6.1 Aurora results

The results of the Aurora module for 32 bit data of 4 lanes with Lane_up, Channel_up, Tx_D, Rx_D, count signals are analyzed in Chipscope Pro which is an analyzer software as shown below:-

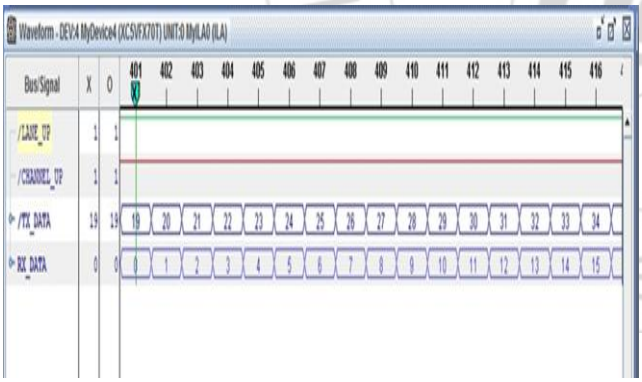

Figure 6.2 Chipscope-Pro waveform of Tx_D, Rx_D.

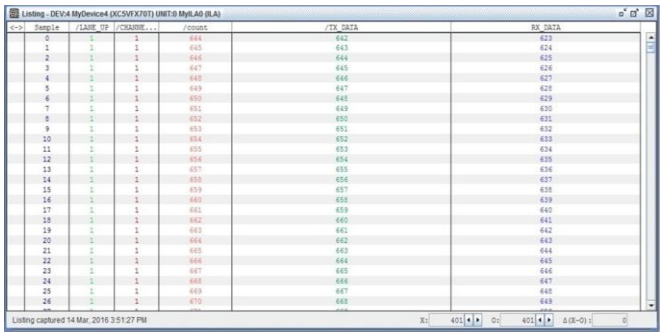

Figure 6.3 Listing Waveform

The verification of the speed which is 3.125 Gbps is done by IBERT core generator.

Figure 6.4 IBERT test

6.2 Ethernet results

The results of the implementation of the Gigabit Ethernet upon launching the hardware, the print results of the lwIP result of Ethernet module with the Board IP, Netmask, Gateway and link speed and the Xilkernel POSIX_thread template can be observed in Hyper Terminal as shown in the below figures.

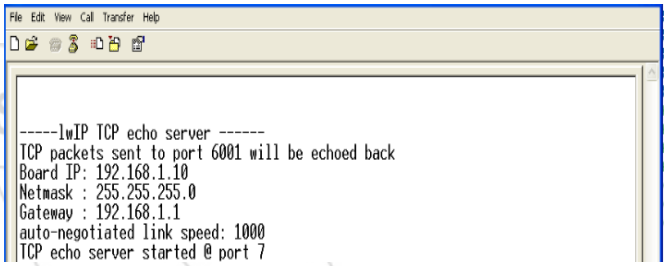

Figure 6.5: lwIP result

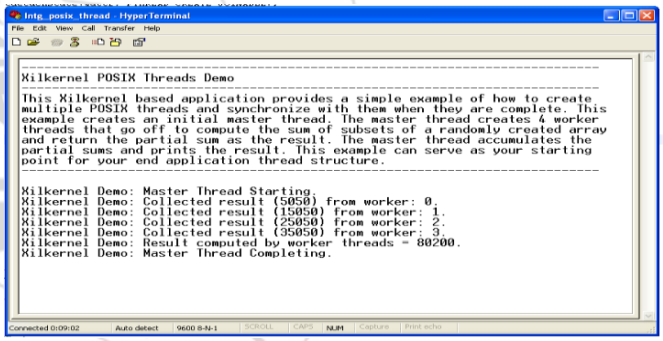

Figure 6.6 Xilkernel result

The Ethernet Ping is executed in Command Prompt by entering the FPGA board IP. The number of packets sent, received and lost along with loss percentage are displayed.

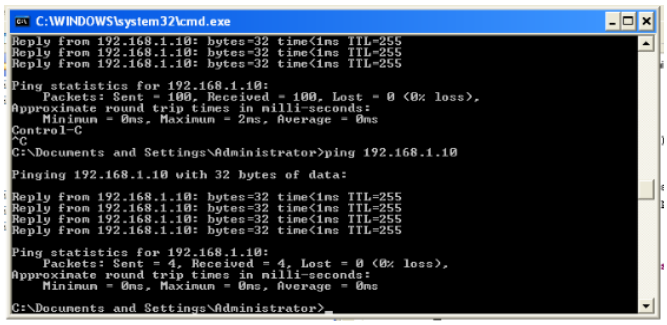

Figure 6.7 Ethernet Ping result

6.3 Integrated Results

Hard Ethernet MAC and custom Aurora IP core are integrated to the single processor on FPGA in EDK and the registers which are enabled in the custom Aurora IP core are

accessed by Ethernet whose assigned data is sent on the Ethernet cable and the results are analyzed.

The register reg0 at an address 0xCFC00000 is assigned with data 0x12. Another register reg5 which is at an address 0xCFC00014 is configured in the code to access the data present in the reg0 with an incrimination +1 and stored as 0x13. Below are the prints of the integrated application which is executed by taking the data from IP core and porting it onto the Ethernet cable.

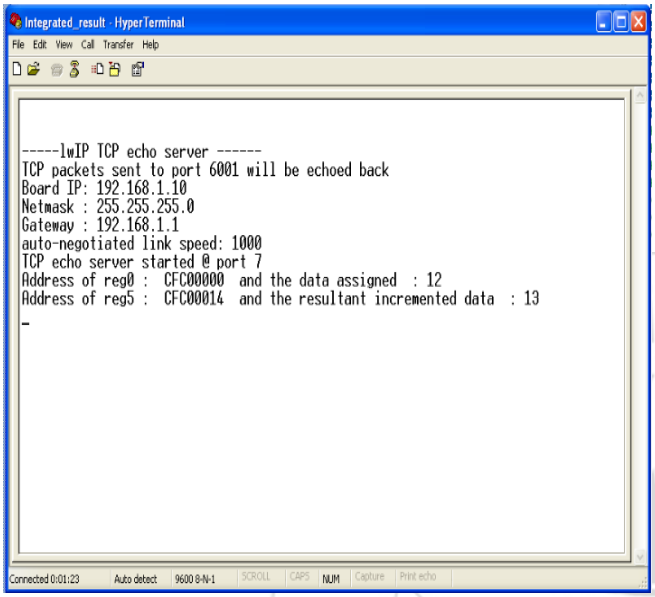

Figure 6.8: Registers output with IwIP in Hyper Terminal

The Ethernet and Aurora Integration Ping is executed in Command Prompt by entering the FPGA board IP. The number of packets sent, received and lost along with loss percentage are displayed.

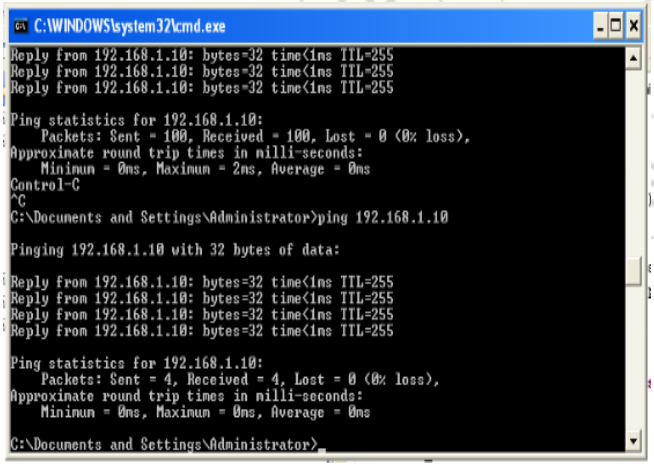

Figure 6.9: Ping result of Integrated module in cmd

The combined result of both Xilkernel and Standalone OS templates which are Xilkernel_POSIX_thread template and lwIP templates are modified and below are the prints in Hyper Terminal.

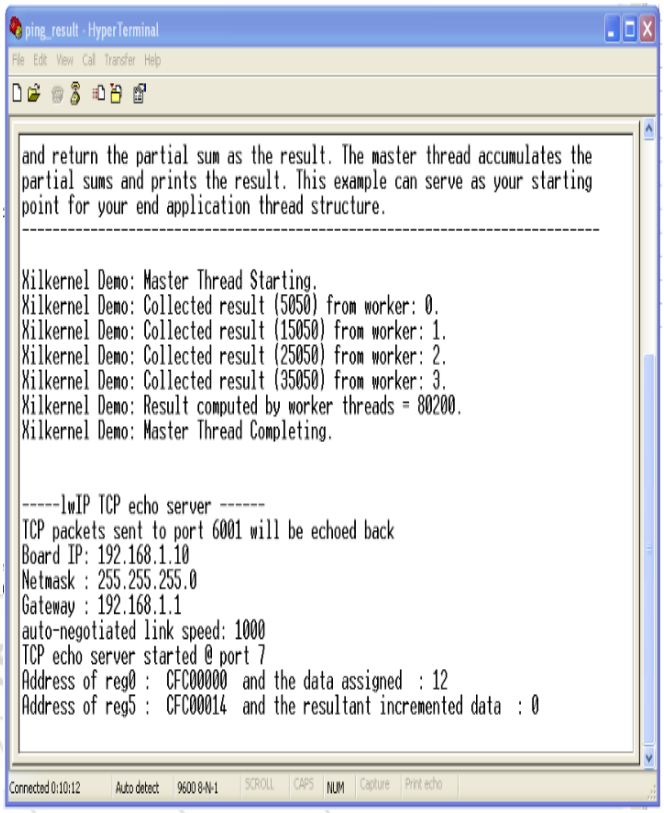

Figure 6.10: Combined result of Xilkernel and IwIP in Hyper Terminal

The Ethernet and Aurora Integration continuous Ping is executed in Command Prompt by entering the FPGA board IP with –t.

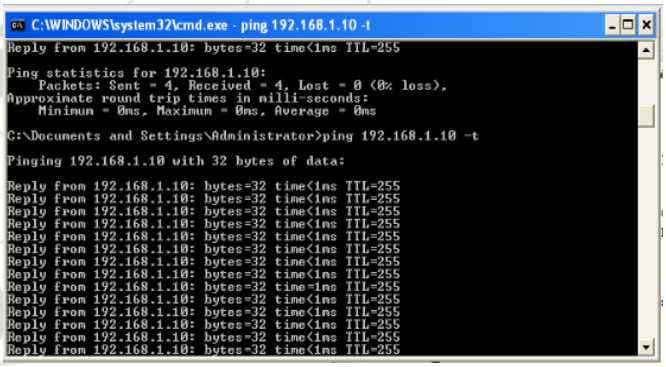

Figure 6.11: Continuous Ping result of Integrated module in cmd

7. Conclusion and Future Improvements

Hard Ethernet MAC and custom Aurora IP core which are integrated to FPGA and the data given to the registers which are enabled in the custom Aurora IP core are sent on the Ethernet cable. Another module of the project involves implementation of Aurora protocol targeted to virtex-5 FX70T FPGA to achieve the above mentioned line rates. The MGT functionality is checked using IBERT tool. All the results are analyzed, Aurora on Chipscope-pro whereas, Ethernet and Integrated results on Hyper terminal.

The hardware utilization of FPGA for Ethernet and Aurora Integration are analyzed with design summary in the EDK as in figure 7.1.

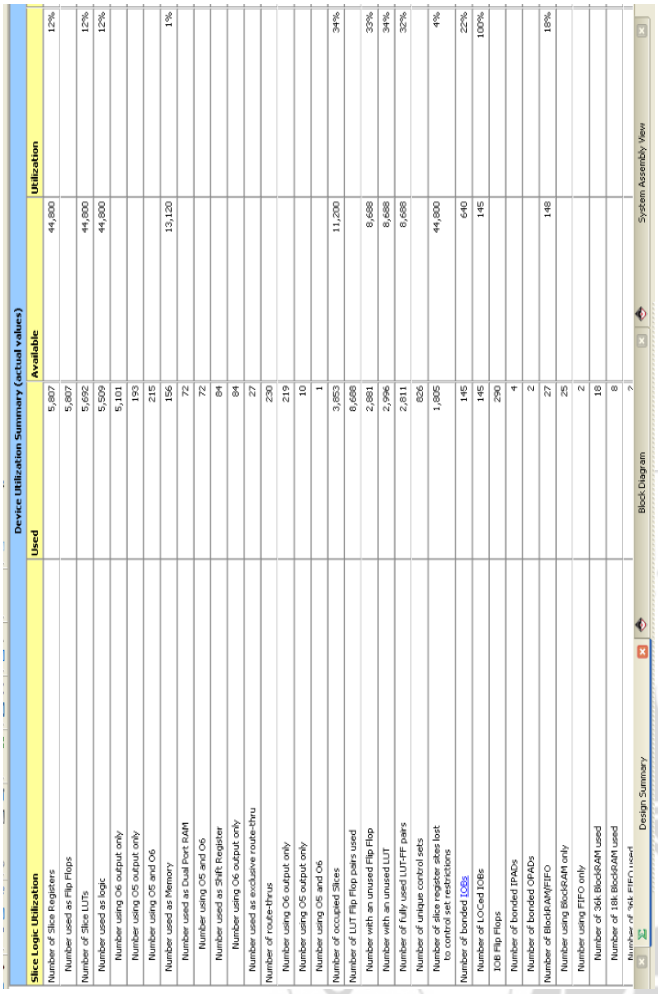

Figure 7.1: Device utilization

With the help of BSD programming, threads can be created to access the packets at TCP/IP level for both server and client where the sent and received packets output could be seen on wireshark. Audio and Video files can be played.

This project used independent aurora link on one GTX DUAL TILE for serial data transmission at the rate of 3.125Gbps using aurora protocol. The speed upto 17Gpbs can be achieved with some other Virtex series boards which support higher line rates using aurora protocol. Much higher data rate can be achieved if all of the 16available Rocket I/OS are used.

References

[1] "High speed serial I/O made simple- A designer's guide with FPGA applications", 1st ed.,Xilinx, San Jose, CA,2005.
[2] Volnei A. Pedroni ,"Circuit Design with VHDL", Cambridge, MA: MIT Press, 2005.
[3] Clive 'Max' Maxfield, "The Design Warrior Guide to FPGA",MA, 2004.
[4] "Virtex-5 special edition",Xcell Journal, Issue 59, , Xilinx, San Jose, CA, 2006.
[5] "EDK Concepts, Tools, and Techniques user guide", Xilinx, San Jose, CA.
[6] "Virtex-5 FPGA Embedded Tri-Mode Ethernet MAC User Guide" (v1.10), Xilinx, San Jose, CA.
[7] "Chip-scope pro integrated bit error ratio test (IBERT) for virtex-5 FPGA GTX" (v2.01a), DS774, Xilinx, San Jose, CA, October, 2011.
[8] "Aurora 8b/10b protocol specification", 2010 by XILINX. http://www.XILINX.com/support/documentation/ip_documentation Aurora_8b10b_protocol_spec_sp002.pdf.

Author Profile

Chaitanya Kumar N.V.N.S completed B.Tech in Electronics and Communication Engineering from Mahaveer Institute of Science and Technology, Hyderabad in June 2016. His current area of interest is on FPGAs.

Mir Mohammed Ali completed B.Tech in Electronics and Communication Engineering from Mahaveer Institute of Science and Technology, Hyderabad in June 2016. His current area of interest is on FPGAs and IoTs.