Research Paper

Volume : 2 | Issue : 3 | Mar 2013 • ISSN No 2277 - 8179

FPGA Implementation of Real Time Ethernet Communication Using RGMII Interface

Engineering

KEYWORDS : FPGA; Ethernet; RGMII; 1 Gigabit Mbps, Real Time

C. Jithendar — VLSI System Design, Anurag group of institutions

Praveen Kumar Landela — Assistant professor, ECE Department, CVSR College of Engineering

ABSTRACT

FPGA-based solution has become more common in embedded system these days. These system need to communicate with external world .Considering high-speed and popularity of Ethernet communication, a reliable real-time Ethernet component inside FPGA is of special value. To that this end, this paper presents a new solution for 1 Gigabit Mbps FPGA-based Ethernet communications with timing analysis. The solution deals with "Reduced Gigabit Media Independent Interface" in its physical layer.UDP are a network protocol which implemented from physical to transport layer. For getting used in real-time applications, timing analysis is done in the communication system. In order to test the component inside FPGA, two different approaches are utilized. Signal measurement in combination with introduced windows based application contributes much in testing and validation phases. We propose consequences of VHDL coding styles on area utilization and speed. Optimization for maximum speed can be achieved by FSM based approach.

1. INTRODUCTION

FPGA (Field Programmable Gate Array)-based systems are playing an increasingly important role in embedded systems. Ever since FPGA has vastly used in embedded systems, communication between FPGA and other parts of system was turned to be an important necessity. Depending on amounts of data which should be transferred, different types of connections can be used. Ethernet communication provides enough bandwidth for most of the high demanded applications. This paper is rs version of a complete version which will be published later. In this paper, in order to solve the communication problem a UDP/IP core on 1000Mb Ethernet is introduced. The solution uses advantages of CBSE (Component-based soft-ware engineering) in design and development phases. Using some testing techniques, validity of the solution is evaluated after implementation phase. Significant of solution is using reduced gigabit media-independent interface (RGMII) in the physical layer of communication protocol.

In this paper, section 2 provides basic theoretical information which is absolutely necessary for implementation of 1000Mb Ethernet communication. Then section 3 describes the solution and the way it is implemented. Section 4 shows what hardware and software tools are used in the implementation and validation of solution. In section 5 validation of solution is discussed. Finally, conclusion is presented in section 6.

2. ETHERNET COMMUNICATION

This section covers OSI (Open System Interconnection) and Protocols of the layers which are discussed in this paper.

2.1 OSI model

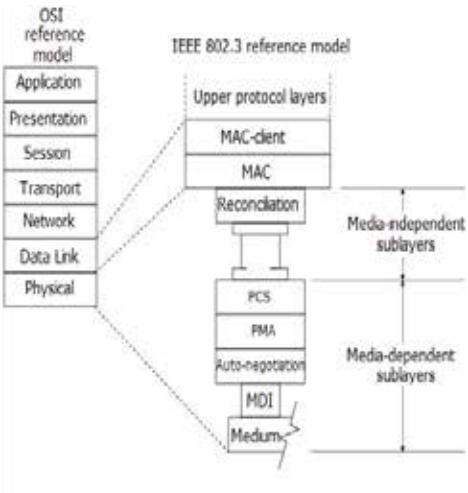

FIGURE 1: OSI REFERENCE MODEL

2.2 PHYSICAL LAYER

The Physical layer indicates how signals can be transmitted on a network, gives interfaces for a network and defines the different types of physical aspects. In order to have an Ethernet connection physical layer uses PHY device. For connecting PHY device to the FPGA, RGMII interface is used.

2.2.1 RGMII CHARACTERISTICS

The RGMII is intended to be an alternative to the IEEE802.3u MII, the IEEE802.3z GMII and the TBI. The principle objective is to reduce the number of pins required to interconnect the MAC and the PHY from a maximum of 28 pins (TBI) to 12 pins in a cost effective and technology independent manner. In order to accomplish this objective, the data paths and all associated control signals will be reduced and control signals will be multiplexed For Gigabit operation, the clocks will operate at 250MHz, and for 10/100 operation, the clocks will operate at 2.5MHz or 25MHz respectively

The RGMII module significantly reduces pin counts between the MAC and the physical layer. .In applications where a number of Ethernet MAC and PHY interfaces are necessary, savings of up to 50% of the pin count are possible. This pin reduction is achieved by multiplexing data and control signals on both edges of the reference clocks. There are two modes of operation, RGMII mode and RTBI mode, with the current mode being selected by the tbi input signal. In RGMII mode, the number of data pins has been reduced from 8 to 4 for both receive and transmit, with a saving of 8 pins in total. This requires the use of both edges of the clock in order to maintain the bandwidth. In RTBI mode, the ten bit receive and transmit code groups are each split into two separate 5 bit groups and driven across the four data pins and the control pin, saving 10 pins in total.

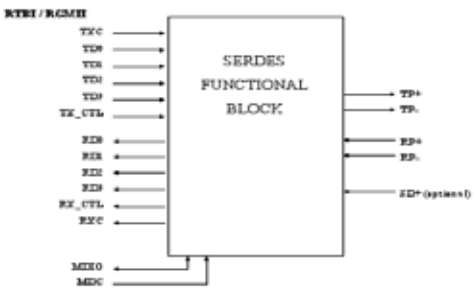

FIGURE 2: PORT MAP OF RGMII

2.2.2 RGMII RECEPTION TIMING

In order to receive packets with RMII Interface, there are some points which should be taken into consideration. According to figure 3 CRS DV signal means that following bits are valid data, but after that preamble can start after undefined number of clock cycles. This problem is solved using one state machine

which after CRS DV signal waits for 7bytes of preamble followed by one byte SFD.

2.2.3 RGMII TRANSMISSION TIMING

In transmission an important point is to change TXEN on rising edge and TXD on falling edge

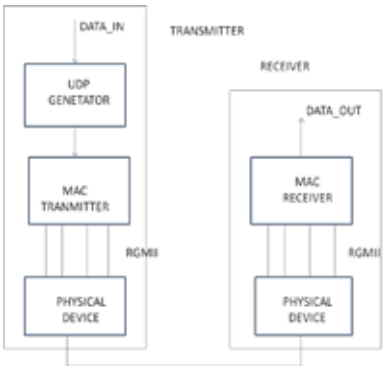

FIGURE3: SYSTEM BLOCK DIAGRAM

2.3 DATA LINK LAYER (MAC)

One of two sub layers of the Data Link layer is the Media Access Control Layer (MAC). The MAC sub layer uses Carrier Sense Multiple Access protocol which has Collision Detection ability (CSMA/CD) to ensure sent signals from different stations over the same channel do not collide. Considering that IEEE802.3 which is one of the world's most used protocols is of the CSMA/CD type, importance of CSMA/CD class becomes more obvious. The MAC layer is responsible for delivering data packets over a shared channel. All MAC fields should be sent by sequence and their sizes are fixed except data field which could vary from 46 to 1500 bytes

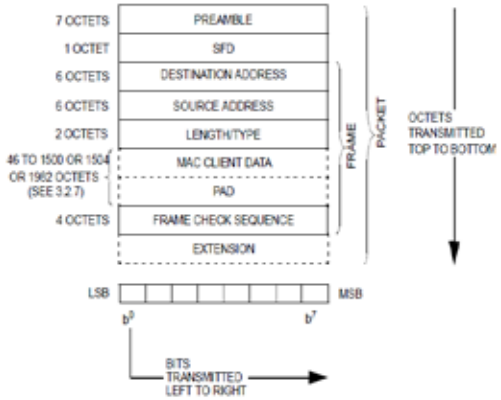

FIGURE 4: MAC PACKET FORMAT

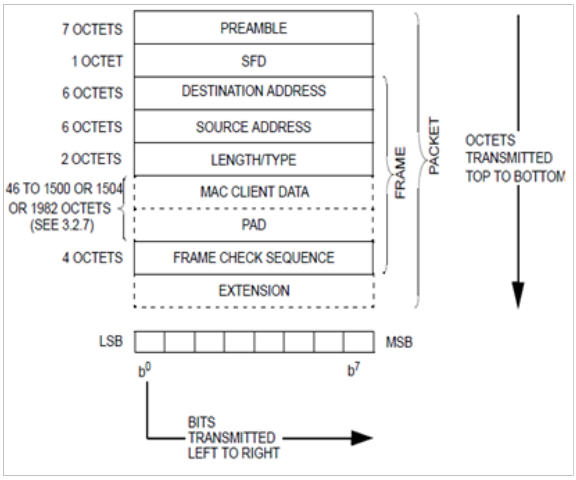

FIGURE 5: MAC TRANSMITTER

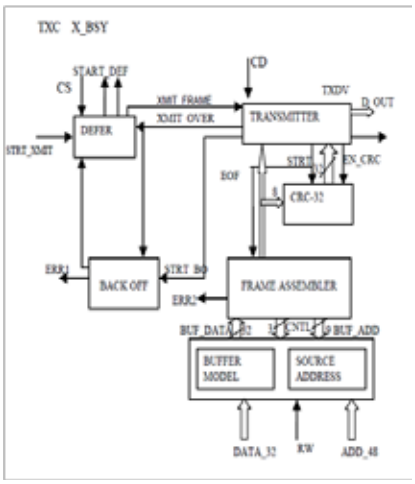

FIGURE 6: MAC RECEIVER INTERFACE

2.3.1 CRC

CRC (Cyclic Redundancy Check) is a popular and reliable technique of error detection in data communication systems. Which can detect a large number of errors? This method is based on polynomial arithmetic. CRC is used in order to distinguish damaged frames from correct ones. CRC-32 is one of common CRC standards used for Ethernet which defines polynomial function G(x) as a common generator polynomial.

$G(x) = x32 + x26 + x23 + x22 + x16 + x12 + x11 + x10 + x8 + x7 + x5 + x4 + x2 + x1 + x0.$

X represents bit value and superscript of x shows the position of bit in a bit stream. CRC should be calculated by both MAC sender and receiver and in a case that they do not have equal values receiver should discard the packet.

2.4 TRANSPORT LAYER

UDP is such a simple transport protocol that allows applications to send datagram and handle translation between ports and sockets. UDP has twofold ports: source and destination; Also its segments consist of 8-byte header which is shown in figure.

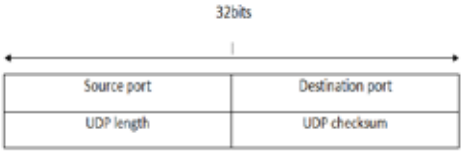

FIGURE 6: THE UDP HEADER

3. PHY MANAGER

Initiating Ethernet PHY is the first step in establishing communication. Based on desired communication type, inner PHY registers should be manipulated. PHY Manager Component is responsible for initiating Ethernet PHY. It sends I2C (Inter-Integrated Circuit) commands to PHY and prepares it for specific speed and duplex mode. MII or RMII is chosen using these I2C commands. I2C uses two lines (data and clock) for manipulating registers that are inside PHY.

3.1 UDP SENDER

This component uses three synchronized processes to make a packet which is sent out by using two lines of data. One process is responsible for TXEN output, the other sends bits out and the last one is a state machine. Diffeerent layers of network protocols are built using this state machine. Fig-ure 10 shows preamble state that is inside the state machine which generates appropriate MAC preamble. As it is clear in the figure after sending preamble, state changes to MAC destination address. This component starts working after a pulse load input is high. It sends all header data such as MAC header, IP header and UDP header

and in the need of sending data field sends out a request and the communication layer provides a byte of data for each request. In the meantime UDP Sender makes CRC and attaches four bytes of CRC to the end of the packet. While UDP Sender sends the packet, the output "Is Active" is high to show other components that sender is not ready to receive new order for sending packet.

4. EXPERIMENT CONSTRAINTS

The solution is synthesized and transferred into a FPGA by Xilinx ISE which is from Xilinx Company. A custom board is used for testing purpose. Specifications of the board components are described. The FPGA which is used in this project is Xilinx SPARTAN 3E DSP (Digital signal processing). This FPGA, in addition to the features of SPARTAN 3E family, is using 90 nm process technologies which are consecutive to more bandwidth. Moreover, in comparison to 3E family, this family of Xilinx FPGA's are using an additional block RAM. This RAM has some additional output registers which help it to perform faster

5. TEST AND DEBUG

In order to test the presented solution, two steps have been taken. First, output signals of FPGA are measured using Picoscop. Second, communication between FPGA and computer is tested by using a windows application.

6. SUMMARY AND CONCLUSIONS

In this work, we introduce a new solution for Ethernet communication in FPGAs. The solution uses RGMII interface in physical layer. Although RGMII interface uses fewer pins for communicating with Ethernet PHY, it has no other advantages and when there is no hardware limitation, it may not worth it to develop a new component for interfacing RGMII inside FPGA because there are already made IP cores for MII and GMII. For real-time applications FPGA is a reliable option since timing analysis can be done easily. The program is converted to hardware blocks which do not require operating system to be run and do not accept interrupts. Actually FPGA is as flexible as software and as reliable as hardware. Some problems such as lack of knowledge about synthesizer and hardware connection problems make it so difficult to develop components in FPGA.

REFERENCE

[1] Davicom datasheet 10/100 mbps fast ethernet physical layer tx/fx single chip transceiver. pages 1-41, September 2008. | [2] Micrel KSZ8873MML datasheet 3-port 10/100 | integrated switch with phy and frame buffeer. page 1, | May 2005. | [3] Andreas Lofgren, Lucas Lodesten, Stefan Sjoholm, Hans Hansson. An analysis of FPGA-based UDP/IP stack parallelism for embedded Ethernet connectivity. NORCHIP Conference, November 2005. | [4] Andrew S. Tanenbaum. Computer Networks. Prentice Hall. 4:0-13-066102-3, March 2003. | [5] Chris Borrelli. Ieee 802.3 cyclic redundancy check. pages 1 { 2, march 2001 | [6] Christophoros Kachris Design and Implementation of a TCP/IP core for recon_gurable logic. Technical University of Crete Electronic and Computer Engineering Department, July 2001. | [7] Cisco Systems. Internetworking technology handbook,2010. | [8] IEEE. Ieee std 802.3-2008. page 49, 2008. | [9] Ivica Crnkovic. Component-based software engineering new challenges in software development. Proceedings of the 25th International Conference on Information Technology Interfaces. June 2003. | [10] L. A. Jorgen Lidholm, Fredrik Ekstrand. Two camera system for robot applications; navigation. 13th IEEE International Conference on Emerging Technologies and Factory Automation, pages 1 { 2, march 2008.. | [11] Miahrugger. Raw ethernet packet sending. The Code Project. October 2003. | [12] Nikolaos Alachiotis, Simon A. Berger, Alexandros Stamatakis. E_cient PC-FPGA Communication over Gigabit Ethernet. 10th IEEE International Conference on Computer and Information Technology, 2010. |