Volume 4, Issue 2, April 2017 ISSN: 2348-8565 (Online)

IJMERT

International Journal of Modern Engineering and Research Technology

Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

VLSI Implementation of UDP/IP Protocol

Ashish Rai
M. Tech. Research Scholar
Takshshila Institute of Engineering & Technology
Jabalpur (M.P.), [INDIA]
Email: ashishraimay23@gmail.com

Mohit Khilwani
Assistant Professor
Department of Electronics and Communication Engg.
Takshshila Institute of Engineering & Technology
Jabalpur (M.P.), [INDIA]
Email:mohitkhilwani@takshshila.org

ABSTRACT

The User Datagram Protocol is a transport protocol for sessions that need to exchange data UDP/IP in the three layers of OSI model those are data link layer, network layer and transport layer. This paper is about UDP/IP (user datagram protocol / internet protocol) stack protocol in FPGA (field programmable gate array), which is widely used in voip (voice over internet protocol), video conference, Ethernet connectivity and LAN network applications. Due to use of FPGA board the less number of resources are required. We are implementing This architecture is implemented using XILINX ISE (integrated synthesis environment) tool and synthesized to a SPARTEN-3E FPGA.

Keywords:— *FPGA, Verilog, OSI Layer, UDP/IP.*

I. INTRODUCTION

UDP stands for User Datagram Protocol. UDP provides an unreliable packet delivery system built on top of the IP protocol. As with IP, each packet is an individual, and is handled separately. Because of this, the amount of data that can be sent in a UDP packet is limited to the amount that can be contained in a single IP packet. Thus, a UDP packet can contain at most 65507 bytes (this is the 65535-byte IP packet size minus the minimum IP header of 20 bytes and minus the 8-byte UDP header). UDP packets can arrive out of order or not at all. No packet has any knowledge of the preceding or following packet. The recipient does not acknowledge packets, so the sender does not know that the transmission was successful. UDP has no provisions for flow control packets can be received faster than they can be used. We call this type of communication connectionless because the packets have no relationship to each other and because there is no state maintained. The destination IP address and port number are encapsulated in each UDP packet. These two numbers together uniquely identify the recipient and are used by the underlying operating system to deliver the packet to a specific process (application). Each UDP packet also contains the sender's IP address and port number.

II. NETWORK PROTOCOL

Protocol is a set of rules that govern data communications. A protocol defines what is communicated, how is communicated and when it is communicated. The key elements of protocols are syntax, semantics and timing [5]. There are various types of protocol slike TELNET (Telecommunication Network), FTP (file transfer protocol), SMTP (simple mail transfer protocol) and DNS (domain name system) come under application layer of OSI (open system interconnection) model, UDP and TCP come under transport layer and IP and ARP come under network layer.

Usually Internet protocol delivers messages to destination which is selected by unique IP address. ICMP (Internet control message protocol) is used by network devices like routers to send error messages. It is nothing but a network diagnostics protocol and is used to report problems.

III. Internet Protocol Stack

This section introduces the basic components of the Internet protocol stack and relates the stack to the ISO OSI reference protocol stack model. The model of the Internet protocol stack is illustrated in the figure below. It is not a protocol it is just a theoretical model for understanding and designing a network architecture. It consists of seven layers which are related to each other, which defines a part of process of moving information across a network. Each layer serves the layer above it and served by the layer below it.

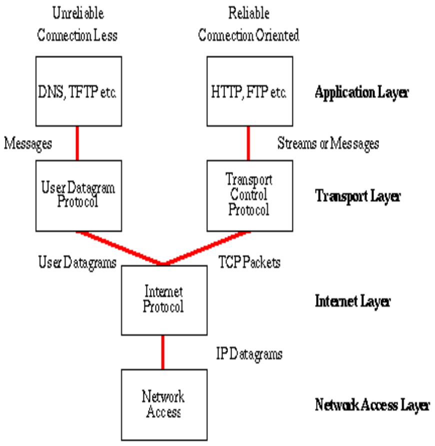


Figure 1: OSI Layer

The TCP/IP protocols correspond to the Transport layer and Network layer of the OSI model. The TCP/IP protocol suite defines two standard transport protocols: TCP and UDP. TCP (Transmission Control Protocol) implements a reliable data-stream protocol, whereas UDP (User Datagram Protocol) implements an unreliable data-stream protocol.

3.1 Physical Layer

It deals with the mechanical and electrical specifications of the data connection. It also defines relationship between device and physical transmission medium. It establishes and terminates the connection between two directly connected nodes, it is flow controlling layer. It predicts transmission mode that is whether simplex, half duplex or full duplex.

3.2 Data link layer

It transforms the physical layerto a reliable link and exchanges data within networks by detecting and possibly correcting errors which may occur in physical layer. Data link packet frame is the basic unit of data transfer for this layer. Frame carries the destination and source link address and other control information in the header.

3.3 Network layer

It is responsible for source to destination delivery of the packet across multiple networks, whereas the data link layer oversees the delivery of packet within same networks. If two systems are connected to the same link, there is no need of network layer. IP is the most important protocol of this layer. It converts logical network address to physical machine address.

3.4 Transport layer

It is responsible for process to process delivery of the entire message. A process is an application program running on the host. It treats each packet independently. The transport layer ensures that the whole message arrives intact and in order, overseeing both error control and flow control from the source to destination. The TCP is the most used protocol of transport level which gives connection oriented communication. Another transport layer protocol is the UDP which provides a ununreliable and connectionless communication service.

3.5 Session layer

For some applications service provided by the first three layers (physical, data link and network) is not sufficient then we go foe session layer. It is the network dialog controller, establishes, maintains, and synchronizes the interaction among communicating systems.

3.8 Presentation layer

It deals with the syntax and semantics of the information which is going to exchange between two systems and also responsible for translation, compression, and encryption.

3.9 Application layer

The application layer is closest to end user and enables the user, whether the human or software to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared data base management.

IV. Implementation

The proper block diagram of UDP/IP stack is as shown in the figure 2. The layers, transport, network, data link in UDP/IP stack are designed using Verilog.

4.1 Control transmitter/ receiver

It receives the packet from application and stores it in the RAM transmitter. Control transmitter is for sending data to UDP transmitter block.

In RAM receiver control receiver writes the data in packets from UDP receiver and then sends to the application layer.[2]

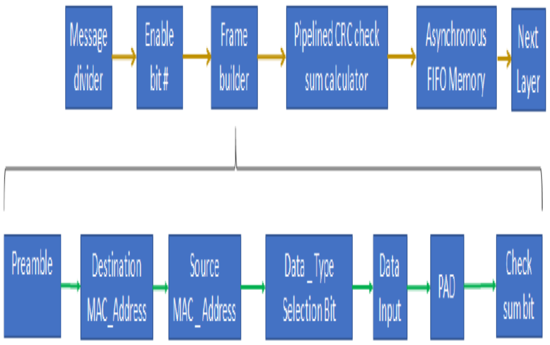


Figure 2 : Proposed Block Diagram of UDP/IP

4.2 UDP transmitter/ receiver

It manages UDP packets and represents transport layer. UDP transmitter encapsulates the packet with the UDP header and sends out to the block IP transmitter. UDP receiver checks the packet and sends it to the control receiver without UDP header information. [2]

4.3 IP transmitter / receiver

IP transmitter represents the network layer and manages IPv4 packets. It calculates checksum and encapsulates packet with IP header. IP receiver verifies the checksum of the packet and the destination IP address. Only IP addresses that matches with the core's IP address and broadcast IP address are accepted and send to UDP receiver and others will get discarded. [2]

4.4 MAC transmitter / receiver

MAC transmitter represents the link layer and manages outgoing and incoming of the packets. It sends packet to the physical layer and preamble, where the last nibble is start of the frame delimiter is sent at the beginning. MAC transmitter sets the control signal high. Each byte is sent to the CRC (cyclic redundancy check) generator which progressively calculates the CRC. When packet end is reached the calculated 32- bit CRC is sent.

MAC receiver checks for new packet, when the new packet is detected it will be sent to the CRC checker which will progressively calculates the CRC checksum.

4.5 ARP

ARP(address resolution protocol) block is used to convert network address to physical address and manages ARP packets. It allows the MAC address from other node when only IP address of its neighbors is known. It uses simple message format that contains one address resolution request.

V. Synthesis & Simulation

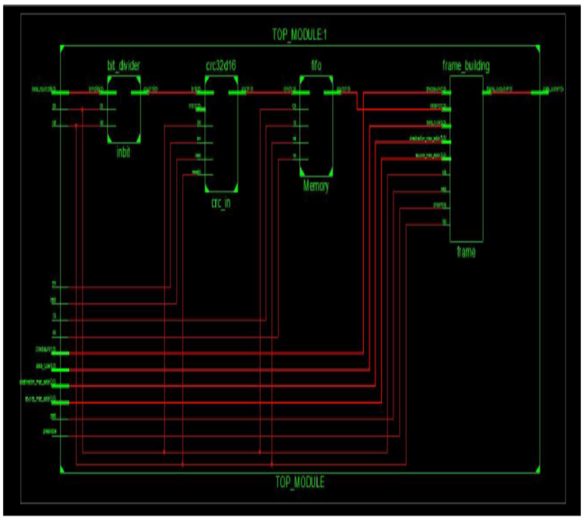


Figure 3: Top Module RTL Schematic Diagram

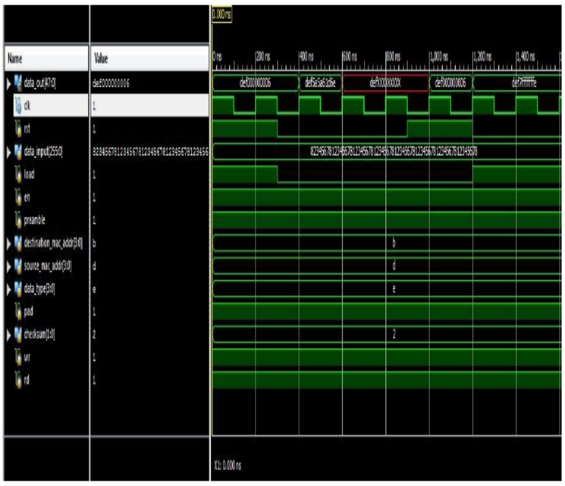


Figure 4:Transmitter Module Simulation Results

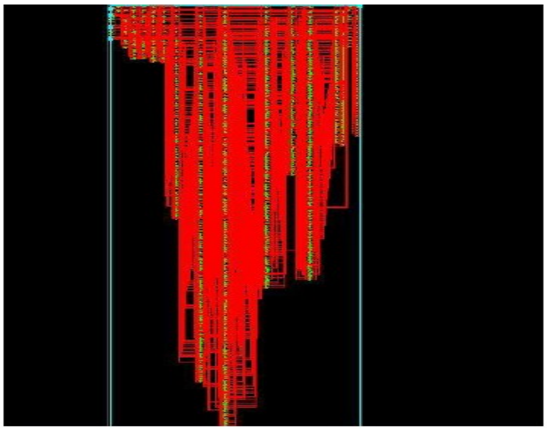
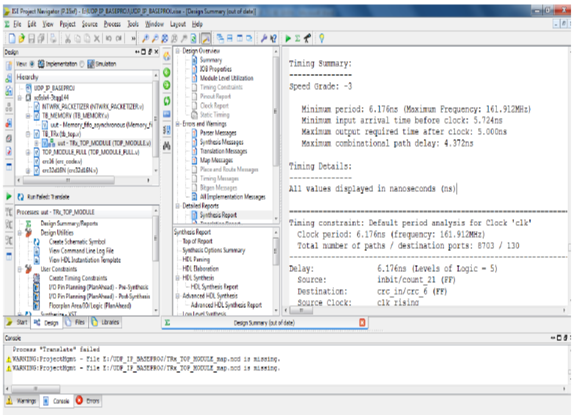


Figure 5 : Transmitter Module Technology Schematic Diagram



```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 6.176ns (Maximum Frequency: 161.912MHz)
   Minimum input arrival time before clock: 5.724ns
   Maximum output required time after clock: 5.000ns
   Maximum combinational path delay: 4.372ns

Timing Details:
---------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default period analysis for Clock 'clk'
  Clock period: 6.176ns (frequency: 161.912MHz)
  Total number of paths / destination ports: 8703 / 130
-------------------------------------------------------------------------
Delay:               6.176ns (Levels of Logic = 5)
  Source:            inbit/count_21 (FF)
  Destination:       crc_in/crc_6 (FF)
  Source Clock:      clk rising
```

```
Process "Translate" failed
WARNING:ProjectMgmt - File E:/UDP_IP_BASEPROJ/TRx_TOP_MODULE_map.ncd is missing.
WARNING:ProjectMgmt - File E:/UDP_IP_BASEPROJ/TRx_TOP_MODULE_map.ncd is missing.
```

Figure 6 : Timing Summary of Transmitter Module

V. Comparative Result

Table 1: Comparative Result of Deferent Parameter

Parameter	Nikolaos Alachiotis, Simon A.Berger[1]	Result Obtained
Slice Registers	184	82
FMax(MHz)	128.8	161.91

REFERENCES:

[1] Nikolaos Alachiotis, Simon A. Berger, Alexandros Stamatakis “A Versatile UDP/IP Based PC$FPGA Communication Platform”, 2016.

[2] Nima Moghaddami Khalilzad Sheida Pourshakour “FPGA implementation of Real-time Ethernet communication

using RMII Interface”, 2013.

[3] Burak Batmaz and Atakan Doğan “UDP/IP Protocol Stack with PCIe Interface on FPGA”, International Conference. Embedded Systems and Applications | ESA’15 |.

[4] Swapnali Sangale, Puja Satibhaski “Implementation of UDP/IP stack in FPGA” International Journal of Advance Engineering and Research Development Volume 2, Issue 7, July -2015 @IJAERD-2015.

[5] Andreas Löfgren Lucas Lodesten Stefan Sjöholm “An Analysis of FPGA-based UDP/IP Stacks Parallelism for Embedded Ethernet Connectivity”, 2014.

[6] Christophoros Kachris “Design and Implementation of a TCP/IP Core for Reconfigurable 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.

* * * * *