Implementation of ATM Packets over MPLS Network on FPGA

Satish M B[1], Savitha C[2], Dr. M Z Kurian[3]
[1]PG Student, Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India
[2]Asst. professor, Dept of ECE Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India
[3]HOD, Dept of ECE, Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India

Abstract— *MPLS is high lightened as the most promising technology for the ATM backbone network. This MPLS improves in reducing the traffic in the network and increases the bandwidth. ATM switch network for the fast Internet services which makes use of virtual network for switching between routers by adding a layer 3 routing module to the existing ATM network and can provide scalable Internet services to users with various service levels. This paper presents an implementation of MPLS for an ATM network on FPGA which replaces the virtual circuits by use of labels in the network.*

Keywords— *MPLS, Asynchronous Transfer Mode, Ingress packet processing module, Egress packet processing module.*

I. INTRODUCTION

The exponential growth of the Internet over the past several years has placed a tremendous strain on the service provider networks. As we all know there has been an increase in the number of users with that there has been a multifold increase in newer applications, backbone traffic, connection speeds. At the start of communication ordinary data applications required only store and forward capability in a best effort manner. Now days there are many newer applications like voice, multimedia and real-time ecommerce applications are pushing toward higher bandwidth and services, irrespective of the dynamic changes or interruptions in the network.

To make the service level guarantees to the service providers not only have to provide large data pipes (Which are also costlier), but also look for architectures which can provide & guarantee QoS (quality-of-service) and optimal performance with minimal increase in the cost of network resources.

IP-based networks typically lack the QoS features available in circuit-based networks, such as ATM and Frame Relay. MPLS brings the sophistication of a connection-oriented protocol to the connectionless IP world. MPLS brings performance enhancements and service creation capabilities to the network. MPLS technology enables Internet Service Providers to offer additional services for their customers, scale their current offering, and exercise more control over their growing networks by using its traffic engineering capabilities.

Below figure shows the MPLS is inserted between layer 2 and layer 3 of the OSI (Open Systems Interconnection).

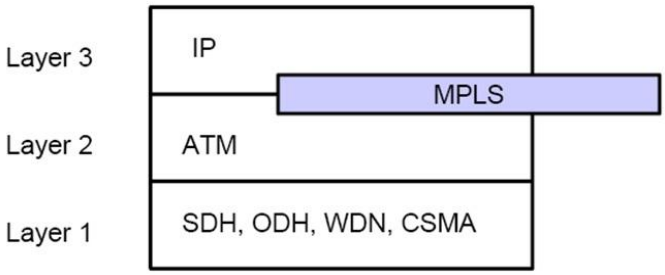

Fig. 1 MPLS B/N layer 2 & layer 3

II. RELATED WORK

MPLS is comprised of different protocols, each performing a different task in the MPLS work flow. Each protocol have been researched differently with respect to hardware implementation. The work in [2] describes hardware implementation of IS-IS protocol. In [3] research on hardware implementation of OSPF protocol is discussed. Research in [4] and [5] discuss the hardware implementation of subset of RSVP-TE and CR-LDP protocols respectively. The work in [7] describes the hardware implementation of reconfigurable MPLS router.

The work in [6] has introduced a hardware processor for the implementation of MPLS using RSVP-TE as its signalling protocol. In [8], an embedded architecture for the MPLS protocol was proposed. The design uses both hardware and software to implement different aspects of MPLS. The architecture proposed implementing routing functionality in software, label switching functionality in hardware.

III. MPLS AND ATM

A. MPLS

IP-based networks typically lack the quality-of-service features available in circuit-based networks, such as ATM and Frame Relay. MPLS replaces the virtual circuits (VC) which reduces the hardware components for connection between routers in the ATM network. MPLS provides an increase in the performance enhancements and service creation capabilities to the network.

MPLS stands for Multiprotocol Label Switching here are some of the terms which are used extensively in MPLS

1. Forwarding Equivalence Class (FEC): a group of IP packets which are forwarded in the same manner (e.g., over the same path, with the same forwarding treatment).

2. MPLS header: The 32-bit MPLS header contains the following fields:

i. The label field (20-bits) carries the actual value of the MPLS label.

ii. The Experimental bits (3-bits) can affect the queuing and discard algorithms applied to the packet as it is transmitted through the network. Since this field has 3 bits, therefore 8 distinct service classes can be maintained.

iii. The Stack field (S) (1-bit) supports a hierarchical label stack. This MPLS supports the processing of a labelled packet is always based on the top label. An unlabeled packet can be thought of as a packet whose label stack is empty (i.e., whose label stack has depth 0). If a packet's label stack is of depth n, we refer to the label at the bottom of the stack as the level 1 label, to the label above it then as the level 2 label if such exit, and to the label at the top of the stack as the level n label. The label stack is used for routing packets through LSP Tunnels.

iv. The TTL (time-to-live) field (8-bits) provides conventional IP TTL functionality.

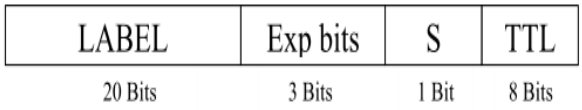

Figure 2 MPLS header format

B. ATM- Asynchronous Transfer Mode

A high performance cell oriented switching and multiplexing technology that utilizes fixed length packets to carry different type of traffic. It contains voice, data and video signals. It was designed for a network that must handle both traditional high-throughput data traffic (e.g., file transfers), and low-latency content, real-time such as voice and video.

ATM is a core protocol mainly used over the SONET/SDH backbone of the public switched telephone network (PSTN) and Integrated Services Digital Network (ISDN).

Structure of an ATM cell

An ATM cell is of 53 bytes which consist of 5-byte header and a 48-byte payload.

GFC = Generic Flow Control (4 bits)
(Default: 4-zero Bits)
VPI = Virtual Path Identifier (8 bits UNI) or
(12 bits NNI)
VCI = Virtual Channel identifier (16 bits)
PT = Payload Type (3 bits)
CLP = Cell Loss Priority (1-bit)
HEC = Header Error Control

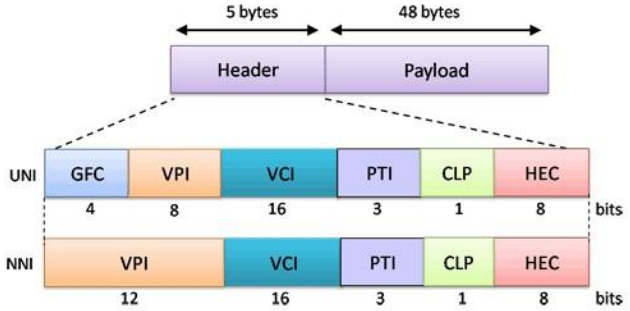

Figure 3 ATM Cell

IV. ARCHITECTURE FOR MPLS

The MPLS packet processing includes label lookups, packet forwarding, label manipulation and routing protocol functionality. Below figure illustrates a high level description of MPLS architecture

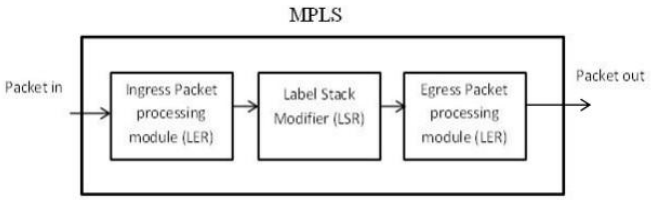

Figure 4 MPLS Architecture

The architecture consists of two packet processing modules, and a separate module to modify the label stack. Ingress packet processing module work as LER (Label Edge Router), it will process the packet and attach the label to the packet, and the label stack modifier as a LSR (Label Switch Router) will replace the label with new label. The packet with the new label will transfer to the next router that is LER, the egress packet processing module is the last router before the destination will process the packet and then it will remove the label from the packet and transfer packet with no label to the destination. Egress LER has not an outgoing label.

IV. DESIGN

A. Ingress Packet Processing Module

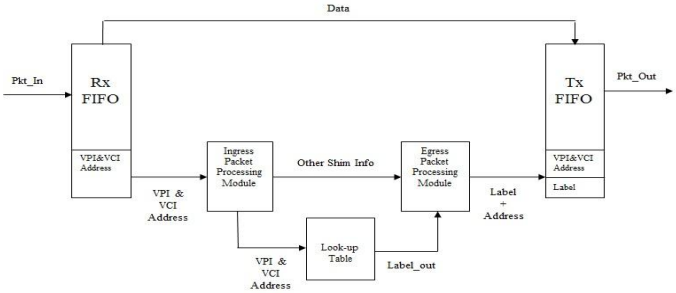

Figure 5 Label Edge Router

As in the figure 5 shows the whole ATM cell is transmitted to the receiver FIFO, this receiver FIFO will help to store whole cell, from this ingress packet processing module will access only the VPI and VCI address data, this data is transmitted to the label look up table where the corresponding label will be taken out according to the VPI & VCI address. This label will be sent to the Egress packet processing module where this label is added to the cell, even the VPI & VCI address also sent to the transmitter FIFO. So total ATM cell plus MPLS label is of 57 bytes is stored into the transmitter FIFO. This stored data is transmitted to the LSR.

B. Label Stack Modifier (LSR)

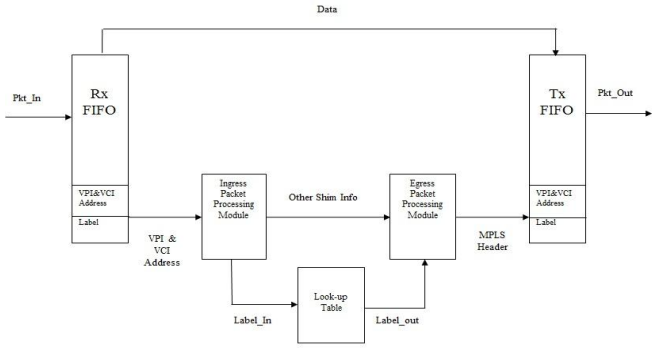

Figure 6 Label Switch Router

The above figure shows the Label Switch Router where the label will be accessed and this label will be switched to another label which indicates the next router label. The whole 57 bytes of ATM cell is accessed by the receiver FIFO, from this only the Label will be taken and sent to the label look up table. Here the corresponding label will be taken out and attached to the ATM cell and sent to transmitter FIFO.

C. Egress Processing Module (LER)

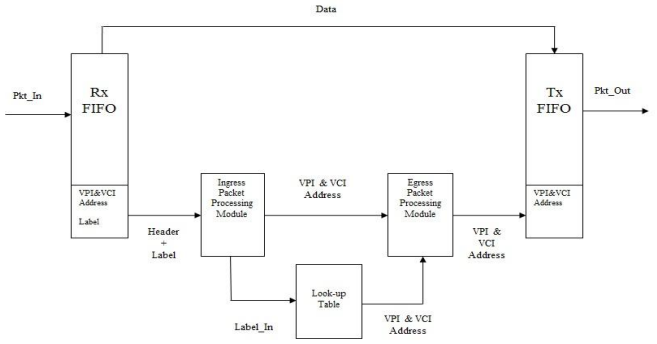

Figure 7 Label Edge Router at Destination

When the ATM cell with MPLS reaches the last node or router before the destination, in that LER the label will be checked in the label look up table and corresponding VPI & VCI address is taken out and compared with the original VPI & VCI address. So finally in the in last router only the data and VPI & VCI address will be there and this sent to the destination.

V. IMPLEMENTATION

The MPLS architecture was implemented using Verilog hardware description language. These descriptions were then processed by standard Xilinx ISE 10.1 design tool suite, which performed synthesis, placement, routing, and bitstream (FPGA physical programming information) generation. The bitstream generated was dumped onto XC2VP30 device of Xilinx Virtex 2 pro family.

Below figure 7 shows the FPGA implemented result which seen on chip scope. It shows the packet in into the LER, here label will be added to the packet and sent to the LSR. In the implementation we have been used two LSRs so in the first LSR show the output with a label attached to the packet. Finally when it reaches the destination LER then the label will be removed and send the ATM packet to the final destination.

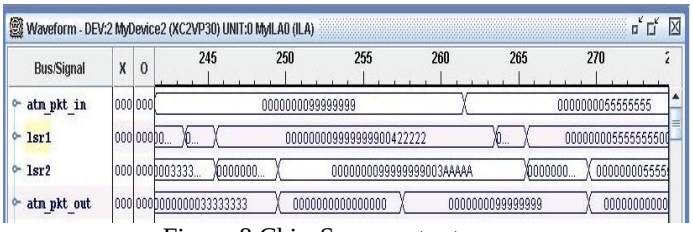

Figure 8 Chip-Scope output

CONCLUSION

ATM network is a connection oriented which gives better quality-of-service features available in circuit-based networks. MPLS over ATM network replaces the VC by inserting a label in each packet and forwards the packet with label throughout network which performs as connection-oriented protocol from this it maintains the QOS. Based on simple improvements in basic IP routing, MPLS increases performance enhancements and service creation capabilities to the network. So ATM with MPLS usage into the enterprise can meet emerging requirements for scalable transport for end user services, including data, voice and video. In this paper, MPLS hardware architecture for Ethernet packets was proposed. This prototype implementation of MPLS on FPGA hardware has demonstrated the potential for 100x speedup in packet processing and forwarding when compared to software implementations.

REFERENCES

[1] E. Rosen, A. Viswanathan, R. *Callon* "RFC 3031:Multiprotocol Label Switching Architecture", January 2001.

[2] M. Abou-Gabal, R. Peterkin, D. Ionescu: "IS-IS protocol Hardware Architecture for VPN solutions", in Proceedings of the 7th WSEAS International Conference on Communications, Athens, Greece, July 12-15, 2004.

[3] M. Abou-Gabal, R. Peterkin, D. Ionescu "An Architecture for a Hardware Implementation of the OSPF Protocol", CAINE 2004 - 17th International Conference on Computer Applications in Industry and Engineering, Orlando, Florida, USA, November 17-19, 2004.

[4] H. Wang, M. Veeraraghavan, R. Karri, T. Li, "A Hardware-Accelerated Implementation of the RSVP-TE Signaling Protocol", 2004 IEEE International Conference on Communications, Volume 3, 20-24 June 2004 Page(s):1609 – 1614.

[5] T. Li, Z. Tao, H. Wang, M. Veeraraghavan, "Specification of a Subset of CR-LDP for Hardware Implementation", January 2005.

[6] Raymond Peterkin, " A Reconfigurable Hardware Architecture for VPN MPLS based Services" University of Saskatchewan Electrical Engineering Master Thesis, August 2007.

[8] Peterkin.R, ionescu.D, "Embedded MPLS Architecture" Parallel and Distributed processing symposium, IEEE 2005.

[9] Peterkin.R, ionescu.D, "A Hardware/Software Co-Design for RSVP-TE MPLS" Parallel and Distributed processing symposium, IEEE 2005.

[7] S. Li, "System Architecture and Hardware Implementations for a Reconfigurable MPLS Router" University of Saskatchewan Electrical Engineering Master Thesis, August 2003.