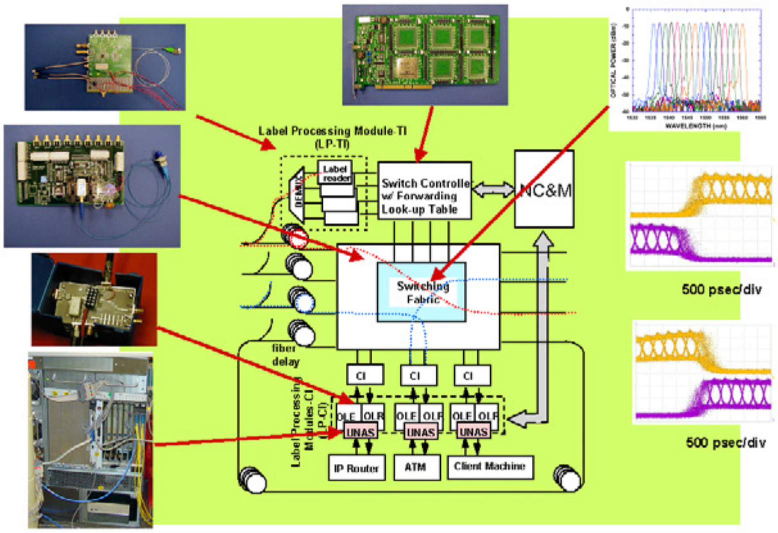


Fig. 2. A schematic of an optical-label switching core router with edge routers at the client layer integrated by UC Davis (the UNAS part of the edge router is courtesy of Kirk Boyer et al at Tektronix, Inc, Oregon).

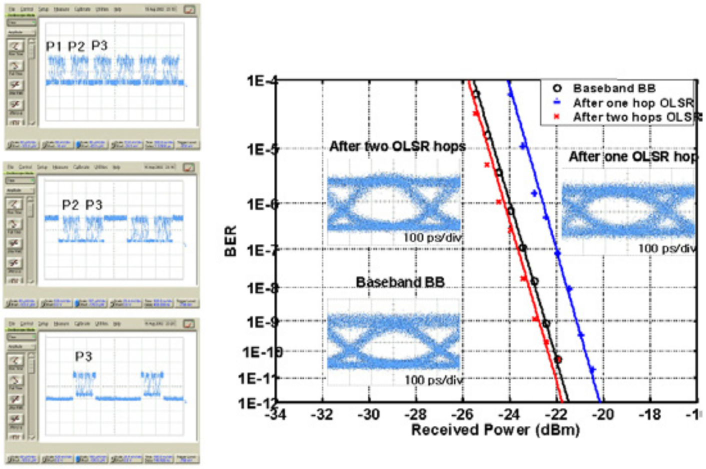
P1, P2, and P3 to be forwarded to relevant directions or destinations. At each hop, an all-optical label swapping takes place with the new label content on each packet generated by the OLSR controller. This controller, as mentioned in section 2 also conducts contention resolution in wavelength, time, and space domains with a switching speed of approximately 1 nsec. Based on the forwarding decision, the switch controller sends a control signal to the tunable wavelength conversion to switch to the designated wavelength within the switching fabric, which in turn triggers switching in the wavelength, time, and space domain switching based on the optical switching fabric architecture. Packet by packet bit-error-rate measurements took place on the P3 at each hop. Fig. 3 (a) shows the packet patterns at the ingress (top trace), at the first hop (middle trace), and at the second hop (bottom trace), which clearly demonstrates the desired optical-label based packet dropping. Fig. 3(b) shows packet BER measured on P3. A negative power penalty after 2 hop OLSR is mainly due to the 2R regeneration and the decrease of received average power after two packet droppings. The eye diagrams of the switched payload are shown as insets in Fig. 3 showing clear openings.

4. Network Evolution

The viability of optical-packet switching, the current prevalence of circuit switching, the popularity of MPLS and MPLambdaS will all necessitate the flexible interoperability of optical-label switching networks. The edge router will not only provide important packet aggregation points for improved end-to-end performance, but it will also function as an interface to generate labels or to set up label switched paths. Via the proper edge router functions, we envision that the current IP overlay on ATM architecture will evolve to IP overlay on MPLS, and to IP over MPLambdaS, and eventually IP over Optical-Label Switching/WDM. The current optical-label switching can directly interoperate with MPLS, IP, and MPLambdaS through the GMPLS, however, a new GMPLS extension can support unique optical-label switching capability to achieve packet switching directly in the optical layer.

5. Summary

The optical-packet switching and optical-label switching will play key roles in the future optical Internet. The seamless integration of data and optical networking with full interoperability can be achieved by optical-label switching. Recent progress achieved by a number of research groups including demonstration of cascaded OLS routers, edge routers, and the field network trial show promising future for this technology.



(a) (b)

Fig. 3. (a). Scope traces showing (top) the incoming P1, P2 and P3, (middle) P2 and P3 sent to the second hop after dropping P1, and (bottom) P3 at the final destination output after two-hop OLSR. (b) BER test results of the cascaded OLSR (Insets: eye diagrams of the baseband payload and signals after OLSR)

6. References

[1] DARPA BAA 98-02 Next generation internet, http://www.darpa.mil/ipto/Solicitations/CBD_9802.html

[2] S.J.B. Yoo, G.K. Chang, *High-Throughput, Low-Latency Next Generation Internet Using Optical Tag Switching*, in *U. S. Patent 6,111,673*. 1997, Bellcore: U. S. A.

[3] B. Meagher, et. al., “Design and implementation of ultra-low latency optical label switching for packet-switched WDM networks,” Journal of Lightwave Technology, vol.**18**, (no.12), IEEE, Dec. 2000. p.1978-87.

[4] S. Yao, et al, “Electrical Ingress Buffering and Traffic Aggregation for Optical Packet Switching and Its Effect on TCP Performance,” in IEEE Commun. Mag., Nov. 2002.

[5] Chang, G.K, et. al., “A proof-of-concept, ultra-low latency optical label switching testbed demonstration for next generation Internet networks.,” Optical Fiber Communication Conference, Baltimore, MD (2000).

[6] S.J.B. Yoo, et. al., “Optical-label based packet routing system with contention resolution in wavelength, time and space domains,” Optical Fiber Communication Conference, Anaheim, California, paper WO2 (2002)

[7] Y. M. Lin, et al, “A novel optical label swapping technique using erasable optical single-sideband subcarrier label”, IEEE Photon. Tech. Lett. 12, pp.1088-1090 (2000).

[8] B. E. Ollsson, et al, “Wavelength routing of 40 Gbit/s packets with 2.5 Gbit/s header erasure/rewriting using all-fiber wavelength converter”, IEE Electron. Lett. **36**, pp. 345-347 (2000).

[9] Hyuek Jae Lee, et. al., “A simple all-optical label detection and swapping technique incorporating a fiber Bragg grating filter,” IEEE Photonics Technology Letters, vol. **13**, (no.6), IEEE, June 2001, p. 635-7

FS6 **12:00 PM**

Optical Time-to-Live Decrementing and Subsequent Dropping of an Optical Packet

J. McGeehan, S. Kumar, A. Willner, *University of Southern California, Los Angeles, CA;* J. Bannister, J. Touch, *USC Information Sciences Institute, Marina del Rey, CA, Email: mcgeehan@usc.edu.*

An optical time-to-live (TTL) decrementing module for optical packet-switched networks is demonstrated. Our module acts on an NRZ-modulated binary TTL field and decrements it, drops a packet if its TTL is zero, and is independent of the TTL length.

1. Introduction

All-optical packet switching remains a laudable goal for efficient and high-throughput networking. One problem in many packet-switched networks is “routing loops”, where misdirected or mislabeled packets are routed in circles, never reaching their destination and leading to severe network congestion [1]. Such loops are often prevented by a “time-to-live” (TTL) field within a packet header. The TTL field determines the maximum number of hops that a packet can take before getting dropped for being too “old” for the network. In general, the TTL is a binary number (commonly 8 bits long) that is decremented by 1 when traversing a switching node. This requires editing the packet header, something easy in electronics but often difficult in optical systems.

There has been scant research reported on all-optical technologies for manipulating the TTL field or for dropping packets that have a zero TTL, particularly for NRZ systems. Specifically, there has been one report of using a discrete series of ultrashort RZ optical pulses as a “TTL burst”, where instead of a TTL field in a packet header the number of optical pulses corresponded to the TTL value. A single pass through the TTL module extinguished a single pulse, and when no pulses remained, the packet was dropped [2]. However, an optical embodiment of true modification of a TTL field similar to that in modern

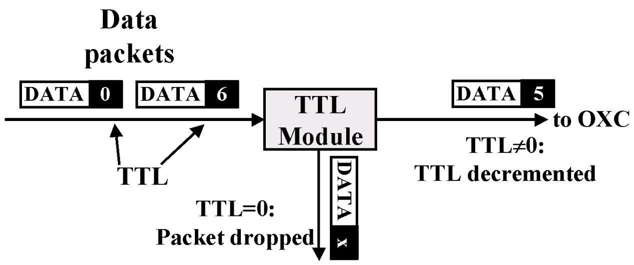


Fig. 1(a). Conceptual diagram of the optical TTL module.

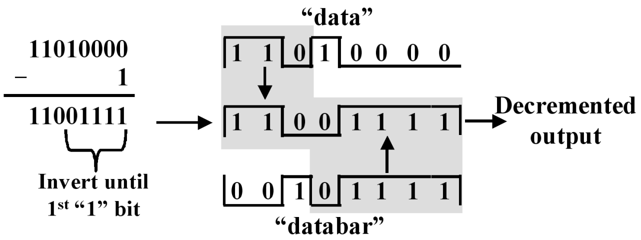


Fig. 1(b). Binary subtraction via insertion of conjugate data.

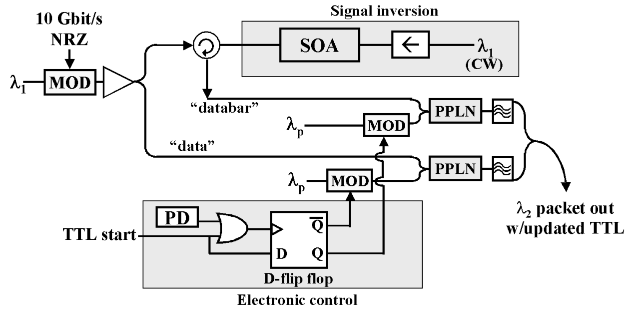


Fig. 1(c). Experimental setup of the TTL module. Signal inversion is accomplished via gain saturation in an SOA. A D-flip-flop controls the insertion of conjugate data by controlling the PPLN waveguide pumps.

network packets has not been reported.

We demonstrate an optical TTL decrementing module that acts upon a standard NRZ-modulated 8-bit binary TTL field and uses gain saturation in a semiconductor-optical-amplifier (SOA) and difference-frequency-generation (DFG) in a periodically-poled lithium-niobate (PPLN) waveguide to perform the TTL decrementing function. If the TTL value on entering the module is "0", the packet is dropped from the network, else the TTL field is decremented by one and passes through. Our technique is independent of the TTL length, does not require the use of ultrashort optical pulses, requires no guard time between the end of the TTL field and the packet data, has >10 dB extinction ratio, works for both NRZ and RZ data, has only a 2.4 dB power penalty, and using a decision circuit can generate a signal that drops a zero TTL packet.

2. Optical TTL Setup and Operation

A conceptual diagram of our optical TTL module is shown in Fig. 1(a). Packets with TTL fields within the header (not necessarily at the start of the header, but contained within the header with a 1-bit guard time before the TTL to ensure proper timing) enter a switching node and the TTL module. If the TTL value of an incoming packet is nonzero, the module decrements the TTL by one and the packet continues through the switching node. If the TTL of the incoming packet is zero, the packet is dropped.

The binary subtraction process occurs as follows: for any arbitrary starting binary value (for example, 11010000), subtracting "1" results in each successive bit (from the least-significant-bit to the most-significant-bit (LSB-to-MSB)) being inverted (due to the "borrowing" that takes place during subtraction) until such time as a "1" bit is encountered. As there is no need to "borrow" when subtracting from a "1" bit, no further bits are inverted. Thus the algorithm is simple – once the TTL begins, invert each bit, starting with the LSB, until a "1" bit is encountered. Invert the "1" bit, then stop. This inversion can be done by replacing the data with its conjugate, stopping after the first "1" bit, as shown in Fig. 1(b).

The experimental setup of our optical TTL module is shown in Fig. 1(c). An incoming data packet, with an 8-bit TTL field, is amplified and split into three branches. The middle branch is the "data" sent to the "data" PPLN waveguide. The top branch is transmitted through a 3-port optical circulator and then through a semiconductor optical amplifier (SOA) to create an inverted copy of the data stream ("databar") via gain saturation [3] with an extinction ratio >10 dB. The bottom branch of the input data stream is sent to a receiver and the resulting signal is used (after an OR gate) as the clock input to a D-flip-flop that selects which of the data streams ("data" or "databar") is used at any given moment by controlling the PPLN waveguide pumps. Using a cascaded $\chi^{(2)}:\chi^{(2)}$ process, the PPLN waveguides efficiently perform DFG and λ-shift an input signal to a new wavelength [4]. By modulating the PPLN waveguide pumps appropriately, we control which of the two PPLN waveguides will be λ-shifting its input signal. The key to the system is the use of the single D-flip-flop, where the "Q" output controls the "databar" pump modulator and the "Qbar" output controls the "data" pump modulator, guaranteeing that only one of the two data streams will be λ-shifted at any given time. The optical spectrum showing the input, pump, and DFG-generated λ-shifted output is shown in Fig. 1(d). We shift "databar" to λ_2 only when the TTL-modification algorithm requires the conjugate data at the output, and shift "data" all other times.

A synchronization pulse (perhaps provided by some sort of optical preamble detection such as in [5]) signals to the electronics the start of the TTL field and sets the "Q" output of the flip-flop to "1", resulting in "databar" being λ-shifted by the first PPLN waveguide, and shuts off the pump to the "data" PPLN. The first "1" bit in the incoming packet data acts as a "clock" input to the flip-flop, resetting the "Q" output to zero, which shuts off the pump to the "databar" PPLN, beginning λ-shifting of the "data" stream to λ_2. After filtering out everything save λ_2, the two synchronized λ-shifted signals are combined by a coupler and the result is a TTL-modified λ_2 output. The "Q" signal can also be used to control a switch that drops a packet when the TTL is zero. As any "1" bit in the TTL field will cause the "Q" output to reset to "0", a decision circuit can test "Q" immediately after the TTL field has passed. Only if all TTL bits are zero will the "Q" output be high after the TTL has passed. This decision output can be used to reset the flip-flop and drive an optical switch (or other module) that drops the packet. In addition, as any packet with a nonzero TTL will stop shifting "databar" and begin shifting "data" by the end of the TTL field, no guard time between the end of the TTL and the rest of the packet is required.

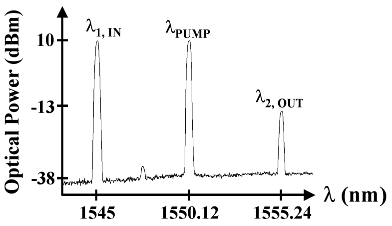


Fig. 1(d). Optical spectrum at the output of the PPLN waveguides.

3. Optical TTL Decrementing and Packet Dropping

Three 20-byte packets, each with varying TTL values and data and .8 ns of guard time between packets, were NRZ modulated at 10 Gbit/s (by replacing the CW SOA pump with a series of RZ "1" bits, RZ data can be used as well) onto a 1545 nm signal wavelength and sent into the TTL module. Two of the three packets had non-zero TTL values. A closer look at the TTL field of one of the packets is shown in Fig. 3 for a packet with a TTL of "00111100" (LSB-to-MSB). The input "data" signal is shown in Fig. 2(a), and the "databar" stream generated via SOA gain saturation is shown in Fig. 2(b). The SOA bias current was 80 mA. The PPLN pump modulation signals (the outputs of the D-flip-flop) are shown in Figs. 2(c) and (d), where 2(c) is the signal modulating the "data" PPLN pump, and 2(d) is the signal modulating the "databar" PPLN pump. The "databar" pump is high only for the first three bits of the TTL (up to and including the first "1" bit) and thus the "databar" stream is only λ-shifted during that time period. At all other times, the "data" stream is λ-shifted. The pump wavelength for both PPLNs was 1550.12 nm, and the pump power into each PPLN was approximately 10 dBm, resulting in an output λ-shifted signal at 1555.24 nm at around –13 dBm. After amplification and filtering, the resulting streams were combined and the TTL-modified result is shown in Fig. 2(e).

The total power penalty of the module, shown in Fig. 2(f), is ~2.4 dB when compared to the back-to-back receiver sensitivity at 10^{-9} bit-error-rate, caused mostly by the λ-shifting of the PPLN and insertion of parts of "databar" into the TTL field. The use of a higher-efficiency PPLN and an SOA with lower chirp may reduce these numbers considerably.

To demonstrate packet dropping, an optical switch was placed in series with the TTL module and controlled by a decision circuit as described above. The "Q" output is sampled on the 1st bit

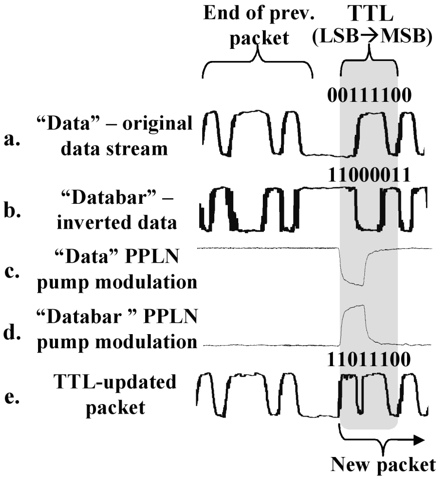
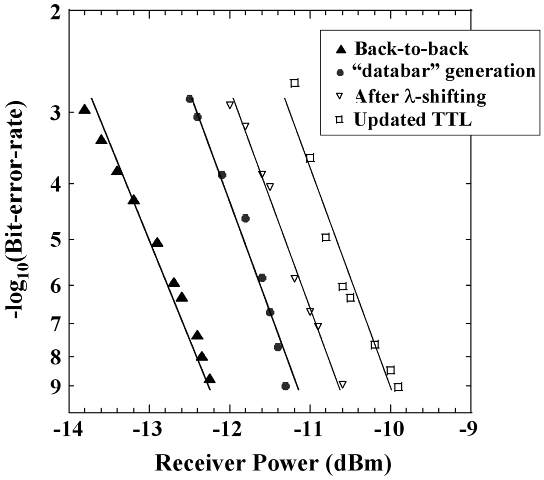


Fig. 2. (a) A close-up look at the TTL field of a packet. The TTL value is "00111100" (LSB-to-MSB). (b). Conjugate data generated by the SOA. (c). The "data" PPLN pump modulation signal. (d). The "databar" PPLN modulation signal. (e). The TTL-modified output , with a new TTL of "11011100".



(f) Power penalty curves for the optical TTL module. The total penalty is ~2.4 dB.

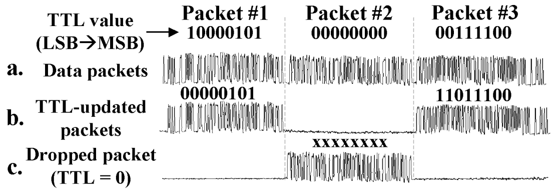


Fig. 3. (a). Packets with different TTL values and data before entering the TTL module. (b). Nonzero-TTL-packets pass through the module and are decremented. (c). Zero-TTL-packets are dropped and the resulting TTL value is irrelevant.

after the TTL – if it is high, it signals the switch to drop the packet and resets the flip-flop so that the "Q" signal is again "0". The three packets are shown in Fig. 3(a), and the switch outputs are shown as Figs. 3(b) and (c), where 3(b) shows the "through" port of the switch and 3(c) the "drop" port.

4. References

[1] W. Zaumen et al,*Internetworking: Res. Exper.*,vol.3,pp.161-188,1992.
[2] W. Hung et al,OFC 2002,paper ThGG111,2002.
[3] M. C. Cardakli et al,OFC 2000,paper TuN2,2000.
[4] K. R. Parameswaran et al,LEOS 2001,paper ThL3,2001.
[5] M. C. Cardakli et al,*PTL*,vol.14,no.9,2002.

FS7 **12:15 PM**

40 Gbit/s Interface, Optical Code Based Photonic Packet Switch Prototype

N. Wada, H. Harai, F. Kubota, *Communications Research Laboratory, Tokyo, Japan, Email: wada@crl.go.jp*.

A 40Gbit/s interface, optical code based photonic packet switch prototype is developed for the first time. Photonic packet swtching with 200Gchip/s all-optical label processing, 40Gbit/s/port packet switching, and optical buffering to avoid packet collision is experimentally demonstrated.

1. Introduction

It is necessary to improve the node throughput as well as the link capacity, in order to build a backbone network capable of handling tremendous amount of traffic in the Internet. While the link capacity is easily increased by bundling optical fibers, the electronic processing in the current node systems may limit the increase in the node throughput. As it is necessary to increase the interface (i.e., port) speed and the number of ports at nodes, the electrical limitations motivate us to introduce optical technology into packet forwarding.

In spite of immatureness of optical technology, some packet switching systems have been developed [1-3]. The functions of packet switches are roughly divided into five groups of functions: label processing, switching, scheduling, buffering, and routing functions (See Fig. 1). In [1-3], although their payload's street such as switch and buffer is an optical one, these systems rely on electronic processing for label processing, in which a processor recognizes a label of an optical packet and determines the designated port for the packet. The electronic label processing does not affect optical transparent transmission. However, since the electronic label processing requires memory access, it limits optical potential to switch extremely large amount of traffic. Packet switching more than 10Gbit/s/port may be difficult while 10Gbit/s/port electronic packet switching is commercially available. We thus developed a photonic bufferless packet switching systems with optical label processing (i.e. optical bipolar code [4] and multi-wavelength code [5] based label switching) in which label processing is performed optically. We also investigated electronic scheduling and optical buffering. System performance of the proposed optical buffering were confirmed by simulation [6].

In this paper, we develop a 40Gbit/s interface, optical code based photonic packet switch prototype for the first time. The prototype includes optical label processing based on Planner Lightwave Circuit (PLC), optical switching based on semiconductor optical amplifier (SOA) gate switches, electronic scheduling based on Field Programmable Gate Array (FPGA), and optical buffering based on $LiNbO_3$ intensity modulators and optical fiber delay lines (FDL). We experimentally demonstrate photonic packet switching with 200Gchip/s all-optical label processing, 40Gbit/s/port packet switching, and optical buffering to avoid packet collisions.

2. Photonic packet switch with optical label processing and buffering

Figure 1 shows our concept of functional partitioning in a photonic packet switch with optical label processing. As in this figure, functions in photonic packet switch are roughly into five: routing, label processing, switching, scheduling, and buffering. Developed prototype has these functions except the routing.

An optical packet consists of a header and a data as shown in Fig. 1. The header has an optical code-based label of a destination node. In the photonic packet switch, an optical packet is simply divided to two arms: header and payload. A header goes to label processing part. A payload goes through the optical switching and buffering parts. In the label processing part, an optical label processor analyzes a packet header optically. All-optical label analyzing method is represented in Fig. 2. In an optical label processor, a set of optical correlators works as a label bank, which stores optical codes correspond to the destination addresses in the routing table. Label recognition is based on parallel optical correlation in the time domain between an input optical code and the codes in the label bank. The correlator decodes the input code of each packet header and outputs the signal which has high or low value at matched and unmatched cases, respectively. Label processor controls the optical switch and gives packet arriving information to the scheduler. After packets are switched to designated directions, an electrical scheduler controls optical buffers to avoid collision of the packets. By separating the individual functions of a packet switch, we can fully use optical forwarding capability. The photonic packet switch is composed of N '1xN switches' and N 'Nx1 buffers', where N is the number of ports. The 1xN switch consists of label processing and switching parts while the Nx1 buffer consists of scheduling and buffering parts. The prototype has 2 ports. Each switch and buffer pair is connected in a mesh manner. We assume that packets arrive synchronously in order to simplify buffer