Multi-Terabit/s IP Switching with Guaranteed Service for Streaming Traffic

Mario Baldi
Politecnico di Torino

Yoram Ofek
Università di Trento

Abstract — **As traffic on the Internet continues to grow exponentially, there is a real need to solve transmission and switching scalability. Moreover, future Internet traffic will be dominated by streaming media flows, such as video-telephony, video-conferencing, 3D video, virtual reality, and many more. Consequently, network solutions will need to offer quality of service and traffic engineering together with the abovementioned scalability — i.e., over-provisioning is not likely be a viable solution to accommodate streaming media traffic. This paper describes the architecture of a ultra-scalable IP switch and the first experiments with a prototypal implementation. The switch scalability is a consequence of it operating pipeline forwarding of packets, which also results in quality of service guarantees for UDP-based streaming applications, while preserving elastic TCP-based traffic as is, i.e., without affecting any existing applications based on "best-effort" services. Moreover, the prototype demonstrates the low complexity of pipeline forwarding implementation as the deployed network gear was realized from off-the-shelf components in only nine months through the design, implementation, and testing efforts of the authors.**

I. The Problem

The steady Internet growth over the past few years is impressive, but services so far deployed over the Internet are nothing compared to the ones that can still be deployed. One likely scenario is that the future Internet will be dominated by applications such as (3D) video on demand, high quality videoconferencing, distributed gaming, (3D) virtual reality, remote surveillance, and many more. These applications generate traffic that is either by nature streaming or can be effectively handled as such (e.g., large file transfers). Moreover, most of these applications need a minimum guaranteed quality in order to be usable. Consequently, there is a real need to solve scalability *and* traffic engineering simultaneously — specifically, without using over-provisioning in order to provide predictable service.

Concerning scalability, it is interesting noting that Cisco's top-of-the-line router, CRS-1, has a per chassis switching capacity of 640 Gb/s (the announcement of 92 Tb/s is to be divided by 2, to avoid twice packets first entering and exiting the switch, and then by 72 chassis's), which represents an improvement over the Cisco 12000 by a factor of only 2 after 5 years of development — not the 18 months during which the Internet traffic doubles.

This paper shows how the Internet can benefit from UTC-based pipeline forwarding of IP packets that enables (*i*) ultra-scalable IP switches – 10-50 Tb/s in a single chassis, (*ii*) quality of service (QoS) for UDP-based streaming applications (as a bonus since a deterministic service is inherent to the switching solution itself), while (*iii*) preserving elastic TCP-based best-effort traffic as is. Notice that no change can be seen when observing a link: standard (whole) IP packets encapsulated into Ethernet or PPP frames transit.

II. UTC-Based Pipeline Forwarding

A. *Basic Operating Principles*

Implementing UTC-based pipeline forwarding for real-time packet scheduling requires IP packet switches to be synchronized with a *common time reference* (CTR). *UTC* (coordinated universal time) offers a CTR that is globally available through various time-distribution systems such as the global positioning system (GPS) and, in the future, Galileo. An extensive and detailed description of UTC-based forwarding is outside the scope of this paper and is available in [1].

Synchronized IP packet switches use a basic time period called time frame (TF) whose duration T_f is derived, for example, from the UTC second. Time frames are grouped into time cycles (TCs) and TCs are further organized into super cycles, each of which typically lasts one UTC second. The transmission capacity during each TF is partially or completely reserved to one or more flows. The TC and the super cycle provide the basis for periodic repetitions of the reservation.

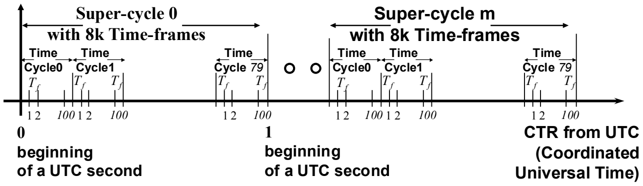


Fig. 1. Common time reference structure

For example, in Fig. 1, the 125-μs time frame T_f is obtained by dividing the UTC second by 8000; sequences of 100 TFs are grouped into one TC, and runs of 80 TCs are comprised in one super cycle (i.e., one UTC second).

The periodic scheduling within each node results in a *periodic packet forwarding* across the network, which is also referred to as *pipeline forwarding* for the ordered, step-by-step fashion, with which packets travel deterministically (see Fig. 2). The periodic scheduling fits particularly well to periodic (e.g., streaming) traffic, but it can be beneficial in various other contexts, such as large file transfers. UTC-based forwarding guarantees that reserved traffic

This work was supported by funds from the European Commission under the IP-FLOW contract No MC-EXC 002807 and partly under the E-Next Project FP6-506869.

experiences: (*i*) bounded end-to-end delay, (*ii*) delay jitter lower than one TF, and (*iii*) no congestion and no resulting loss. These properties are ensured also when multicasting is implemented, as described in [6].

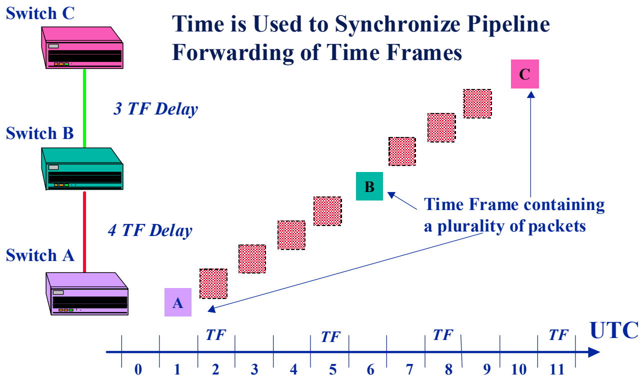

Switch C
Time is Used to Synchronize Pipeline Forwarding of Time Frames
C
3 TF Delay
Switch B
B
Time Frame containing a plurality of packets
4 TF Delay
Switch A
A
TF
TF
TF
TF
UTC
0
1
2
3
4
5
6
7
8
9
10
11

Fig. 2: UTC-based pipeline forwarding

Moreover, it is worth highlighting that TFs are virtual containers; consequently, packets can extend beyond the time boundaries of a TF. In order for pipeline forwarding to operate properly and ensure the above properties, network nodes must deploy means to uniquely identify the TF a packet logically belongs to, such as either TF delimiters or time stamps or measuring the transmission time of the first bit of a packet.

In pipeline forwarding, *a synchronous virtual pipe (SVP)* is a predefined schedule for forwarding a pre-allocated amount of bytes during one or more TFs along a path of subsequent UTC-based switches. A hierarchical resource reservation model can be used to set-up SVPs, which enables multiple component SVPs to be aggregated in larger, possibly pre-provisioned, SVPs in the core of the network. This results in scalability comparable to the DiffServ model, while ensuring guaranteed quality of service to single component SVPs.

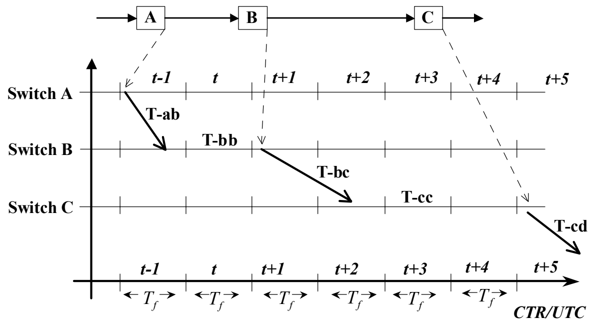


Fig. 3. Pipeline Forwarding Operation

As exemplified in Fig. 3, which depicts the journey of an IP packet from node A to node D along three UTC-based switches, the forwarding delay may have different values for different nodes, due to different propagation delays on different links (e.g., T_{ab}, T_{bc}, and T_{cd}), and different packet processing times in heterogeneous nodes (e.g., T_{bb} and T_{cc}). Moreover, two variants of the basic pipeline forwarding operation are possible. When node *n* deploys *immediate forwarding*, the forwarding delay has the same value for all the packets transmitted by node *n*. When implementing *non-immediate forwarding*, node *n* may use different forwarding delays for packets belonging to different flows.

The UTC accuracy required to implement pipeline forwarding can be relaxed by using two means in the design of routers and protocols: (*i*) time frame delimiters and (*ii*) time stamps. The reason for such a relaxed requirement is that the UTC is not used for detecting the time frame boundaries, as they are detected by means of the delimiters. For example, time frame delimiters enable correct mapping of incoming packets to the input channel time frames, i.e., the UTC time frames in which they were transmitted by the upstream node. When a time frame delimiter exists, the correct mapping of TFs can be maintained with a UTC accuracy of ½·$\boldsymbol{T_f}$ — i.e., UTC±1/2·(10µs to 100µs) — without compromising on performance and complexity. Additional mechanisms and buffering can be used to enable an even sloppier synchronization to UTC. Deployment of time stamps (e.g., including in packets the TF in which they were transmitted) enables higher fault and loss tolerant solutions. However, tradeoffs between UTC accuracy, system complexity, delay, and robustness are outside the scope of this paper and will be the subject of further work.

Today, time cards with 1 pps (pulse per second) UTC with accuracy of 10-20 ns are available from various vendors. These cards are small and cost around $100 each. By combining such time cards with Rubidium or Cesium clocks it is possible to have a UTC reference within the required accuracy for days (with Rubidium) and months (with Cesium) in the event the GPS signal is lost.

Two implementations of the pipeline forwarding were proposed thus far: Time-Driven Switching (TDS) and Time-Driven Priority (TDP) [2]. This paper focuses on TDS as it enables the implementation of highly scalable switching architectures.

B. Time-Driven Switching

TDS was proposed to realize sub-lambda or fractional lambda switching (FλS) in highly scalable dynamic optical networking [1][6], which requires minimum optical buffers. In this context, TDS has the same general objectives as optical burst switching and optical packet switching: realizing all-optical networks with high wavelength utilization. TFs can be viewed as virtual containers for multiple IP packets that are switched at every TDS switch based on and coordinated by the UTC signal.

In TDS all packets in the same TF are switched the same way. Consequently, header processing is not required, which results in low complexity (hence high scalability) and enables optical implementation. The allocation granularity depends on the number of TFs per TC allocated to each flow. For example, with a 10 Gb/s optical channel and 1000 TFs in each TC, the minimum capacity (obtained by allocating one TF in every TC) is 10 Mb/s.

Scheduling through a switching fabric is based on a pre-defined schedule, which enables the implementation of a simple controller. Moreover, low-complexity switching fabric architectures, such as Banyan, can be deployed notwithstanding their blocking features, thus further enhancing scalability. In fact, blocking can be avoided during schedule computation by avoiding conflicting input/output connections during the same TF. Previous results [1] show that (especially if multiple wavelength division multiplexing channels are deployed on optical links between fractional λ switches) high link utilization can be achieved with negligible blocking using a Banyan network without speedup.

C. *Non-pipelined Traffic*

Non-pipelined (i.e., non-scheduled) IP packets, i.e., packets that are not part of a SVP (e.g., IP best-effort packets), can be transmitted during any unused portion of a TF, whether it is not reserved or it is reserved but currently unused. Consequently, links can be fully utilized even if flows with reserved resources generate fewer packets than expected. A large part of Internet traffic today is generated by TCP-based elastic applications (e.g., file transfer, e-mail, WWW) that do not require a guaranteed service in term of end-to-end delay and jitter. Such traffic can be dealt with as non-pipelined and can benefits from statistical multiplexing.

Each TDP node performs statistical multiplexing of best-effort traffic, i.e., inserts best-effort packets in unused TF portions. Therefore, SVPs are not at all TDM-like circuits: SVPs are virtual channels providing guaranteed service in terms of bandwidth, delay, and delay jitter, but fractions of the link capacity not used by SVP traffic can be fully utilized. Moreover, any service discipline can be applied to packets being transmitted in unused TF portions. For example, various traffic classes could be implemented for non-pipelined packets in accordance to the Differentiated Services model [10]. In summary, pipeline forwarding is a best-of-breed technology combining the advantages of circuit switching (i.e., predictable service and guaranteed quality of service) and packet switching (statistical multiplexing with full link utilization) that enables a true integrated services network providing optimal support to both multimedia and elastic applications.

Since in TDS switching is based on time, statistical multiplexing of best-effort traffic is not provided at each node. Nevertheless, best effort packets can be inserted at the TDS network edge in any unused portion of a proper SVP based on their destination. Although with somewhat limited flexibility compared to TDP, statistical multiplexing of best-effort packets and high link utilization can be achieved.

When the network is highly loaded and almost fully booked, unused capacity between almost fully reserved TFs might be smaller than the size of queued non-pipelined packets. In such situation, non-pipelined queues cannot be emptied and a fraction of link capacity is wasted. This can be avoided by applying non-disruptive preemptive priority as proposed in [2]. The data link layer at the transmitting end of a link splits non-pipelined packets over multiple frames fitting in the unused bandwidth at the end of subsequent TFs. The receiving end of the link reassembles the received fragments. For example, an extension of Multilink PPP in the time domain could be deployed to this purpose.

III. Optimally scalabile design

Pipeline forwarding is a method known to provide optimal performance independent of specific implementation. Introduced by Henry Ford, and still deployed today, in manufacturing processes, pipeline forwarding is part of computers' central processing unit (CPU) operating principles.

Applying pipeline forwarding to IP packets over the Internet enables the construction of a 10-50 Tb/s switch (Fig. 4) in a single chassis with the following optimal properties: (*i*) only input buffers of minimum size with optimal speedup of 1, (*ii*) switching complexity O[$N*log_aN$], (*iii*) switching speedup of 1, (*iv*) minimum switching controller complexity, and (*v*) unaffected support of (TCP-based) elastic traffic through best-effort or differentiated service.

In the switch design shown in Fig. 4, non-pipelined IP packets (i.e., packets that are not part of the reserved traffic, such as IP best-effort packets) can be effectively supported by a hybrid design. In essence, streaming media and large file transfer are handled accommodates optimally through pipeline forwarding, while elastic best-effort traffic (which will constitute only a small fraction of the future traffic) through traditional high complexity routing.

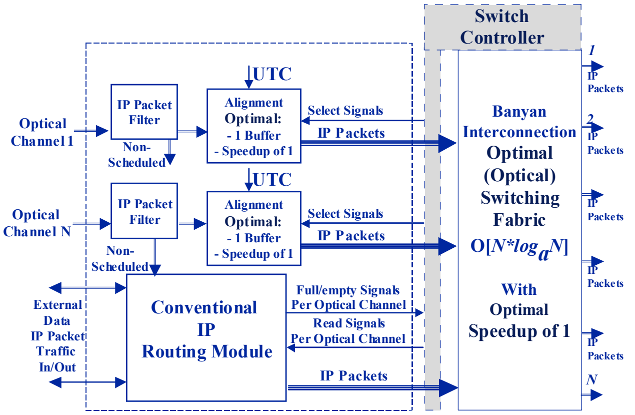


Fig. 4: Hybrid IP routing and UTC-based pipeline forwarding

IV. Experimentation

UTC-based pipeline forwarding was implemented recently in prototypal switch at the University of Trento that is scalable to multi-terabit/s switching capacity. The successful implementation, that required a few (master and PhD) students and researchers and took only 9 months, is a direct outcome of the simplicity (and optimality) of the pipeline forwarding method. The simplicity of this realization did not compromise two most desired performance properties for the future Internet: (1) switching scalability to 10 Tb/s in a single chassis and (2) predictable QoS performance for streaming media and large file transfers.

In particular, two key issues in the scalability of the switch are the switching fabric and its controller. The former is implemented by interconnecting in a Banyan topology (i.e., the lowest complexity, thus most scalable, interconnection network) commercially available Mindspeed M21151 switches, that are 144 x 144 crosspoint switches with a transfer rate or 3 Gb/s (i.e., a 400 Gb/s switching capacity).

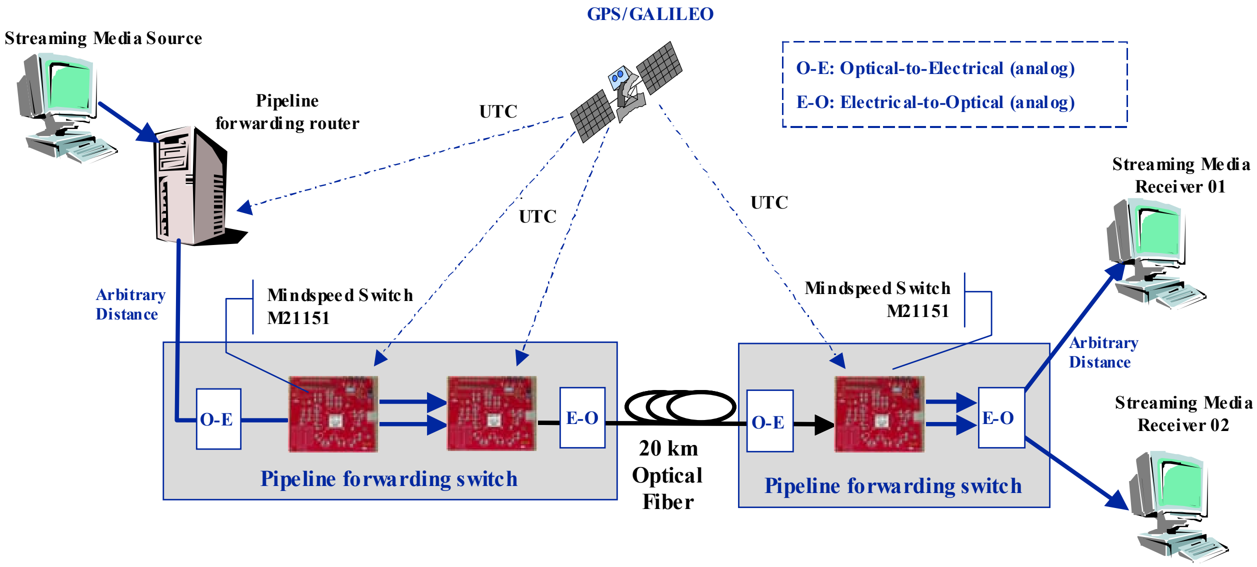
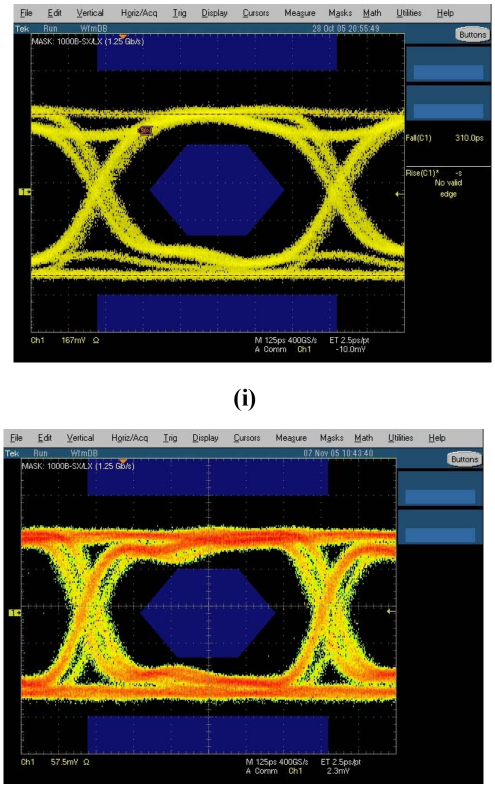


Fig. 5: Multi-Tb/s switch testbed prototype setup

The switch controller was implemented with limited effort on an FPGA. It receives a pulse per second signal from a GPS receiver to realize the common time reference and controls multiple switches.

The prototypal pipeline forwarding switch was deployed in the testbed shown in Fig. 5 that includes also a prototypal pipeline forwarding router developed at the Politecnico di Torino [8]. Two streaming video flows are generated by a video server (to the left), transported, with *deterministic quality of service*, through a network of one router and two multi-terabit/s switches (all implementing pipeline forwarding) and delivered to two different video clients. The pipeline forwarding router is responsible for time-shaping the packet flows generated asynchronously by the vireo streaming sources, i.e., to forward packets towards the first multi-terabit/s switch during the proper TFs. IP packets carrying video samples are transported unchanged as a whole end-to-end. Namely, no change can be seen by observing packets flowing on any link of the testbed as only conventional IP packets encapsulated into Ethernet frames travel across the network testbed.

Moreover, the switch underwent an extensive evaluation by means of various measurements. For example, data integrity has been validated at the output of each switch by matching the *eye pattern* with a standard Gigabit Ethernet 1000 base SX/LX test mask [9]. The result is shown in and for location 1 and location 4, respectively, from a snapshot of the screen of an oscilloscope. The signal received at the measurement point is sampled by the oscilloscope and its value plotted on the screen, one yellow dot for each sample. The rectangles and hexagon drawn on the screen reproduce the transmitted eye mask defined in Section 38.6.5 of [9]; the signal is conformant, hence can be correctly received by a compliant receiver, as long as its plotted measures do not touch these geometrical shapes. Noteworthily, the signal measured at the exit of the second switch has traveled through 25 km of single mode fiber. In addition each data link, including switch boards and optical transceivers (GBIC), has been tested for bit error rate (BER) from 0.1 to 3.0 Gb/s, in order to allow for high safety margins. Note that the boundaries of all these eye diagrams are within the range specified by masks (in dark blue color), hence the signal passes the compliance test. Moreover there is sufficient margin between the signal and the mask, which means a good level of distortion and noise can be accommodated.


File Edit Vertical Horiz/Acq Trig Display Cursors Measure Masks Math Utilities Help
Tek Run WfmDB 28 Oct 05 20:55:49
MASK: 1000B-SX/LX (1.25 Gb/s)
Buttons
Fall(C1) 310.0ps
Rise(C1)* -s
No valid
edge
Ch1 167mV Ω
M 125ps 400GS/s ET 2.5ps/pt
A Comm Ch1 -10.0mV
File Edit Vertical Horiz/Acq Trig Display Cursors Measure Masks Math Utilities Help
Tek Run WfmDB 07 Nov 05 10:43:40
MASK: 1000B-SX/LX (1.25 Gb/s)
Buttons
Ch1 57.5mV Ω
M 125ps 400GS/s ET 2.5ps/pt
A Comm Ch1 2.3mV

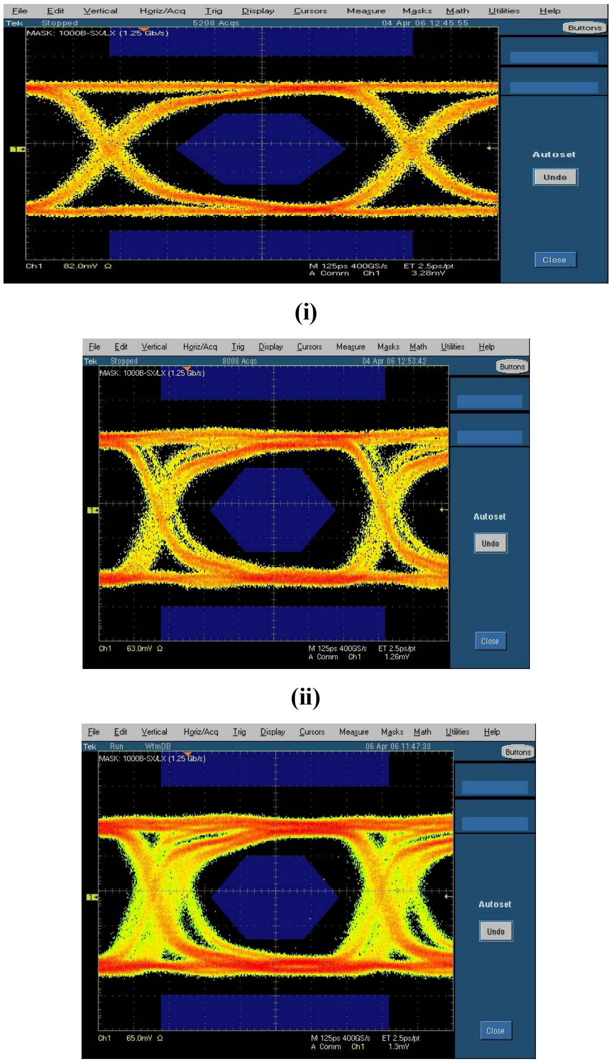
(i)

(ii)

Fig. 6: Eye Pattern at (i) the ingress to the first pipeline forwarding switch and (ii) at the egress of the second pipeline forwarding switch in Fig. 5

To further evaluate the robustness of the solution and assess its limits, the testbed was extended in two ways: (1)

using four 25 km segments of fiber — for a total of a 100 km single mode fiber — and (2) cascading six TDS switches for a total of nine stages of cross-point switches. The standard eye pattern test was performed in various location of this advanced testbed configuration and some of the results are shown in Fig. 7. Specifically, the test was performed on the signal after it has traveled through two switches and 25 km of fiber (Fig. 7-i), three switches and 50 km of fiber (Fig. 7-ii), and 5 switches and 100 km of fiber (Fig. 7-iii). The measurements show that, although the eye patterns have gradually reduced quality as the signal travels through more stages of cross-point switches and through more fiber, the signal is still compliant with the standard, i.e., it can be properly received.


File Edit Vertical Horiz/Acq Trig Display Cursors Measure Masks Math Utilities Help
Tek Stopped 5208 Acqs 04 Apr 06 12:45:55
Buttons
MASK: 1000B-SX/LX (1.25 Gb/s)
Autoset
Undo
Close
Ch1 82.0mV Ω
M 125ps 400GS/s ET 2.5ps/pt
A Comm Ch1 3.28mV
File Edit Vertical Horiz/Acq Trig Display Cursors Measure Masks Math Utilities Help
Tek Stopped 8008 Acqs 04 Apr 06 12:53:42
Buttons
MASK: 1000B-SX/LX (1.25 Gb/s)
Autoset
Undo
Close
Ch1 63.0mV Ω
M 125ps 400GS/s ET 2.5ps/pt
A Comm Ch1 1.26mV
File Edit Vertical Horiz/Acq Trig Display Cursors Measure Masks Math Utilities Help
Tek Run WfmDB 06 Apr 06 11:47:33
Buttons
MASK: 1000B-SX/LX (1.25 Gb/s)
Autoset
Undo
Close
Ch1 65.0mV Ω
M 125ps 400GS/s ET 2.5ps/pt
A Comm Ch1 1.3mV

(i)

(ii)

(iii)

Fig. 7: Eye Pattern of the signal after traveling through (i) two switches and 25 km of fiber, (ii) three switches and 50 km of fiber, and (iii) 5 switches and 100 km of fiber

REFERENCES

[1] C-S. Li, Y. Ofek, A. Segall and K. Sohraby, "Pseudo-isochronous cell forwarding," Computer Networks and ISDN Systems, 30:2359-2372, 1998.

[2] C.-S. Li, Y. Ofek, and M. Yung, Time-driven priority flow control for real-time heterogeneous internetworking, in *Proc. IEEE (INFOCOM' 96), vol. 1*, (Mar. 24–28, 1996), 189–197.

[3] D. Grieco, A. Pattavina and Y. Ofek, "Fractional Lambda Switching for Flexible Bandwidth Provisioning in WDM Networks: Principles and Performance," *Photonic Network Communications*, Issue: Volume 9, Number 3, Date: May 2005, Pages: 281 – 296.

[4] V. T. Nguyen, R. Lo Cigno, Y. Ofek, "Design and Analysis of Tunable Laser-based Fractional Lambda Switching (FLS)," IEEE INFOCOM 2006.

[5] M. Baldi and Y. Ofek, "Fractional Lambda Switching - Principles of Operation and Performance Issues", *SIMULATION: Transactions of The Society for Modeling and Simulation International*, Vol. 80, No. 10, Oct. 2004, pp. 527-544.

[6] M. Baldi, Y. Ofek, B. Yener "Adaptive Group Multicast with Time-Driven Priority," *IEEE/ACM Transactions on Networking*, Vol. 8, No.1, Feb. 2000, pp. 31-43.

[7] D. Grieco, A. Pattavina and Y. Ofek, "Fractional Lambda Switching for Flexible Bandwidth Provisioning in WDM Networks: Principles and Performance", *Photonic Network Communications*, Issue: Volume 9, Number 3, Date: May 2005, Pages: 281 – 296.

[8] M. Baldi, G. Marchetto, G. Galante, F. Risso, R. Scopigno,F. Stirano, "Time Driven Priority Router Implementation and First Experiments," *IEEE International Conference on Communications (ICC 2006), Symposium on Communications QoS, Reliability and Performance Modeling*, Istanbul (Turkey), June 2006.

[9] IEEE 802.3 Working Group, "Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications," IEEE Std 802.3 2000 Edition, The Institute of Electrical and Electronics Engineers, 2000, ISBN 0-7381-2673-X

[10] S. Blake *et al*., "An Architecture for Differentiated Services," *IETF Std. RFC 2475*, Dec. 1998.