Scalable Fractional Lambda Switching: a Testbed

Mario Baldi, Michele Corrà, Giorgio Fontana, Guido Marchetto,
Yoram Ofek, Danilo Severina, and Olga Zadedyurina

An early version of the paper was presented at IEEE ICC 2007 – Optical Networks and Systems Symposium where it received the Best Paper Award.

***Abstract*—This paper presents experiments on a testbed based on ultra-scalable switches realized using off-the-shelf optical and electronic components. The scalability of this switching architecture, direct outcome of the deployment of pipeline forwarding, results — in addition to much lower cost — in the need for a smaller amount of components, and consequently, lower power dissipation, which is key to a "greening" of the Internet. Although an all-optical architecture is demonstrated, we reached the conclusion that given the current state of the art, a hybrid electro-optical architecture is the "best-of-breed" switch solution.**

***Index Terms*—optical networks, multi terabit packet switching, sub-lambda switching, time-driven switching, pipeline forwarding, testbed.**

I. Introduction

The steady (exponential) growth of the Internet over the past few years is impressive, but applications so far deployed are nothing in terms of bandwidth and service requirements when compared to multimedia ones, especially if interactive in nature. In fact, bandwidth demand is expected to grow significantly driven in particular by the growth in popularity of multimedia applications (see some recent predictions by Cisco Systems [1]). A large part of those applications will be deployed by home users, who are accustomed not to pay very much. Furthermore, the long term success of such bandwidth intensive applications, together with the opportunity for service providers to benefit from new sources of revenue, is dependent on the cost of such predictable service. Consequently, over-provisioning, which is the basis for providing predictable services on today's networks, is not likely to be a viable solution to accommodate these growing new types of traffic. This considered, how will the Internet be able to support this additional growth and who will pay for the construction and energy consumption of such a network? The development of highly scalable (i.e., to multi-terabit/s) switches that can handle such media-based applications, while ensuring the right level of service quality [2], is urgently needed. *Scalability* implies not only (*i*) the *capability* of achieving high bit rates of aggregated switched traffic, but also (*ii*) lower *cost* and (*iii*) lower *energy consumption* per bit switched, which is emerging as a major requirement for the future Internet expected to be "green" [3]-[5].

Wavelength division multiplexing (WDM), originally proposed as a solution to tap into the large transmission potential of fiber optics to increase the capacity of existing and future fiber infrastructures, has been used as the basis for the realization of high performance switching systems and multi-wavelength optical networks that have been the subject of research for many years (e.g., [6][7]). Optical switching has the potential of being highly scalable, in all three dimensions mentioned above: capability of building a high performance architecture, low cost and low power consumption. However, besides the cost of optical components still being very high[1], the bandwidth granularity in lambda (λ) switched networks is the one of a whole optical channel, i.e., only either the whole optical channel capacity or nothing can be allocated. Switching a whole optical channel is not efficient since each optical channel has a capacity ranging from 2.5 Gb/s to 100 Gb/s (in the near future), which accommodates a very large number of sessions/connections/flows. The obvious solution is the implementation of asynchronous IP-packet switching, but when looking at all-optical networking, asynchronous IP-packet switching is not yet practical. Moreover, future traffic will include a large portion of multimedia distribution that is inherently predictable and of long-duration (i.e., minutes or more).

More recently, Optical Burst Switching (OBS) [8] has been proposed for the realization of high performance asynchronous switching systems. A burst accommodates a possibly large number of packets. In OBS, control packets are transferred on a control channel to configure switching nodes before the arrival of corresponding bursts, reducing the requirements for optical buffers. Though OBS is interesting and some solutions for quality of service were defined for it (e.g., [9]-[12]), the asynchronous nature of burst switching makes the necessary

This work was supported in part by funds from the European Commission (contract No MC-EXC 002807).

M. Baldi and G. Marchetto are with the Department of Control and Computer Engineering, Politecnico di Torino, Torino, Italy (e-mail: {mario.baldi, guido.marchetto}@polito.it).

M. Corrà is with the TRETEC s.r.l., Trento, Italy (e-mail: michele.corra@3tec.it).

G. Fontana, Y.Ofek, D. Severina, and O. Zadedyurina are with the Department of Information and Communication Technology, University of Trento, Trento, Italy (e-mail: {giorgio.fontana, ofek, severina, olga.zadedyurina}@disi.unitn.it).

[1] The cost of optical components, currently very high due to the immaturity of the technology, is expected to decrease, due to economy of scale and their inherent simplicity.

switching fabrics hard to implement and control, even when the traffic load is moderate.

Traditional TDM (time division multiplexing) systems, such as SONET/SDH, represent a synchronous solution that uses frequency synchronization with known bounds on clock drifts. In order to overcome possible data loss, SONET/SDH is using rather complex overhead information to accommodate: (1) the accumulation of delay uncertainties and (2) continuous clock drifts from the nominal value. Multiplexing in SONET/SDH is based on time slots (TS) organized in a reoccurring structure called frame. Due to the lack of phase synchronization among nodes (resulting from the abovementioned continuous clock drifts), a TS incoming from an input interface might be stored up to the duration of a whole frame before being sent out on its output link. In order to keep the delay introduced by each node small, the frame duration is defined as 125 μs and, consequently, the TS is defined to hold small-size data units, specifically one byte. The 1-byte TS, with duration of about 1 ns at 10 Gb/s (OC-192/STM-48) and the other mechanisms required to cope with continuous clock drifts (such as the floating payload) significantly contribute to SONET/SDH complexity and cost, and make its optical implementation impractical.

This paper presents a testbed demonstrating a method known as *fractional λ switching* (FλS), a.k.a., *time-driven switching* (TDS), based on the application of *pipeline forwarding* (see, for example, [13][14]). In FλS, all nodes share a *common time reference* (CTR) that coordinates switching operations in the entire portion of the network in which the solution is used. In essence, fixed size time slots are defined and used as minimum switching unit. In this way, the optical channels can be partitioned into a number of sub-lambda (or "fractional" lambda) channels that can be switched independently, thus making the network more bandwidth efficient than whole optical channel switching. Scalability is maximized as neither (*i*) *header processing*, nor (*ii*) reliance on *small size switching units* (such as in SONET/SDH), nor (*iii*) *large buffering* is required as "stopping of the serial bit stream" is minimized. Furthermore, FλS provides service guarantees for multimedia applications as a "bonus", i.e., without any added complexity, i.e., cost and power consumption. In summary, FλS enables:

1. *High scalability* of network switches to 10-100 Tb/s in a single chassis;
2. *All-optical implementation* with state of the art technology;
3. *Quality of service guarantees* (deterministic delay and jitter, and no IP packet loss) for (UDP-based) constant bit rate (CBR) and variable bit rate (VBR) streaming applications — as needed; while
4. Preserving the *support of elastic, TCP-based traffic*, i.e., existing applications based on "best-effort" services are not affected in any way.

The primary objectives of this work are: (*i*) to demonstrate the feasibility of our design principles, (*ii*) to confirm the low complexity, cost and energy consumption of FλS by implementing a testbed using off-the-shelf optical and electronic components and (*iii*) to provide a testbed for measurements and further experimental studies. It is worth mentioning that although one-way media streaming is deployed in the presented experiments for simplicity, the very low end-to-end delay and jitter guaranteed by the system make it ideal for the support of interactive services. Even though an all-optical FλS switch prototype is presented to demonstrate the feasibility of the implementation fully in the optical domain, a hybrid electro-optical architecture is concluded to be the "best-of-breed" switch solution given the current state of the art.

The potential of using time to drive switching operations in optical networks is confirmed by several other works. For example, in CANON [15][16] the network is divided in clusters interconnected by a mesh optical network. Within each cluster, nodes are connected to form an optical ring accessed in a TDMA fashion, thus based on a common time reference. A proper reservation of time slots coordinated by a Master Node allows avoiding contentions and packet loss on each ring. Since the common time reference is not global, contentions may instead occur in the mesh network connecting the Master Nodes, where different optical switching solutions (e.g., optical interconnection, wavelength switching, OBS) may be used depending on the network complexity and expected traffic profile. CANON and FλS are complementary. First, by extending the common time reference beyond the single clusters, FλS could be deployed on the mesh optical network interconnecting Master Nodes and seamlessly integrated with the TDMA, thus extending the congestion free service end-to-end. Second, the hierarchical, on-the-fly resource reservation solution proposed in CANON could be deployed in FλS for which periodic reservation had been so far proposed. Time Sliced OBS (TSOBS) [17] applies the concept of time-driven operation in the context of OBS to limit the buffering requirements of OBS switches. A time reference defines fixed size time slots, grouped in *frames*, which are reserved to incoming bursts. Since each burst can allocate any of the available time slots, it may be buffered for a whole frame before being forwarded. In order to keep buffering requirements compatible with current optical technology, the frame must be short (e.g., 100 ns – 1 μs), particularly with high capacity links. The main difference between TSOBS and FλS is the scope of the time reference, which is common to all network nodes in FλS, and local to each switch in TSOBS. As a consequence, the time difference between the beginning of frames on the inputs of TSOBS switches (defined by the time reference on upstream nodes) and the ones on the outputs is not constant. This makes the deployment of immediate forwarding (explained in Section II.A) with its very limited buffering requirements impossible and buffering for at least one frame necessary.

The rest of the paper is organized as follows. Section II presents the basic concepts and ideas underlying the proposed scalable switch design. The architecture and the

implementation of a prototypal switching system are presented in Section III. Section D is devoted to the switch controller card that is at the heart of the switching system. Section IV presents the testbed realized with prototypal switches and the results of some testing and measurement experiments, most significantly with six nodes and 100 km of single mode fiber (4 fiber segments of 25 km). Section V summarizes the work and draws some conclusions.

II. Scalable Design

A. Pipeline forwarding with UTC

Pipeline forwarding of packets is one of the key components of our scalable switch design. Thus, this section briefly introduces this method. An extensive and detailed description of pipeline forwarding is outside the scope of this paper and is available in the literature [13][14] [18][19].

The necessary condition for pipeline forwarding is having a *common time reference* (CTR). See for example the initial work presented in [20]. In the design presented in this paper, the UTC (coordinated universal time) is used. This is globally available via tens of sources on earth and in space (e.g., GPS (USA) and GLONASS (Russian) and in the near future Galileo (EU) and Beidou (China)), and can be easily distributed through the network itself (i.e., in-band) using a variety of mechanisms and standard protocols, e.g., [21].

All packet switches utilize a time period called time frame (TF). The TF duration T, that does not have to be the same throughout the network, is derived as a fraction of the UTC second. As shown in Figure 1, TFs are grouped into time cycles (TCs) and TCs are further grouped into super cycles; this timing structure aligned in all nodes constitutes the CTR. Each super cycle duration may be one UTC second, as shown in Figure 1, where the 125 µs time frame duration T is obtained by dividing the UTC second by 8000; sequences of 100 time frames are grouped into one time cycle, and a sequence of 80 time cycles are comprised in one super cycle (i.e., one UTC second).

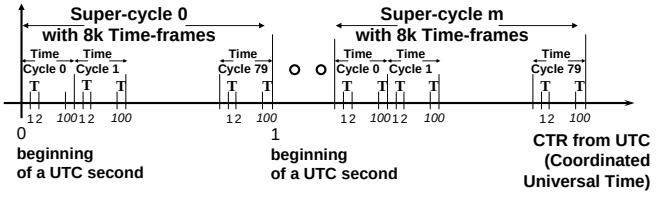

Figure 1. Common time reference structure

During a resource reservation phase TFs are reserved for each flow on the links of its route. Thus, TFs can be viewed as virtual containers for multiple IP packets that are switched and forwarded according to the UTC. The TC provides the basis for a periodic repetition of the reservation, while the super cycle offers a basis for reservations with a period longer than a single TC.

A signaling protocol should be chosen for performing resource reservation and TF scheduling, i.e., selecting the TF in which packets belonging to a given flow should be forwarded by each router. Existing standard protocols and formats should be used whenever possible. Many solutions have been proposed for distributed scheduling in pipeline forwarding networks [19] and the generalized MPLS (G-MPLS) control plane provides signaling protocols suitable for their implementation.

The basic pipeline forwarding operation as originally proposed in [18] is regulated by two simple rules: (*i*) all packets that must be sent in TF t by a node must be in its output ports' buffers at the end of TF t-1, and (*ii*) a packet p transmitted in TF t by a node N_n must be transmitted in TF $t+d$ by node N_{n+1}, where d is a predefined integer called *forwarding delay*, and TF t and TF $t+d$ are also referred to as the *forwarding TF* of packet p at node N_n and node N_{n+1}, respectively. It follows that packets are timely forwarded along their path and served at well defined instants of time at each node. Nodes therefore operate as they were part of a *pipeline*, from which the technology's name is derived. 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.

As demonstrated in many previous publications (e.g., [18][19]), pipeline forwarding guarantees that real-time traffic transmitted according to a reservation experiences: (*i*) bounded end-to-end delay, (*ii*) low delay jitter independent of the number of nodes traversed with (*iii*) no congestion and resulting packet loss. Whenever a flow generated by a source exceeds its reservation, excess packets are dealt with at the ingress to the FλS portion of the network (see the concept of FλS interface in Section D) where they might be buffered (delayed), statistically multiplexed on purposely provisioned "best-effort" SVPs, and possibly experience loss.

Pipeline forwarding is the basic operating principle underlying *fractional lambda switching* (FλS), also known as *time-driven switching* (TDS). In FλS, all packets in the same TF are switched in the same way, i.e., through the same output port. It is based on the setting of a switching schedule of IP packets in TFs along a predefined route/path in the FλS network, which results in the partitioning of optical channels into a number of sub-lambda (or "fractional") channels. Header processing is not required, with consequent low complexity and, hence, high scalability. In FλS there are two main types of TF forwarding, wherein each TF contains multiple packets:

1. Immediate forwarding (IF): upon the arrival of each TF to a FλS switch, the content of the TF (e.g., IP packets) is scheduled to be "immediately" switched and forwarded to the next switch during the next TF. Hence, the buffer that is required is bounded to one TF and the end-to-end transmission delay is minimized. Consequently, the amount of buffer for one TF of 12.5 µs at 10 Gb/s is only 15 KB. Clearly, no buffering is required if IP packets scheduled in TF k arrive to the switch at the exact time TF k begins.

2. Non-immediate forwarding (NIF): TFs may be delayed at the input of the switch for a predefined number of additional TFs, typically, one or two (e.g., 2-frame forwarding up to k-frame forwarding). The main advantage of NIF is the reduction in blocking probability, which is the probability that there are available TFs but not in the right sequence or schedule.

B. Design guidelines

The design approach is based on the following three implementation principles which, as it is shown below, lead to high scalability:

1. Electronic and/or optical switching components, with
2. Optical interconnection of the switching components, and with
3. Global *pipeline forwarding* (PF) with *time-driven* switching and control.

Today single-chip, high-capacity electronic cross-point switches are available on the market. For example, a 144-by-144 switch matrix with 0-11 Gb/s per input/output port, i.e., up to 1.5 Tb/s aggregate switching capacity, dissipating 0.1-0.15W per input-output pair. However, electrical interconnects suffer from high wire resistance, residual wire capacitance and inter-wire crosstalk as the length and/or the density of the electrical interconnections increases. Hence, constructing a large switching matrix based on such cross-points may require optical interconnects as they allow, at least in principle, any desired interconnection topology, while minimizing various noise and interference sources (see, for example, [22][23]). Finally, as shown in [13][14], the use of universal time leads to an optimal switch design featured by:

1. Input ports with optimal memory access speedup of 1 (s=1), where speedup is defined as the ratio between the link bandwidth and the memory access bandwidth.
2. Input ports with a single queue/buffer (note that typical input buffered switches have N queues – one queue per output in order to prevent head-of-the-line blocking).
3. Optimal output port design without buffers, i.e., the IP packets are forwarded from the switching fabric directly onto the out-going link.
4. Optimal switching fabric complexity by using multistage Banyan interconnection networks [24], with switching complexity of $O(a \cdot N \cdot \log_a N)$, where N is the number of input/output and a is the size of each switching element or block. This aspect is further detailed in the following subsection.
5. Switch control complexity equal to the fabric complexity, $O(a \cdot N \cdot \log_a N)$.
6. Guaranteed quality of service with sub-lambda switching.

To summarize the above discussion, our design is guided by the clear limitations of both "all-optical" and "all-electrical" switching approaches. Our current conclusion is that a hybrid architecture is currently the "best-of-breed" switching solution in order to achieve 10-100 Tb/s switching capacity in a single chassis. Specifically, our current design approach is based on time-driven (i.e., pipeline forwarding based) electronic switching with optical interconnects and without "stopping" the serial bit stream. This may change as optical switching components evolve. It is worth noticing that, although there are many technical challenges, no physical breakthrough is required in order to realize dense electrical/optical switches with optical interconnections.

C. Switch scalability

In order to maximize the switching fabric scalability it is necessary to minimize its complexity (which also minimizes cost and power consumption). The lowest complexity fabric are multistage Banyan interconnection networks, with switching complexity of $O(a \cdot N \cdot \log_a N)$ [24]. Banyan is known to suffer from space blocking, which means that a connection between an available input and an available output may not be possible because there is no available route through the switch interconnection network. However, as shown in [13], with UTC-based pipeline forwarding of data packets *space blocking* has just the effect of reducing the achievable link utilization. Intuitively, the multiple TFs in each time cycle provide an additional degree of freedom for scheduling. Namely, having K TFs in each time cycle is equivalent to having K different possible switch instances that can be used for moving data from inputs to outputs. In addition, non-immediate forwarding of TFs can be used: a TF (as virtual container with multiple IP packets) may be delayed one or more TFs before being switched and forwarded, which increases the chances of matching viable interconnections through subsequent switches. In summary, the combination of plurality of TFs in each time cycle together with non-immediate forwarding enable the efficient use Banyan-based switching fabric with optimal switching complexity. A detailed performance study is given in [13].

Figure 2 is a possible Banyan-based switching fabric configurations based on the Mindspeed M21151/M21156 cross-point switch [25] that is used in our current prototype, as described in the following sections. The Banyan design using the M21151/6 has an aggregate switching capacity of 10 Tb/s. In summary, FλS enables the construction of Banyan based fabrics, which are the most scalable switch design, by significantly limiting the impact of their space blocking nature. Figure 2 is a low-power design of a Banyan based fabric with switching capacity of 10Tb/s. The design is based on the optical interconnection of commercially available electrical switching devices. In comparison, all-optical switches with switching time below 1μs have much larger physical size and are more expensive. However, all-optical switches can operate at line rates of 100s of Gb/s per optical channel.

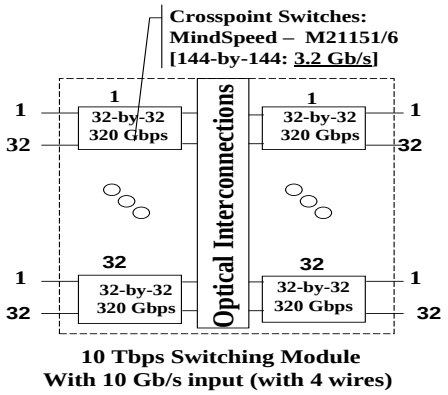

Figure 2. 10 Tb/s Switching Module with M21151/M21156 implemented in the current testbed

D. Multimedia system architecture

Fully benefiting from FλS and UTC-based pipeline forwarding requires providing network nodes and end-systems with a CTR and implement network applications in a way that they use it to maximize the quality of the received service. However, the Internet is currently based on asynchronous IP packet switches and IP hosts. Thus, especially in an initial deployment phase, UTC-based pipeline forwarding must coexist and interoperate with current asynchronous packet switches, hosts and applications (e.g., IP video-phones, video-streaming servers and clients, etc.). This is achieved with a network architecture such as the one depicted in Figure 3, where an intermediate synchronous switching domain is used between the asynchronous edge and FλS core. This is realized with the *Time-Driven Priority* (TDP) [18] technology, a pipeline forwarding realization offering higher flexibility (although less scalability) than FλS. In fact, TDP implements UTC-based pipeline forwarding combined with conventional IP/MPLS routing and full hop-by-hop traffic multiplexing. At FλS switches, there is no packet header processing and routing is based on the TF during which packets are sent. Consequently, TDP routers at the backbone edge are instrumental in properly "time-shaping" asynchronous IP packets by transmitting them in pre scheduled TFs. In a way, a TDP router connecting to the FλS core acts as a FλS interface. At resource reservation time, schedules must be defined seamlessly across the TDP backbone edge and the FλS backbone core. Edge TDP routers are connected to traditional asynchronous IP routers through a *SVP* (synchronous virtual pipe) *interface* that controls access to the pipeline forwarding network by policing and shaping the incoming traffic flows, i.e., asynchronous packets are stored in a buffer waiting for their previously evaluated forwarding TF.

Clearly, the best performance is achieved when the pipeline forwarding domain extends until the end-systems [1], i.e., when also multimedia sources are UTC-aware. However, [26] demonstrated how multimedia applications significantly benefits from pipeline forwarding even when deployed in a limited portion of the network, such as in Figure 3. Furthermore, note how several "last-mile" technologies, such as, cable modem (CM) and passive optical network (PON), are also time-based. Hence, they may seamlessly interoperate with our time-based solutions, thus creating end-to-end SVPs.

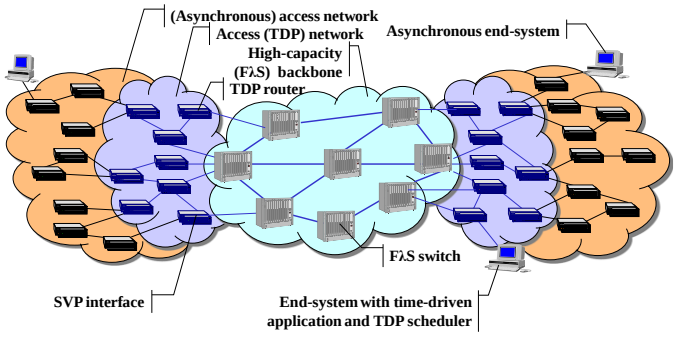

Figure 3. Architecture for deploying pipeline forwarding

E. Future (Green) Internet

High switching scalability implies less hardware per unit of switching capacity, which consequently reduces component and electricity costs. Although reducing power consumption for what is now called "green Internet" [3]-[5] was not one of our initial goals, it is, indeed, a beautiful outcome given the current global energy and environmental crisis.

Specifically, as Internet traffic increases, its power utilization grows. This growth in energy utilization must be considered seriously in the future Internet design otherwise, it may constrain the growth of the network itself. Various studies have shown that current IP packet switching is not energy efficient. However, the time-based IP switching approach demonstrated by our testbed implementation reduces the electricity requirement significantly.

In future studies we will quantify more precisely this electricity reduction in both:

1. Switching, where the proposed architecture brings a major reduction (which can be roughly estimated in a factor of 10-20) in the number of electronic components — ultimately transistors — due to the elimination of buffers and header processing; and
2. Transmission, resulting from the ability to fully utilize each optical channel with sub/fractional lambda switching (FλS), which is not the case when all-lambda switching is deployed (i.e., lambda routing).

Thanks to its switching granularity, FλS enables to extend the cost/energy efficient time-based IP packet switching all the way to the edges of the network, specifically, in the abovementioned "last-mile".

III. Switch Prototype Architecture and Implementation

Guidelines presented in Section II are used to develop an FλS opto-electronic switch prototype, which is part of the wider pipeline forwarding network testbed described in Section IV. This section presents the architecture of the opto-electronic switching system, describing in detail its main components.

The functional diagram of the FλS switch prototype is shown in Figure 4. It has three major parts: (*i*) a field programmable gate-array (FPGA) switch controller, (*ii*) a switching fabric implemented by interconnecting in a Banyan topology commercially available Mindspeed M21151/M21156 cross-point switches, and (*iii*) a GPS time receiver. The switch fabric configuration is determined by the switch controller, which is responsive to timing signals from the GPS receiver.

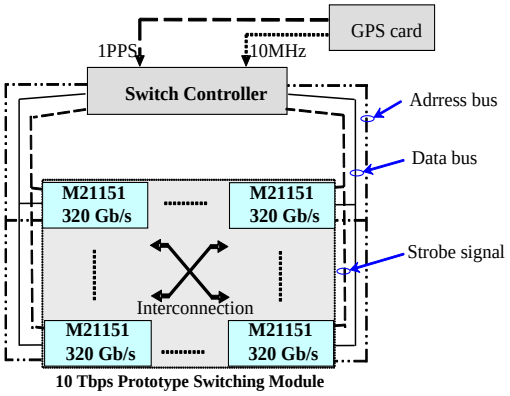

Figure 4. Block diagram of FλS switch: 2-stage Banyan implemented with M21151/M21156 cross-point switches

A. General Description

The GPS receiver provides 1 PPS (pulse per second) and 10 MHz signals (derived from UTC) to the FPGA switch controller. The communication between the switch controller and the cross-point switches can be parallel or serial. There are three main signal types that connect the switch controller with the cross-point switches: address, data and strobe signals, as shown in Figure 4. The cross-point configuration information (i.e., input to which each output should be connected, for all 144 outputs, and for each TF within the time cycle) is stored in the memory table of the FPGA switch controller. Data and address signals are used for writing the switch configuration for the next TF onto the cross-point switches. This writing process shall end before the falling edge of the next strobe signal, which corresponds to the beginning of the next TF and determines the latching of a new switching configuration onto the cross-points. The cross-point switch configuration is ready in less than 10 ns from the falling edge of the strobe signal.

B. GPS Receiver

The GPS receiver, an EPSILON Board OEM II [27], provides accurate and stable time and frequency signals for UTC (coordinated universal time) synchronization. It provides 1 PPS and 10 MHz sine waves, and UTC time-of-day output. Furthermore, the 10 MHz frequency reference is cycle locked to the 1 PPS, which is the standard UTC second. This implies that within 1 PPS there are exactly 10,000,000 cycles of the 10 MHz GPS card output.

C. Mindspeed Cross-point Switch

Mindspeed cross-points M21151/M21156 [25] are the primary components of the presented switch implementation. These are low-power CMOS, high-speed 144-by-144 cross-points. Each input can receive a serial signal from 0 to 3.2 Gb/s — thus, its nominal aggregate capacity is 460 GB/s. The serial input signal goes through a sequence of internal inverters inside the M21151/M21156, and therefore, does not have any power loss. The cross-point switch ports are equipped with input equalization (IE) and output integrated clock data recovery (CDR), which further preserves the serial signal quality. Each CDR is preceded by a programmable IE.

D. Switch Controller Card

The switch controller card is the "brain" of this scalable switching system. The controller is implemented using an Opal Kelly XEM3001 integration module [28]. The XEM3001 consist of an EEPROM, a USB 2.0 micro-controller, a phase locked loop, a 400,000-gate Xilinx Spartan-3 FPGA sub module [28] and a 1MHz to 150MHz multi-output clock generator. The block diagram of the implemented FPGA controller is shown in Figure 5.

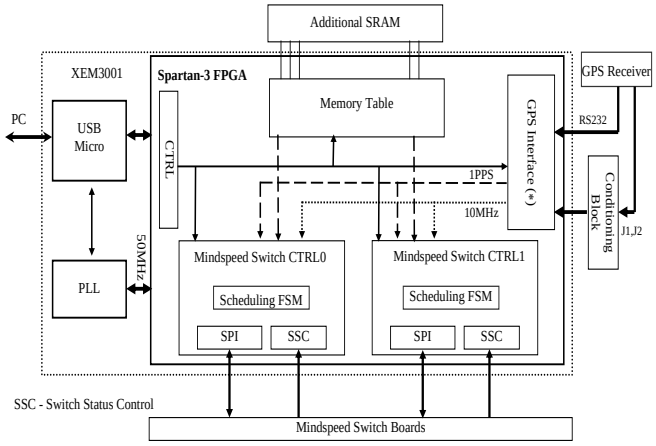

Figure 5: Block Diagram of the Switch control card

The very high speed integrated circuit hardware description language (VHDL) is used for the implementation of the FPGA controller. As shown in Figure 5, the FPGA sub module blocks are control logic (CTRL), memory table, GPS interface and Mindspeed cross-point controller. The implementations of all functional blocks in VHDL are synthesized into a single bit file, which is uploaded from a PC to the FPGA through the USB 2.0 connection. In the prototype presented here, we have used one controller for controlling two M21151 cross-point switches. There are two scheduling Finite State Machines (FSMs) for the two cross-points.

The Mindspeed switch control logic (CTRL 0 & 1, in Figure 5) contains the FSM that controls and connects all the sub-blocks implemented on the FPGA module. Every clock cycle reads the input status register (target_status) connected to a USB wire and acts accordingly. Each scheduling FSM consists of three main counters: TF counter, time cycle (TC) counter, and UTC second counter. Furthermore, each scheduling FSM receives two reference input signals from the GPS receiver: (1) 1 pulse per UTC second (1PPS) and (2) 10MHz. Every UTC second (1PPS) is divided into a

programmable number of TCs *n* and every TC is divided into a programmable number of TFs *m*, each with a predefined duration. Given a TF duration T, it must be $T \cdot n \cdot m = 1$ UTC second, as shown in Figure 1. The value for *n*, *m*, and T can be set by changing the corresponding parameters stored on the FPGA (CTRL 0&CTRL 1) from a PC via the USB link.

During every TF, the FSM downloads the next configuration into the cross-point switch. The next configuration is activated by the *switch strobe* signal, which is sent via SPI interface. The cross-point switch configuration data is stored in a memory table (see Figure 5) where there is a table entry for each TF within the TC. When a new TC starts, the pointer into the memory table is reset to the first memory entry that corresponds to the first TF in the TC. This is the cyclic operation of the CTRL 0 & CTRL 1. The memory table can be configured from a graphical user interface (GUI) on a PC through the USB communication link.

IV. Testbed Overview and Experiments

A pipeline forwarding testbed scaling to multi-terabit/s switching capacity has been developed deploying the previously described FλS switch. This testbed (see Figure 6) implements a prototypal video distribution network providing guaranteed quality of service with deterministic queuing delay and small jitter in network nodes. The multimedia system architecture presented in Section II.D is considered. In particular, two asynchronous video streams are generated by a video server (to the left), transported with *deterministic quality of service* through a network composed of one TDP router and two multi-terabit/s FλS switches, and delivered to two different video clients. IP packets carrying encoded media 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.

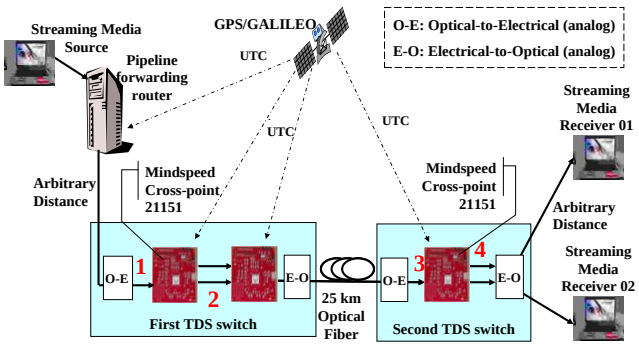

Figure 6. Initial testbed setup without "stopping" the bits

A. Pipeline Forwarding (TDP) Router

Traffic leaving the video server is "asynchronous", thus it has to be "time-shaped" before entering the pipeline forwarding network. In essence, packets have to be sent to the first FλS during the right TFs. This is performed by the pipeline forwarding router, which in our testbed represents both the TDP network edge and the SVP interface (see Section II.D). The implementation of the pipeline forwarding router and the operations it has to perform are briefly described below. A more detailed presentation can be found in [29] and for how scheduling can be done see [30].

The developed pipeline forwarding router is based on the routing software of the FreeBSD 4.8 operating system running on a 2.4 GHz Pentium IV PC; the TDP scheduling algorithm is implemented in the FreeBSD kernel. Generically, in a router data plane packets are moved from input ports to output ports going through three modules that perform *input processing*, *forwarding*, and *output processing*. Thus, the same was realized in our FreeBSD-based pipeline forwarding router.

The router has to shape asynchronous traffic entering the pipeline forwarding network. Consequently, its input module comprises mechanisms to classify incoming packets, identify the data flow they belong to, and select a TF in which they will be forwarded according to the current resource reservation setup.

The forwarding module processes packets according to the specific network technology; specifically, in the presented prototype packet routing and forwarding is based on IP. No modification is required to the forwarding module.

The output module implements a per-TF, per-output queuing system, where packets to be forwarded during the same TF through the same interface are buffered in the same queue. The queue in which each packet is stored is determined by both the input module, which decides the *forwarding TF*, and the forwarding module, which selects the output interface. The output module is also responsible for the timely transmission of all the packets stored in the queues corresponding to the current TF.

The UTC is provided to the router using a Symmetricom bc637PCI-U GPS receiver PCI card [32] that can generate interrupts at a programmable rate ranging between less than 1 Hz to 250 kHz (1PPS and 4 μs).

B. Initial Testbed Set-up

The initial switch prototype experimental setup is shown in Figure 6. The main components are streaming media sources, a pipeline forwarding router for scheduling packet entrance into the first switch, a 25 km single mode optical fiber, two stage source side switch, and single stage receiver side switch. The deployed cross-point switches have 144 channels with capacity of 3.2 Gb/s each.

Two asynchronous streaming media flows — a DVD movie with soundtrack and subtitles and an animation movie with soundtrack — are generated by the streaming media source, as is shown in Figure 6, and transmitted to pipeline forwarding router. The streaming media packets are then forwarded via an optical link by the pipeline forwarding router through Gigabit Ethernet (GE) transceivers to the first FλS switch during different predefined TFs. The first cross-point switch splits packets into two streams forwarded to second cross-point on separate channels. At second cross-point both streams are multiplexed again into one channel that is transmitted via the

25 km single mode optical fiber link to the second switch, which routes each video stream to a different output. Then the separated video streams are forwarded to two receivers through optical links of an arbitrary length. Video flows are therefore multiplexed on the first and second link they traversed, but FλS ensures that video packets reach their corresponding destination with deterministic delay (with no jitter), i.e., during predefined TFs. Switching of all three switching boards and network interfaces are synchronized with the 1PPS signal received from three different GPS receivers.

Data integrity has been validated at the output of each switch (i.e., location 1 and location 4 highlighted in Figure 6) by matching the eye pattern with a standard Gigabit Ethernet 1000 base SX/LX test mask [33]. The result is shown in Figure 7 for location 1 and location 4 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 [33]; 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. Note worthily, the measurement at point 4 shown in Figure 7(b) was performed after 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. Having observed no errors over several test runs involving 1 Tb of data sent across the switch, the BER is lower than 10^{-12} and hence negligible.

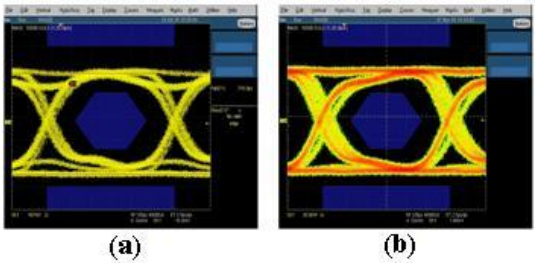

Figure 7 Eye Pattern at (a) location 1 and (b) location 4 in Figure 6

C. Advanced Testbed Testing

To evaluate the robustness of the switch prototype and to know its limits we have extended our testing to a 100 km metro-like network shown in Figure 8. It consists of six nodes, each including either one or two cross-point switches. Five of the six nodes are connected optically through four segments of 25 km optical fiber.

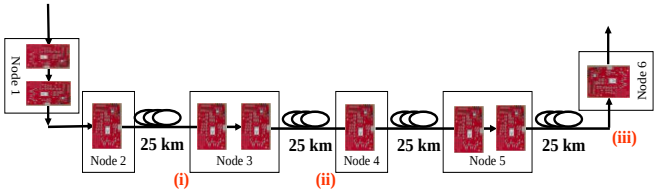

Figure 8. 100 km, 6-node metropolitan network (actually realized using only two FλS switches)

The eye pattern test is performed and measurements were taken at locations (i), (ii) and (iii), as shown in Figure 8. Specifically, Figure 9 shows the eye pattern measurements after 25 km (location i), after 50 km (location ii) and after 100 km (location iii). It is apparent from the eye pattern that there is reduction of safety margin as the optical fiber length increases. However, the signal is compliant to the 1000 base SX-LX standard even after having traveled for 100 km.

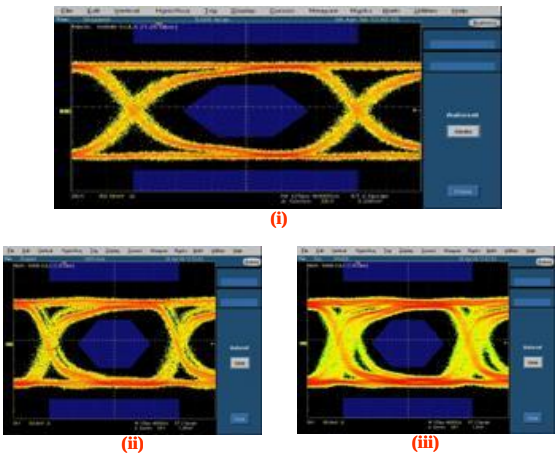

Figure 9. Eye Pattern at locations (i), (ii) and (iii) in Figure 8

The maximum jitter is also measured for each flow in this 100 km testbed. This is computed as the difference between the maximum and the minimum end-to-end delay observed during the whole experiment on packets belonging to the same flow. Delay measurements are performed by time-stamping data packets at the kernel level of the PCs which host the media server and the receivers, respectively. GPS Symmetricom card features are exploited to do this in order to obtain very precise time-stamps. The results obtained are shown in Table I.

TABLE I
END-TO-END DELAY JITTER

Traffic Flow	Jitter [ms]
Media Flow 1	0.14
Media Flow 2	0.16

Delay jitter is low and does not exceed the pipeline forwarding theoretical upper bound 2 T (T =100 μs in our testbed). Furthermore, it is worth noting that in our experiments with flows compliant to their reservation, no packet loss was experienced. This further validates our implementation.

D. Advanced Testbed with an All-optical Switch

Recently an all-optical FλS switch was added to the original testbed (in Figure 6), as shown in Figure 10 and Figure 11. The objective of the all-optical switch is to provide a proof-of-concept for such a hybrid implementation. This has been successfully achieved by using semi-conductor optical amplifiers (SOAs) operating as ON-OFF optical switches. The

all-optical switch facilitates the following primary advantages:

- Eliminating the need for OE (optical-to-electronic) and EO (electronic-to-optical) conversions;
- Serial optical transmission up to hundreds of Gb/s; and
- Interoperability with similar electronic FλS switches that are operating "without stopping the serial bit stream" as it is in the all-optical domain.

In the testbed (Figure 10) the all-optical switch is connected with two 25 km fiber links to the two original FλS electronic switches.

The all-optical switch, shown in Figure 11, is electronically driven and controlled, which can change its state in less than 1 ns. In other words, the all-optical switches require an electronic control system, but the optical data carrier is routed without converting it into electrical signals. The all-optical routing enables the switch to be transparent to optical wavelength and data rates. The current initial switch implementation operates at 1.25 Gb/s Ethernet (GE) at 1550 nm and can also operate at 10 GE (and beyond) at 1550 nm without any change to the current all-optical switch implementation.

The switch consists of ON/OFF Semiconductor Optical Amplifiers (SOAs) interconnected in a network of (3db) passive optical couplers implementing "broadcast-and-select" design. The control mechanism is the same as the one developed for the electronic FλS switch. However, the all-optical switch controller and GPS time receiver are integrated and implemented in a single FPGA in a novel design. This implementation saves considerable amount of money. The all-optical FλS switch includes an extensive diagnostic and testing subsystem, as shown in Figure 11, which is based on LABview.

The current SOA electronic driver operates with power supply of 5 V and 0.2 A, which requires 1 W. However, it is possible to use 1.3 V, which requires only 0.26 W. Since optical channel requires two SOAs, then 100 Gb/s channel will use $0.26 \cdot 2/100 = 0.0052$ W per 1 Gb/s, which is lower than in an electronic cross-connect

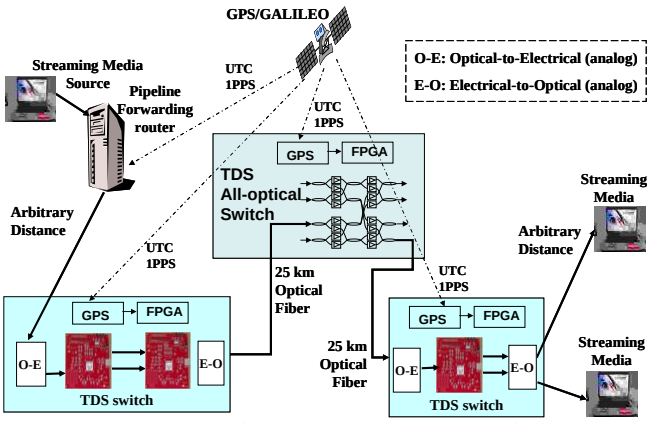

Figure 10: Testbed setup for testing the interoperability of all-optical and electrical FλS switches

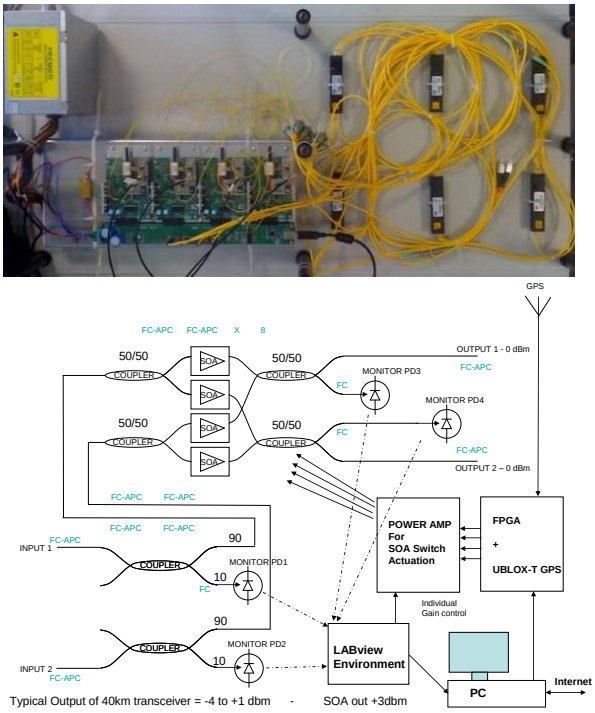

Figure 11: Implementation of all-optical switch including a diagnostic and testing subsystem.

V. Conclusions

Implementing UTC-based pipeline forwarding in a real optical testbed that is scalable to multi-terabit/s switching capacity has been a rewarding experience. The implementation success is a direct outcome of the simplicity of the pipeline forwarding method. The beauty is that the simplicity of this realization does not compromise three highly desired properties for the future Internet:

(1) High switch scalability (e.g., to 10 and 160 Tb/s in a single chassis);

(2) Predictable quality of service (QoS) performance for streaming media and live (sport and entertainment) events, with low end-to-end delay for interactive multimedia applications; and

(3) Low energy (electricity) consumption for a "green", environmentally friendly Internet.

This is significant as it demonstrates that the deployed technology is suitable for providing (1) ultra-scalable switching capacity and (2) sub-lambda reconfigurable optical add-drop multiplexing (ROADM) in metropolitan networks while minimizing the provider cost and complexity per end-user.

References

[1] Cisco Systems, Inc., "Hyperconnectivity and the Approaching Zettabyte Era," June 2, 2010, [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.html.

[2] I. Vaishnavi, P. Cesar, D. Bulterman, O. Friedrich, "From IPTV services to shared experiences: Challenges in architecture design," *IEEE Int. Conf. on Multimedia and Expo (ICME 2010)*, pp.1511-1516, Singapore, July 2010.

[3] M. Gupta and S. Singh, "Greening of the Internet", *ACM SIGCOMM 2003*, Karlsruhe, Germany, Aug. 2003.

[4] K. Christensen, B. Nordman, and R. Brown, "Power management in networked devices", *IEEE Computer Magazine*, Vol. 37, No. 8, pp. 91-93, Aug. 2004.

[5] M. Guenach, C. Nuzman, J. Maes, and M. Peeters, "On Power Optimization in DSL Systems", *IEEE Int. Workshop on Green Communications (GreenComm) at ICC 2009*, Dresden, Germany, June 2009.

[6] R. Ramaswami and K. N. Sivarajan, "Optical networks: a practical perspective," *Morgan Kaufmann Publishers*, 2nd edition, 2001, chapter 8, 9, 10, 11.

[7] A. Pattavina and R. Zanzottera, "Non-blocking WDM switches based on arrayed waveguide grating and shared wavelength conversion," *IEEE Conf. on Computer Communications (INFOCOM 2006)*," Barcelona, Spain, Apr. 2006.

[8] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) – A new paradigm for an optical Internet," *J. of High Speed Networks*, vol. 8, no. 1, pp. 69-84, Jan. 1999.

[9] H. C. Cankaya, S. Charcranoon, and T. S. El-Bawab, "A preemptive scheduling technique for OBS networks with service differentiation," in *IEEE Int. Conf. on Global Communications (GLOBECOM 2003)*, San Francisco, CA, Dec. 2003.

[10] T. Tachibana, S. Kasahara, "Burst Cluster Transmission: service differentiation mechanism for immediate reservation in Optical Burst Switching networks," *IEEE Communications Magazine*, vol. 44, no. 5, pp. 46-55, May 2006.

[11] W.-S. Park, M. Shin, H.-W. Lee, S. Chong, "A Joint Design of Congestion Control and Burst Contention Resolution for Optical Burst Switching Networks," *IEEE/OSA J. of Lightwave Technology*, vol.27, no.17, pp.3820-3830, Sept. 2009.

[12] T. Orawiwattanakul, J. Yusheng, N. Sonehara, "Fair Bandwidth Allocation with Distance Fairness Provisioning in Optical Burst Switching Networks," *IEEE Int. Conf. on Global Communications (GLOBECOM 2010)*, vol., no., pp.1-5, Dec. 2010.

[13] 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, pp. 527-544, Oct. 2004.

[14] V. T. Nguyen, R. Lo Cigno, and Y. Ofek, "Tunable Laser-based Design and Analysis for Fractional Lambda Switches," *IEEE Trans. on Communications*, Vol. 56, No. 6, pp. 957-967, June 2008.

[15] J. D. Angelopoulos, K. Kanonakis, G. Koukouvakis, H.C. Leligou, C. Matrakidis, T. Orphanoudakis A. Stavdas, "An Optical Network Architecture with Distributed Switching Inside Node Clusters Features Improved Loss, Efficiency and Cost", *IEEE J. of Lightwave Technologies*, Vol. 25, No. 5, pp. 1138-1146, May 2007.

[16] A. Stavdas, T. G. Orphanoudakis, A. Lord, H. C. Leligou, K. Kanonakis, C. Matrakidis, A. Drakos, J. D. Angelopoulos, "Dynamic CANON: A Scalable Multi-domain Core Network", *IEEE Communications Magazine*, Vol. 46, No. 6, pp. 138-144, June 2008.

[17] J. Ramamirtham, J. Turner, "Time sliced optical burst switching," *IEEE Int. Conf. on Computer and Communications (INFOCOM 2003)*, pp. 2030- 2038, Apr. 2003.

[18] C.-S. Li, Y. Ofek, and M. Yung, "Time-driven priority flow control for real-time heterogeneous internetworking," *IEEE Int. Conf. on Computer Communications (INFOCOM 1996)*, San Francisco, CA, Mar. 1996.

[19] C.-S. Li, Y. Ofek, A. Segall, and K. Sohraby, "Pseudo-isochronous cell forwarding," *Computer Networks and ISDN Systems*, Vol. 30, No. 24, pp. 2359-2372, Dec. 1998.

[20] Y. Ofek, "Generating a Fault Tolerant Global Clock using High-speed Control Signals for the MetaNet Architecture," *IEEE Trans. on Communications*, Vol. 42, No. 5, pp. 2179-2188, May 1994.

[21] IEEE Instrumentation and Measurement Society, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", *IEEE Std. 1588 (Revision)*, July 2008.

[22] W. Gibbs, "Computing at the Speed of Light" *Scientific American*, Vol. 291, No. 5, pp. 80-87, Nov. 2004.

[23] L. Pavesi and D. J. Lockwood, eds., "Silicon Photonics," *Springer-Verlag*, 2004.

[24] L. R. Goke and G. J. Lipovski, "Banyan Networks for Partitioning Multiprocessor Systems," *ACM Annual Symposium on Computer Architecture (ISCA 1973)*, New York, NY, Dec. 1973.

[25] Mindspeed Technologies, Inc.
[Online] Available: http://www.mindspeed.com.

[26] E. Masala; A. Vesco, M. Baldi, J. C. De Martin, "Optimized H.264 Video Encoding and Packetization for Video Transmission Over Pipeline Forwarding Networks," *IEEE Trans. on Multimedia*, Vol.11, No.5, pp.972-985, Aug. 2009.

[27] Tekelec Systemes
[Online] Available: http://www.tekelec-systemes.com.

[28] Opal Kelly, Inc.
[Online] Available: http://www.opalkelly.com.

[29] M. Baldi, G. Marchetto, G. Galante, F. Risso, R. Scopigno, and F. Stirano, "Time Driven Priority Router Implementation and First Experiments," *IEEE Int. Conf. on Communications (ICC 2006)*, Istanbul, Turkey, June 2006.

[30] Thu-Huong Truong, Mario Baldi, and Yoram Ofek, "Efficient Scheduling for Heterogeneous Fractional Lambda Switching (FλS) Networks," *IEEE Int. Conf. on Global Communications (IEEE GLOBECOM 2007)*, Washington, DC, Nov. 2007.

[31] O. Zadedyurina , Y. Ofek, and A. Pattavina "Space and Time Blocking versus Cost in all optical Banyan Networks," *IEEE Int. Conf. on Communications (ICC 2008)*, Beijing, China, May 2008.

[32] Symmetricom, Inc.
[Online]. Available: http://www.symmttm.com.

[33] 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.