Scalable Optoelectronic ATM Networks: The iPOINT Fully Functional Testbed

J. W. Lockwood, H. Duan, J. J. Morikuni, S. M. Kang,
S. Akkineni, R. H. Campbell

University of Illinois at Urbana–Champaign.
NSF Engineering Research Center for
Compound Semiconductor Microelectronics,[*]
Department of Electrical and Computer Engineering,
Department of Computer Science,[†]
and Beckman Institute of Advanced Science and Technology.
405 N. Mathews Ave.
Urbana, IL 61801

(217) 244-1565
(217) 244-8371 (FAX)
email: ipoint@ipoint.vlsi.uiuc.edu

December 16, 1994

Abstract

Our prototype of a fully-functional Asynchronous Transfer Mode (ATM) switch validates the design of a 128 Gb/s optoelectronic ATM switch. Optoelectronics, rather than all optical componets, are used to simutaneously address all of the specific requirements mandated by the ATM protocol. In this paper, we present the Illinois Pulsar-based Optical Interconnect (iPOINT) testbed, and present our results obtained for the prototype switch in a working environment consisting of an optical network of Sun SPARCStations and other local and wide-area ATM switches.

[*] This research has been supported by National Science Foundation Engineering Research Center grant ECD 89-43166

[†] This research has been supported by SBC CNRI-CAMPBELL-NSF-138

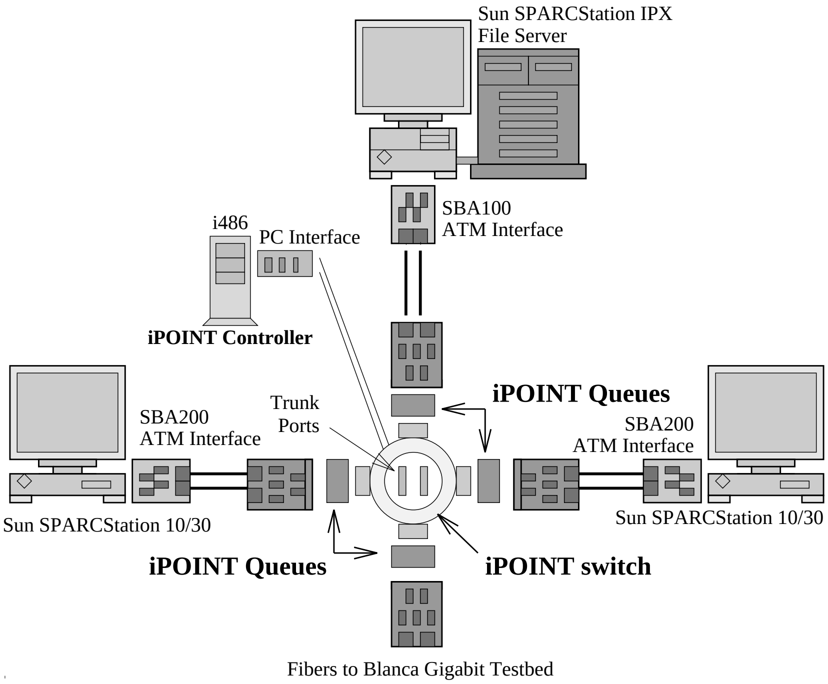

Figure 1: The iPOINT switch

I Introduction

Asynchronous Transfer Mode networks (ATM) will be used to provide a high bandwidth optical backbone for the future National Information Infastructure ("Information Superhighway") [1]. While photonic devices are well suited for high-bandwidth digital communications, the ATM protocol imposes specific switch requirements in terms of cell length, header translation, cell ordering, and buffering that make the design of an all optic switch difficult. In this paper, we present the Illinois Pulsar-based Optical Interconnect (iPOINT) testbed, a scalable switch that maximizes the utility of photonic, electronic, and optoelectronic devices while simultaneously satisfying all of the design requirements for ATM [2]. With a 32-port configuration, the switch will deliver an aggregate throughput of 128 Gb/s.

We have designed and implemented a fully-functional ATM switch and queue modules which prototype the key components of our multi-gigabit/sec design. The switch has an aggregate throughput of 800 Mb/s and is deployed in the iPOINT testbed shown in Figure 1. It optically interconnects two Sun SPARCStation 10/30s, a SPARCStation IPX file server, and a commercial ATM switch that acts as our gateway to the wide-area Blanca/Xunet Gigabit testbed. The switch and queue modules have been implemented using Xilinx Field Programmable Gate Arrays (FPGAs) and the design is ready for device-specific technology-remapping to a GaAs gate-array implementation.

The design of the iPOINT switch was based on the theoretical analysis of the Pulsar switch [3]. Our switch supports general ATM multicast, allowing a cell to be transferred to any permutation of its output ports. The queue module uses input queueing to overcome memory bandwidth limitations, and provides the VPI/VCI translation table functions. A switch controller connects to the switch and queue modules allowing the creation and deletion of dynamic virtual circuits.

After reviewing the motivations for ATM networking, we detail the specific design constraints that led to an opto-electronic design. We discuss the motivation for using input queueing in the switch and then examine other optical computer networking implementations. Next, we describe the iPOINT testbed, and discuss the operation of the switch module, queue module, and switch controller. We describe the Blanca/Xunet Gigabit wide-area testbed and provide performance results of the iPOINT switch when used both in a local-area ATM network as well as in the wide-area testbed. Finally, we describe how the iPOINT switch scales to provide multi-gigabit/sec throughput and present the optical and opto-electronic devices that are being fabricated for use in this configuration.

II Optoelectronic ATM Networking

The ATM technology meets the demands of present and future multi-gigabit computer networks [4] because it offers low-latency, high-bandwidth and asynchronously multiplexed data switching. Circuit-based switching is ill-suited for the transmission of bursty data and compressed video streams. An ATM switch can asynchronously multiplex up to 2^{28} individual connections per link without dedicating fibers, wavelengths, or time-slots to idle or intermittent data sources. The 53-byte length (48-byte payload) of the ATM cell is well suited for multimedia, distributed computing, and real-time applications, where message latency is critical.

ATM interfaces are currently available for Sun, Silicon Graphics, DEC, HP, IBM, and other workstation platforms [5]. These host adapters allow native mode ATM networking, and provide support for existing Internet Protocol (IP) based applications. Applications such as remote login (rlogin), telnet, ftp, Network File System (NFS), and NCSA Mosaic can be run, unmodified, using existing ATM host adapters.

A Design Constraints

The ATM protocol imposes specific requirements on the switch design in terms of cell length, header translation, ordering, and cell queuing [6]. Because of the short cell length, a high-bandwidth ATM switch must provide high cell throughput. Unlike a circuit switch, the data path through the switch is not static; rather, it may be reconfigured on a cell-by-cell basis. To achieve a 90% utilization at link speeds of 4 Gb/s, for example, a space-division switch must provide reconfiguration in less than 10 ns.

The Virtual Path Identifier (VPI) and/or the Virtual Circuit Identifier (VCI) must be translated as the cell passes through the ATM switch. Because the VPI/VCI translation modifies the cell header, it is not possible to build a true ATM switch using only passive waveguides.

When two or more cells are to be switched to the same destination port, only one cell can be transmitted. The other cells must be dropped, deflected, or queued. The dropping of a single ATM cell causes the retransmission of an entire message (AAL5 frame), which introduces inefficiency and performance degradation. The deflection of an ATM cell is not acceptable, as an ATM switch is expected to preserve the order of cells for each virtual circuit. This requirement prohibits the use hot-potato or deflection routing [7]. An ATM switch, therefore, requires queuing. At present, optical storage technologies do not offer the economy or capacity of silicon Random Access Memory (RAM), thus necessitating some combination of optics and electronics.

To meet the requirements listed above, the iPOINT testbed is implemented with optoelectronic components, enabling the maximal use of both photonic and electronic technology. Photonic components excel in terms of data transfer and minimal crosstalk, while electronic components efficiently provide storage and translation.

B Queuing

Data can be queued within a switch at the switch inputs, the switch outputs, or in a common pool of shared memory. The bandwidth of a shared memory switch is fundamentally limited by the access speed of the RAM. Using the largest possible memory width (one cell wide), a minimum of two memory operations are required per cell (one read, one write). For a 50ns RAM cycle time, a simple shared memory switch is limited to $1/(50ns * 2) = 10$ million cells per second, thus limiting the aggregate throughput to 4 Gb/s. Although a slightly faster switch can be built using fast static RAM, it incurs additional expense and still has a limited buffer size [8].

An output-queued switch is also limited by memory bandwidth. In the worst case, an output queue can receive cells from every input port simultaneously, which again requires a buffer with a memory bandwidth equal to the aggregate throughput. In [9], it was shown that for random traffic, an output queue capable of receiving at least eight cells simultaneously could provide an acceptable cell loss probability—comparable to the probability of the link's Bit Error Rate (BER) corrupting the cell due to at least one bit error of the data within the ATM cell. Although this provides an improvement in the memory bandwidth requirements, it still requires the memory bandwidth to be a large multiple of the link rate.

An input queue-based switch requires the least memory bandwidth, and is well suited to meet the requirements of ATM switching. Each queue module is only required to buffer cells at the arrival rate of a single port, rather than at a multiple of the arrival rate or the aggregate arrival rate. From a memory bandwidth perspective, a 32-node ATM switch operating at 4 Gb/s per link could achieve a 128 Gb/s aggregate throughput using low-cost, high-density, economical DRAM devices. For the above reasons, the iPOINT switch design is based on input queuing.

C Other Experimental Optical Testbeds

Bandwidth considerations have motivated research in the design of pure optical networks. *Rainbow* [10] uses WDM to form a local area network using a passive star coupler and tunable optical detectors. Each source (IBM PS/2) transmits an optical signal into the network using a fixed-wavelength, temperature-stabilized, distributed feedback laser. A piezo-tuned fiber Fabry-Perot filter on each receiver scans the wavelengths, until it finds a source transmitting an address that matches it's own network address. Because of the mechanical inertia of the piezo element, *Rainbow* requires 25ms to switch between cells [11]. For an ATM link operating at 1 Gb/s, the reconfiguration time greatly outweighs the cell transmission time (424ns). For random traffic in a 32-node network ATM network, *Rainbow's* throughput would be reduced by a factor of $((31/32) * 25\text{ms})/424\text{ns} = 57,119$, providing an effective throughput of $(1\text{Gb/s})/57119 = 17.5$ kbits/sec, which is less than that available using a telephone-grade modem.

TeraNet [12] somewhat resembles *Rainbow* in that it also employs WDM and passive optical couplers. TeraNet, however, provides efficient packet switching by introducing a Network Interface Unit (NIU). Their NIU is a 3-port electronic packet switch with one port connected to a host, and the other two ports used to transmit and receive optical

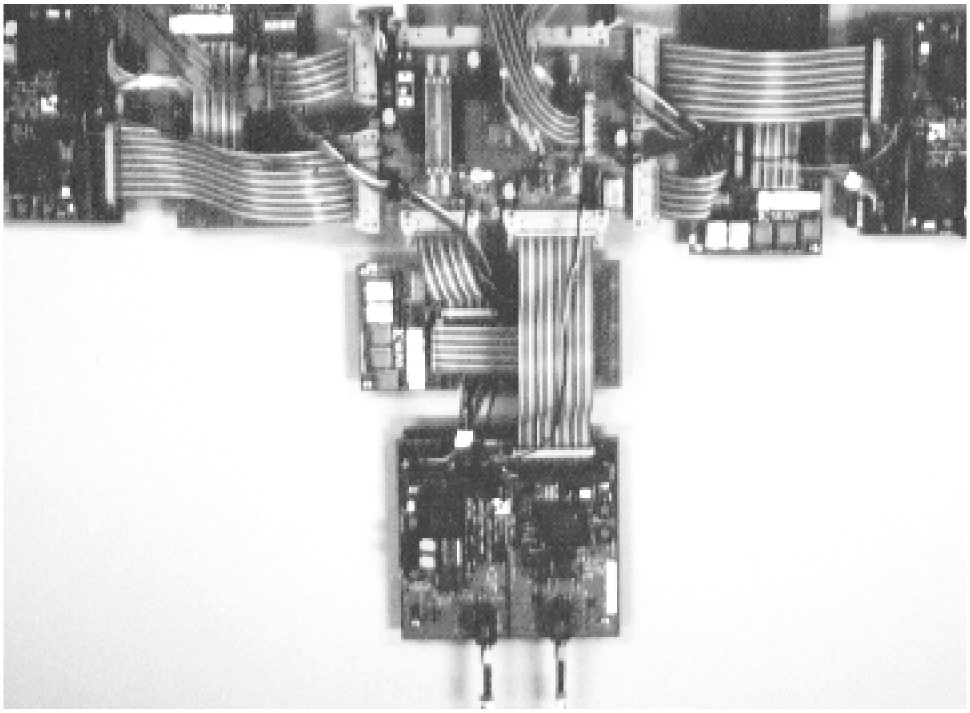

Figure 2: The iPOINT Switch and Queues Modules

data into the passive couplers ("optical ether"). Rather than tuning the wavelength for each cell, wavelengths are tuned over long periods of time, depending on the long-term network demand. An overlay network (a perfect shuffle) is formed over the tuned wavelengths, and cells are forwarded by each NIU until reaching their destination. By equipping some ports of the iPOINT switch with WDM components (the other ports would be used for local optical links and host connectivity), the iPOINT switch could replace the electronic NIU to build large dimension optoelectronic ATM networks.

III The iPOINT Testbed

A fully functional, scalable ATM switch and associated queue modules have been developed, implemented, and tested in the iPOINT testbed. As shown in Figure 1, iPOINT provides optical ATM networking at 800 Mb/s for two Sun SPARCStation 10/30s, a Sun SPARCStation IPX file server, a segment of the Blanca/Xunet Gigabit testbed, and a 400 Mb/s trunk interface. A photograph of the iPOINT switch, iPOINT queue modules, and the optical components is shown in Figure 2.

Each Sun SPARCStation transfers the data across its SBus to the Fore ATM host adapter. For the Internet protocols TCP or UDP, this data is in the form of variable

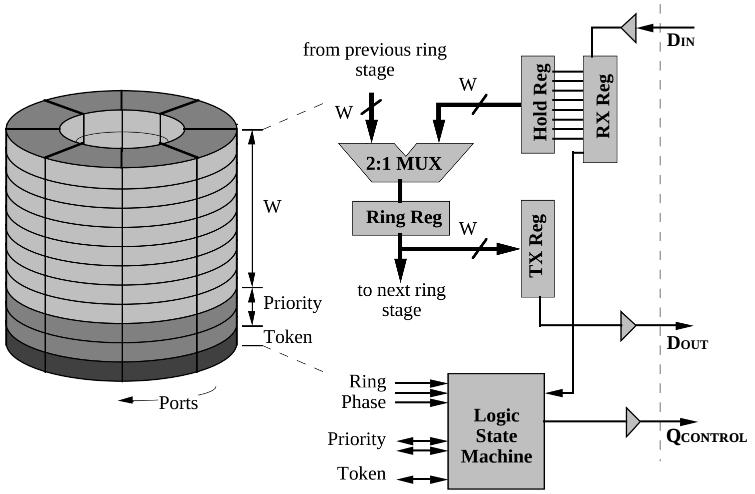

Figure 3: The Pulsar Concept

length IP packets. The host adapter encapsulates an IP packet in an AAL5 adaptation layer frame. The adapter fragments the frame into a series of individual ATM cells that are then optically transmitted from the host to the iPOINT switch via a fiber link.

The $\lambda = 1.3\mu$m optical signal from the workstation is first detected by the photoreceiver, deserialized, and decoded through the use of the AMD Taxi evaluation board [13]. The data are then presented to an iPOINT queue module using a parallel data path. Upon receiving the ATM cell header, the queue module translates the VPI/VCI and presents the switch with a destination vector, indicating to which outgoing ports the cell should be delivered. Upon reading the destination vectors, the switch instructs each queue to transmit or buffer the incoming cell. Cells that win the contention match are switched to their destination port(s) by the iPOINT switch module, while the cells that lose wait to recontend during the next cell cycle.

A The Pulsar Switch

The design of the iPOINT switch was derived from the theoretical analysis of the Pulsar switch [14]. The Pulsar switch uses parallel shift register rings to transfer data from incoming queue modules to outgoing destination port(s), as shown in Figure 3. The data, which is multiplexed onto the ring as a parallel word, is sequentially latched to each destination port. In a circular ring structure, individual transmitters need only move data from one source to the next receiver, rather than transmitting data across an entire backplane. To overcome memory bandwidth limitations, the Pulsar switch relys on input queuing at each port rather than using a shared buffer or output queues. It has been shown that a multiple-head input queue can approach the performance of an output-queued counterpart [15].

B iPOINT Core Switch Module

The prototype iPOINT core switch module is implemented using a Xilinx 4013 Field Programmable Gate Array (FPGA) [16]. This device has 192 I/O pins and a control logic block (CLB) delay of 5ns. The switch has four user ports, each providing 100Mb/s of simultaneous transmit and receive bandwidth. The trunk port has two 32-bit unidirectional data paths to interface at 400Mb/s. This asymmetric configuration may be used to build a hierarchy of switches that interconnects clusters of workstations. Each switch concentrates non-local traffic to a single asynchronously multiplexed data stream. Alternatively, the trunk port can drive two links, each operating at 200Mb/s. This configuration is useful for providing switched service to the home, as it allows for the construction of linear chains of switches. In either configuration, the aggregate throughput of the switch is 800Mb/s.

The iPOINT switch supports general multicast, allowing cells to be multicast to any permutation of the output ports. This feature is important for videoconferencing applications, where multiple users receive one copy of each video source, and for distributed computation, where the results of one processor's computation are used by a selected group of other processors.

The switch first issues a control strobe signal to each of the queue modules. Each queue module returns a valid signal and a destination vector. The valid signal indicates that an ATM cell has arrived and is ready to be switched. The destination vector is a bit-mapped field that indicates to which outgoing ports the cell should be delivered. A destination vector of $[d_4, d_3, d_2, d_1, d_0] = [0, 0, 1, 0, 1]$, for example, would indicate that the cell should be multicast to outgoing destination ports (2 and 0).

The hardware implements a round-robin algorithm so that each input port has a fair chance to contend for the output ports. The switch also supports a weighted round robin algorithm, so the switch bandwidth can be proportionally distributed at virtual circuit set-up time.

Data is transferred to the switch from each queue module using an eight-bit parallel unidirectional data path. In the switch, the data from each incoming port is switched

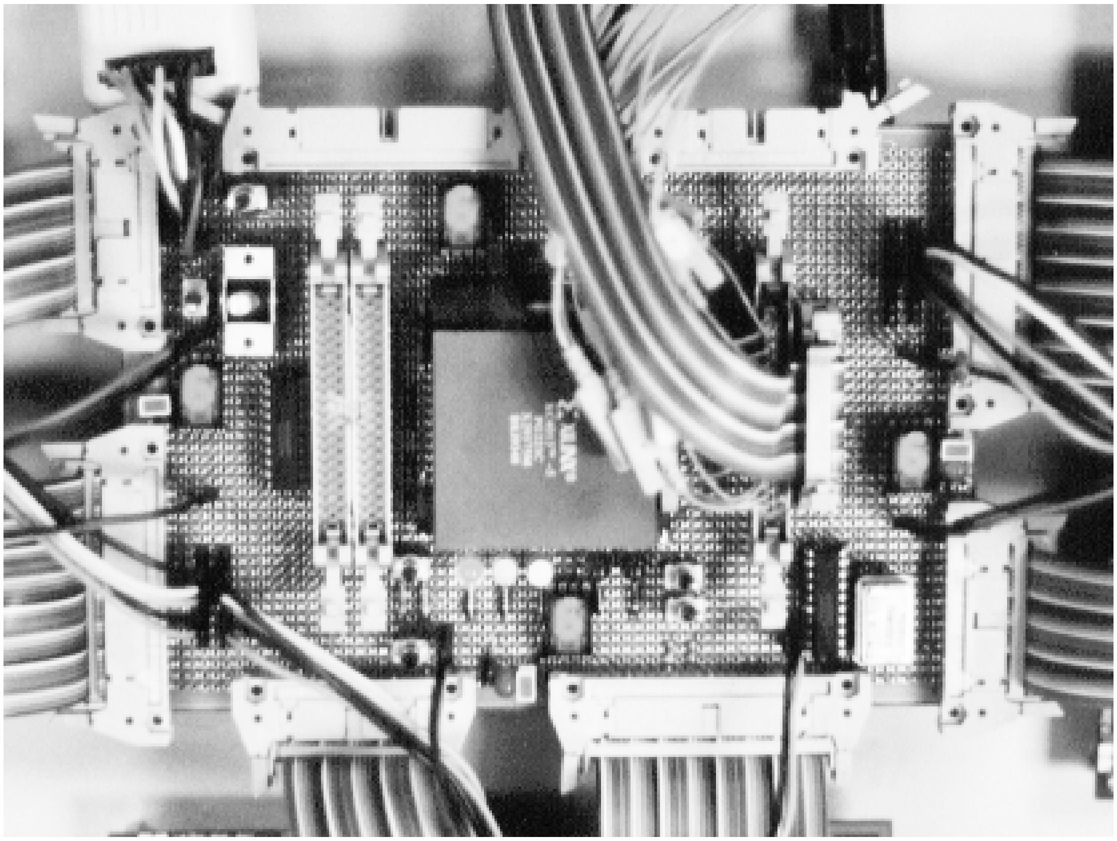

Figure 4: The iPOINT Prototype Switch

to each of the ports specified by the destination vector. Upon reaching the output port, data is transferred to the transmit module using another eight-bit parallel unidirectional data path. For multicast, the data is copied to both the destination port and to the next stage of the ring. A detailed photograph of the switch is shown in Figure 4.

C iPOINT Queue Module

The queue module resides between the optical data receiver and the iPOINT switch. The queue module performs the asynchronous interface logic, VPI/VCI translation, destination vector generation, VPI/VCI translation table updating, and input queue control logic, and has been implemented using a Xilinx 4005pc84-5 FPGA. The block diagram of this queuing module is shown in Figure 5.

Due to the limited memory capacity of the FPGA, an external silicon memory device has been employed for cell storage. Because the queue module currently uses a First-In-First-Out (FIFO) memory, head-of-line blocking limits the random unicast traffic throughput to 58% of that available for noncontending traffic [17]. Although this may

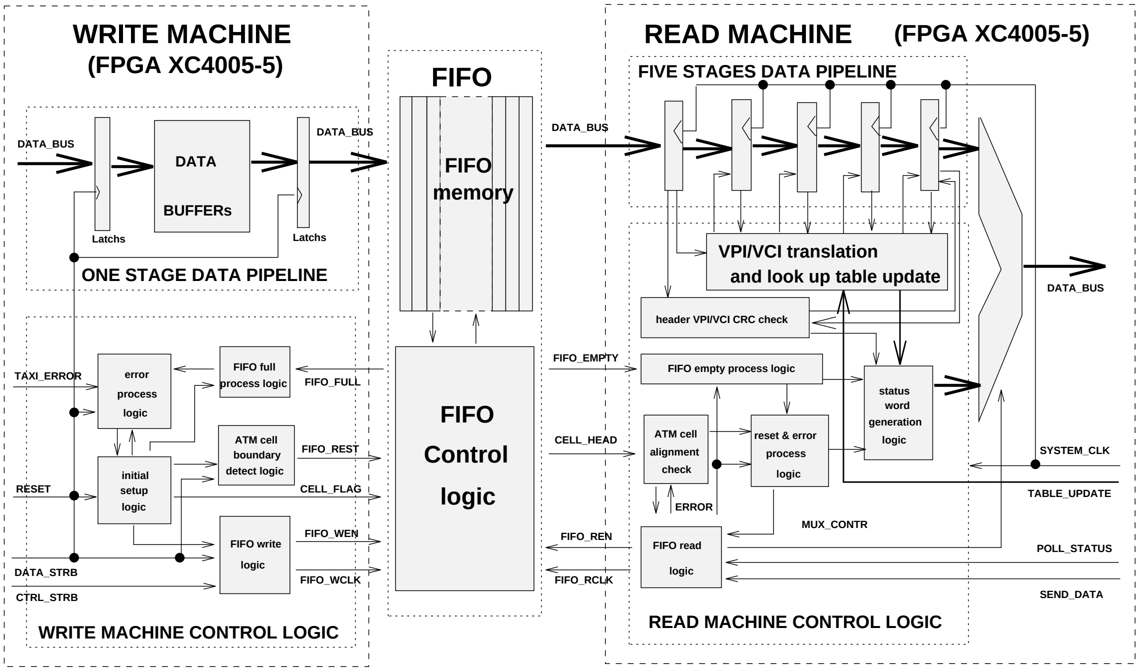

Figure 5: iPOINT Queue Logic

be acceptable for some configurations, improved performance can be obtained by using a general memory that maintains a per-VCI linked-list [18], which can be used to maintain a multiple-head input queue.

When the cell first arrives, it is stored in the memory and the header is processed by the queue module. Selected bits of the VPI and VCI fields are used to identify the destination vector by a hardware lookup into the VPI/VCI translation table located within each queuing module. By distributing the translation table, each port can process incoming cells in parallel rather than relying on a centralized translation table that would require sequential access. A dedicated port is connected to the VPI/VCI translation table so that the contents of this table can be dynamically updated by the iPOINT switch controller.

Upon receiving a control strobe, the queue module transfers the cell's destination vector across the parallel data path to the switch. If the switch accepts the cell, a data strobe is issued to the queue module and the data is transferred to the switch on successive clock cycles. If the data strobe is not issued, the cell remains in memory until the next cell slot. The design of the queuing module was complicated because the receiver's Phase Locked Loop (PLL) and the master clock used on the switch are not synchronized. Due to the multi-step cell processing procedures, pipeline techniques have

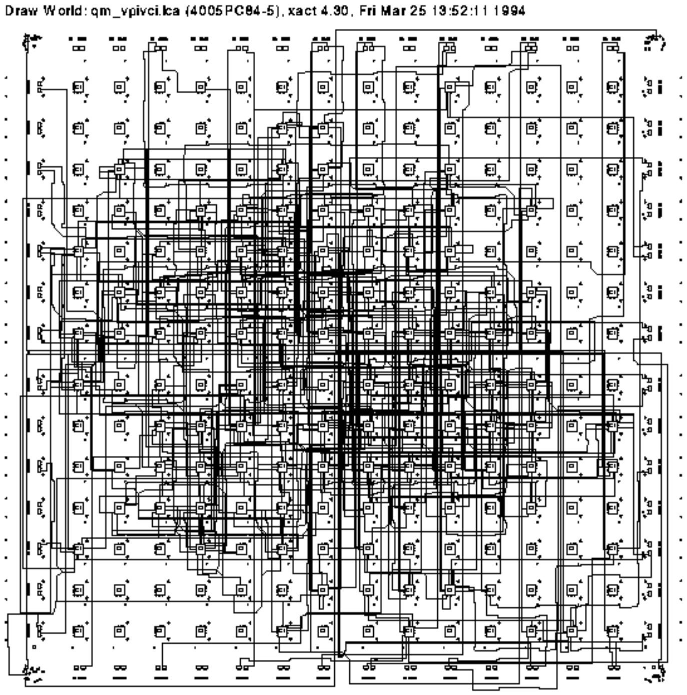
Draw World: qm_vpivci.lca (4005PC84-5), xact 4.30, Fri Mar 25 13:52:11 1994

Figure 6: The XC4005 Queue Module

been used extensively to maintain high throughput. Details of the iPOINT queuing module are described in [19]. The chip layout of the queue module is illustrated in Figure 6.

D Switch Controller

An Intel i486/DX2-66-based PC is used to control the creation and deletion of virtual circuits. The PC runs LINUX, a UNIX-like operating system. A process on the switch controller issues instructions to the PC interface adapter, which in turn transmits command strings to the appropriate queue modules or switch. The processor is used only for updating the VPI/VCI tables during call setup or teardown, not for VPI/VCI translation or ATM cell switching. The basic functions of an ATM switch controller are

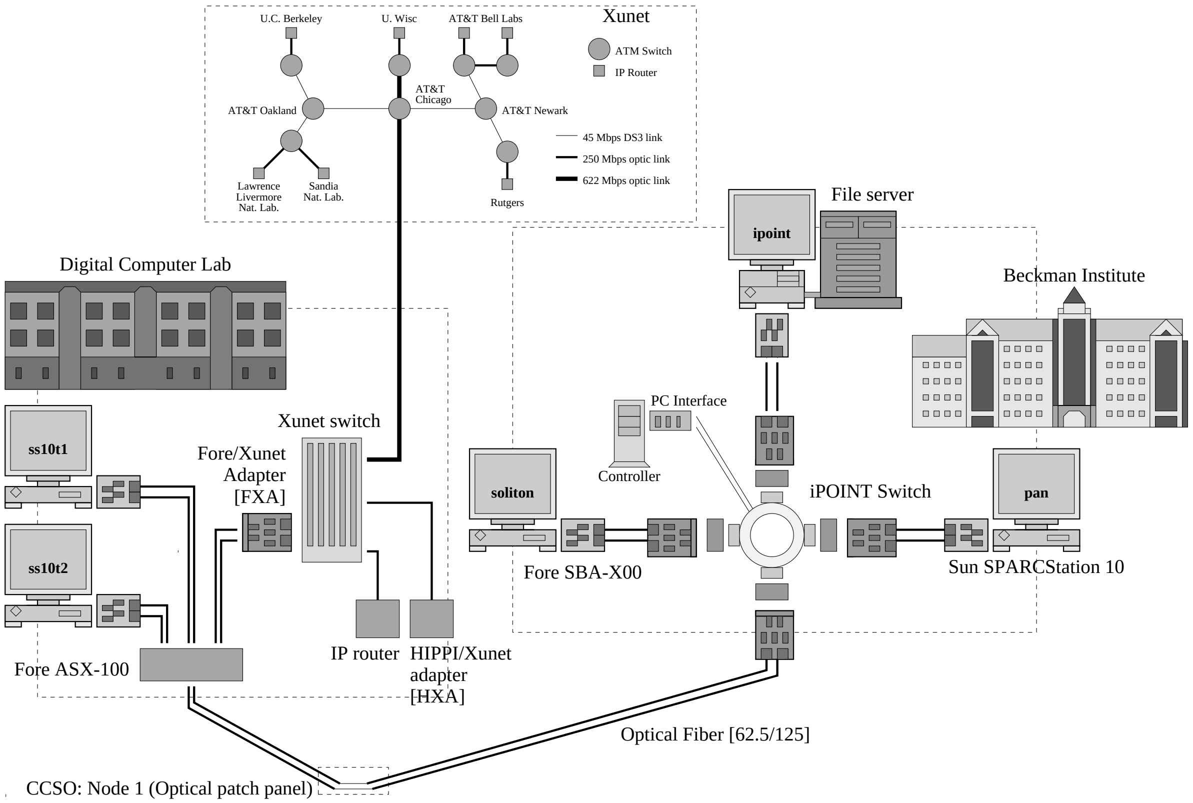

Figure 7: iPOINT interface to the Blanca Gigabit testbed

described in [20].

Using our existing switch control software, virtual circuit translations may be entered from the command line of the local console or from a remote terminal. A public domain software package called VINCE (Vendor Independent Network Control Entity), developed by the Naval Research Laboratory (NRL), provides ATM signaling and virtual circuit management functionality [21]. By running VINCE on our switch controller, the iPOINT switch can support the ATM signaling protocols Q.93b and Spans.

E Xunet Link

The iPOINT prototype switch is optically connected to the Blanca/Xunet nationwide gigabit testbed [22]. The Xunet testbed connects AT&T Bell Laboratories (Murray Hill, NJ), The University of Illinois at Urbana-Champaign, The University of California at Berkeley, The University of Wisconsin at Madison, Lawrence Livermore National Laboratories, Sandia National Laboratories, and Rutgers University using a combination of WDM 622Mb/s fiber links and DS3 links, as shown in Figure 7. ATM cells can be transmitted and received to and from any of the switches and routers in this network.

F Application-to-Application Performance through the iPOINT switch

To validate the iPOINT implementation, we measured the performance of application-to-application data transfer for both the local-area iPOINT and the wide-area iPOINT-XUNET network. Data was sent from one UNIX process running on one Sun SPARCStation to another UNIX process running on a different SPARCStation. The measurements included two experiments based on the Internet Protocols TCP and UDP (the protocols used for ftp, telnet, Mosaic, and most other existing network applications). The TCP/IP protocol provides reliable end-to-end communications (data is retransmitted if any cells are corrupted), while the UDP protocol only provides a best-effort delivery (data is dropped if any cells are corrupted or lost).

The well-known `ttcp` benchmark was used to measured the network throughput. Our tests were run between the buildings over the short-haul optical fiber and over the long haul links shown in Figure 7. The tests involved the iPOINT switch, a commercial Fore ASX-100 switch, and the Xunet switch.

As a control, we first connected the two Sun SPARCStation 10/51s with SBA200 interfaces in a point-to-point configuration, as shown in Figure 7. For the experiment, we routed the data through the Fore and iPOINT switches, and finally routed the data through the Fore, iPOINT, and XUNET switches.

Our TCP results are shown in Figure 8. The TCP/IP window size determines the amount of outstanding data that may be transmitted before receipt of an acknowledgement. With smaller window sizes, switching delays restrict throughput. The graphs for all configurations are similar and show almost no degradation with the introduction of the iPOINT switch.

For the UDP experiments, a SPARCStation 10/30 in Beckman Institute transmitted data across the campus to a SPARCStation 10/51 in DCL, as shown in Figure 7. First, the data was routed directly between the iPOINT and Fore switches. Then the data was routed through the Fore, iPOINT and XUNET switches. These tests formed a set of local area network performance measurements. Finally, using a combination of the iPOINT switch and the wide-area XUNET network, we routed data through the 622 Mb/s optical link to the XUNET switch in Chicago and through two 622Mbp/s links to the XUNET switch in Wisconsin.

Our UDP results are shown in Figure 9. The receive bandwidth is shown as a function of buffer length. We noted that no data was dropped in any of our experiments. We see that the software overhead of processing the UDP/IP protocol severely degrades the thoughput of short messages, suggesting the use of native-mode ATM applications that

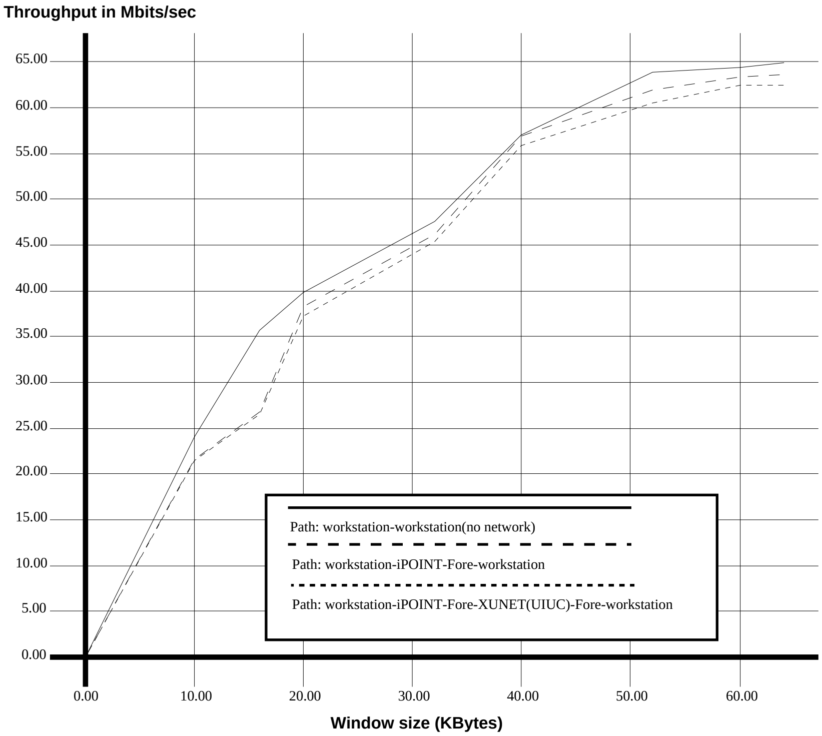

Figure 8: TCP/IP Throughput vs. Window Size

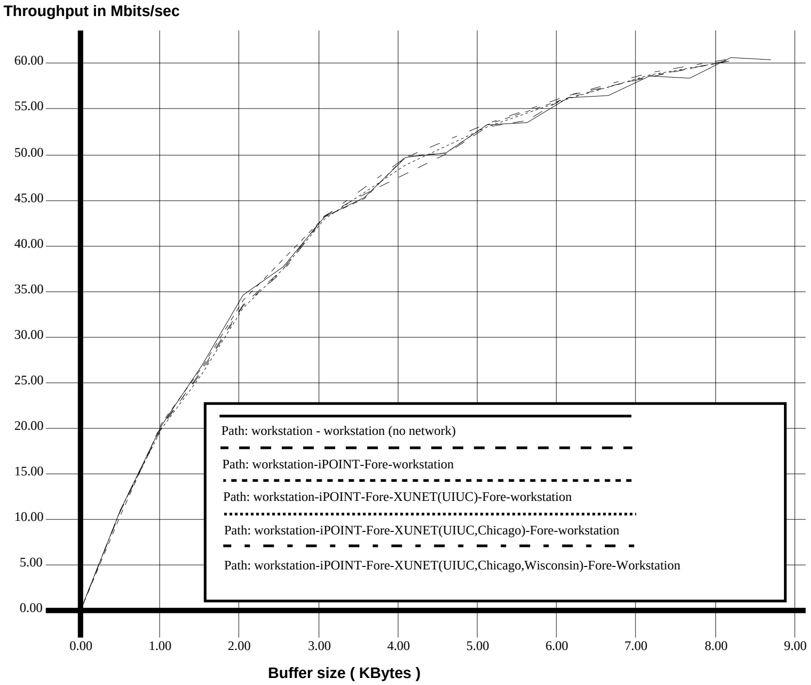

Figure 9: UDP/IP Throughput vs. Buffer Size

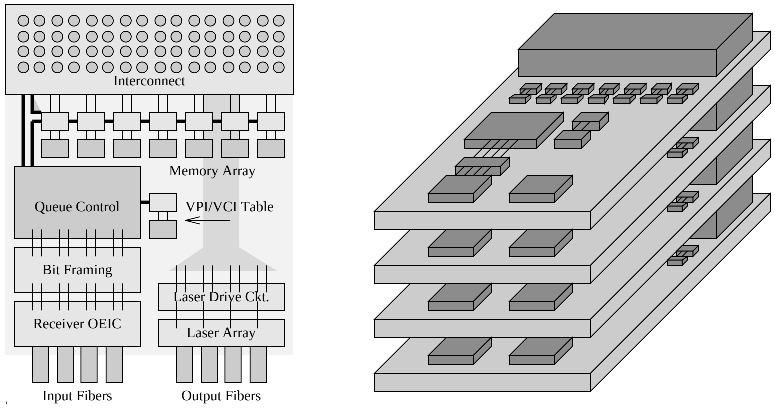

Figure 10: Scalable iPOINT Queue/Switch

could bypass much of the workstation's overhead. For all experiments, we note that the bandwidth and window size was limited by the software protocol processing performed by the host, not by the ATM network.

IV Scalability

The iPOINT architecture scales to provide multi-gigabit per second per port throughput. The scalable architecture distributes the switch function over multiple modules rather than concentrating the switch in a monolithic integrated circuit (which would limit the bandwidth by the pin count and I/O power constraints). Separate components implement each bit slice of the Pulsar ring (a vertical slice of Figure 3). A hybrid multichip implements each queue module with one ring stage placed on each queue board, as shown in Figure 10.

For multi-Gb/s per port throughput, we are developing a GaAs implementation of the distributed iPOINT switch. Each port operates at 4 Gb/s, using a 4×1 fiber array with each channel operating at 1 Gb/s. Although the device is externally clocked at 1 GHz, the internal control logic needs to operate only at a small multiple of the cell arrival rate (1/106ns). With the pipelined design we developed for the FPGA switch and queue modules, we can implement most of the control logic as an embedded gate array operating at 100 MHz. Full-custom layout is required only for the data path and control signal distribution. The input queue can use an external, cell-wide array of silicon RAM devices, with a modest memory-cycle time of 50ns (twice the rate of cell arrivals).

Scaling of the VPI/VCI translation tables is achieved by parallelism and pipeline techniques. Since each input queue independently performs the translation functions on incoming cells, an N-port switch may concurrently process N translations. Each translation, in turn, can be performed by two pipelined memory operations. The first memory access, indexed by the VPI, either provides the outgoing VPI and destination vector, or it provides an offset into the VCI table. The second memory access, indexed by the value of the VCI plus the offset, provides the outgoing VPI, VCI, and destination vector. The sizes of the VPI/VCI tables need only be proportional to the maximum number of virtual paths and virtual circuits that the switch is designed to support.

The design of the iPOINT switch enables the use of point-to-point optical backplane interconnect and waveguide array technology. The input queue has eliminated output-port contention and translated the VPI/VCIs before the cells reach the backplane. Since at least one optoelectronic conversion is needed, the wavelength and optical power levels used on the backplane can be independent of that chosen for the links. While it is clear that $\lambda = 1.55\mu$m devices are required for long-haul links, is likely that short-wavelength ($\lambda = 0.85\mu$m) optoelectronic devices will have utility both for the short-haul optical links and for backplane data transfer. The Pulsar shift register ring lends itself naturally to an optical implementation, as the propagation delays of the waveguide cause signal delays that occur within a relatively fixed, finite time interval. The shift register ring can be implemented by the combination of the propagation delay and by the use of a global, pulsed, optically-distributed, clock signal to maintain bit framing at each ring stage.

A iPOINT Optical Components

Several custom optoelectronic devices have been specifically designed for the iPOINT testbed and will be employed for the trunk port of the existing switch and for all ports of the high-speed iPOINT switch. While iPOINT currently operates at 1.3 μm, due to the short-distance nature of the optical interconnects under consideration, we will implement these devices in the short-wavelength, 0.85-μm range. This will allow us to utilize the more mature and less expensive GaAs, rather than the InP, materials system. Since light at 0.85 μm is limited by fiber dispersion, we have chosen a maximum link distance of 0.5 km as an the target distance for optical interconnects within one site. Long-wavelength devices will, however, eventually be substituted on ports that connect to long-distance links such as the Blanca/Xunet gigabit testbed. Currently under fabrication are a four-channel photoreceiver array as well as a four-channel transmitter array; once completed, they will be integrated with the GaAs gate array.

The photoreceiver is a monolithically-integrated, four-channel MSM-MESFET array and is being fabricated in a 0.6-μm MESFET process. Each of the array elements contains both a transimpedance preamplifier as well as a postamplifier designed with enough gain to produce a ± 800 mV differential swing at the output. The MSMs were sized a rather large 75 × 75 μm^2 in order to facilitate optical coupling from a multimode, 50-μm core fiber. The center-to-center spacing between the MSM detectors was set to 250 μm; this gap was purposely made large not only to allow ample space for the 125 μm cladding of the optical fiber, but also to reduce crosstalk between channels. The photoreceiver is designed to have an output impedance of 50 Ω, a sensitivity of -20 dBm and an operating bit rate of at least 1 Gb/s.

The transmitter, on the other hand, is a four-channel, hybridly-integrated, laser driver/laser array combination. The laser driver is being fabricated in the same 0.6-μm MESFET technology as the photoreceiver and was designed to run off of a 5 V power supply. The electrical inputs to the driver are designed to match the ± 800 mV differential signal levels mentioned earlier. The circuit operates by sinking both a bias current and a signal current through the laser for high-speed direct modulation and was designed to sink enough current to produce optical output powers ranging from 0.25 mW to 1.5 mW. The laser array consists of four GaAs double quantum-well lasers, each operating at the 0.85-μm wavelength with an average threshold current of about 10 mA and a center-to-center spacing of 250 μm. The transmitter integration will be accomplished by hybridly wire bonding the laser drivers to the laser array; while a monolithic implementation is clearly desirable, we have adopted this intermediate approach because of the considerable difficulty involved in integrating the laser and transistor structures on a common substrate. At the 1 Gb/s target speeds, we do not expect to be limited by the bond parasitics.

V Conclusions

The current iPOINT testbed switch has an FPGA implementation and provides the same throughput as 80 Ethernets or 8 FDDI rings. Although throughput of the prototype switch is limited compared with some existing and proposed ATM switches, the switch design is scalable and can be implemented in GaAs technology to provide multi-Gb/s per port throughput. In addition to the optical components used for every port of our switch, the design enabled the use of optical components for the implementation of the optical Pulsar ring.

Both the iPOINT switch and the queue modules are implemented using Xilinx Field Programmable Gate Array (FPGA) technology. Although the FPGA doesn't provide the performance of a full-custom circuit, automated design tools exists to transform the Xilinx design into a gallium arsenide gate array. The FPGA technology required careful attention to timing delays and critical paths; as such, the iPOINT queue module and switch make extensive use of pipelining, critical path avoidance, and parallelism. We have found the circuits to be robust to the routing variations that occur after each design iteration.

Our experience has proven the importance of packaging for optical and optoelectronic devices. Our scalable ATM switch design utilizes hybrid multichip modules, allowing the integration of optical components, logic elements, and memory. Further, the receiver OEIC helped to minimize the component count and cost, as the receiver electronics were combined on the same substrate as the MSM photodetector.

A balance of optical and electrical components is required for the design of high performance ATM switches. The need for cell queues in an ATM switch mandates an optical/electronic conversion. Silicon memory has, and will continue to, provide an economical solution for mass data storage. Optical components have long proven effective for circuit-switched applications. By optimizing the optical-electrical components (the laser array, laser drivers, and receiver array), it is possible to build multi-gigabit packet switches that can provide the bandwidth and service requirements for the upcoming "Information Superhighway".

References

[1] C. Partridge, *Gigabit Networking.* Addison-Wesley, 1994.

[2] J. W. Lockwood, C. Cheong, S. Ho, B. Cox, S. M. Kang, S. G. Bishop, and R. H. Campbell, "The iPOINT testbed for optoelectronic ATM networking," in *Conference on Lasers and Electro–Optics*, (Baltimore, MD), pp. 370–371, 1993.

[3] G. J. Murakami, R. H. Campbell, and M. Faiman, "Pulsar: Non-blocking Packet Switching with Shift-register Rings," in *Computer Communications Review*, vol. 20.4, pp. 145–155, September 1990. SIGCOMM '90 Conference Proceedings.

[4] J. B. Lyles and D. C. Swinehart, "The emerging gigabit environment and the role of local ATM," *IEEE Communications*, pp. 52–58, Apr. 1992.

[5] P. A. Steenkiste, "A systematic approach to host interface design for high-speed networks," *Computer*, vol. 27, pp. 47–57, May 1994.

[6] R. Händel and M. N. Huber, *Integrated Broadband Networks: An Introduction to ATM-Based Networks.* Reading, Massachusetts: Addison-Wesley, 1991.

[7] D. J. Blumenthal, K. Y. Chen, J. Ma, R. J. Feuerstein, and J. R. Sauer, "Demonstration of a deflection routing 2×2 photonic switch for computer interconnnects," *IEEE Photonics Technology Letters*, vol. 4, pp. 169–173, Feb. 1992.

[8] T. Kozaki, Y. Sakurai, O. Matsubara, M. Mizukami, M. Uchida, Y. Sato, and K. Asano, "32 x 32 shared buffer type ATM switch VLSIs for B-ISDN," in *ICC '91*, pp. 711–715, IEEE, 1991.

[9] Y. Yeh, M. G. Hluchyj, and A. S. Acampora, "The Knockout switch: A simple, modular architecture for high–performance packet switching," *IEEE Journal on Selected Areas in Communications*, vol. SAC-5, pp. 1274–1283, Oct. 1987.

[10] P. E. Green, "An all–optical computer network: Lessons learned," *IEEE Network Magazine*, pp. 56–60, Mar. 1992.

[11] F. J. Janiello, R. Ramaswami, and D. G. Steinberg, "A protype circuit-switched multi-wavelength optical metropolitan-area network," *IEEE Journal of Lightwave Technology*, pp. 777–782, May 1993.

[12] R. Gidron, S. D. Elby, A. S. Acampora, J. B. Georges, and K. Y. Lau, "TeraNet: A multi gigabit per second hybrid circuit / packet switched lightwave network," tech. rep., Center for Telecommunications Research, Columbia University, 1991.

[13] AMD, *TAXIchip (DC–CAB) Data Checker Board User's Manual*, 1991.

[14] G. J. Murakami, "Non-blocking packet switching with shift-register rings." Ph. D. Dissertation, University of Illinois at Urbana–Champaign, 1991.

[15] M. Akata, S. Karube, and S. Yoshida, "An input buffering ATM switch using a time–slot scheduling engine," *NEC Research and Development*, vol. 33, pp. 64–72, Jan. 1992.

[16] Xilinx, Inc., *The Programmable Logic Data Book*, 1993.

[17] M. J. Karol, M. G. Hluchyj, and S. P. Morgan, "Input vs. output queueing in space division packet switching," *IEEE Transactions on Communications*, vol. Com-35, pp. 1347–1356, Dec. 1987.

[18] C. R. Kalmanek, S. P. Morgan, and R. C. Restrick, "A high–performance queueing engine for ATM networks," in *ISS*, 1992.

[19] H. Duan, J. W. Lockwood, and S. M. Kang, "FPGA prototype queueing module for high performance ATM switching," Submitted to *ASIC'94,* (Rochester, NY), Dec. 1994.

[20] R. Campbell, S. Dorward, A. Iyengar, C. Kalmanek, G. Murakami, R. Sethi, C.-K. Shieh, and S.-M. Tan, "Control software for virtual-circuit switches: Call processing," Lecture Notes in Computer Science, Springer, 1992.

[21] M. P. Eric Hoffman, Allison Mankin, "VINCE: Vendor independent network control entity." Available via Internet anonymous ftp from `hsdndev.harvard.edu` as `/pub/mankin/vince.ps.Z`, Mar. 1993.

[22] A. G. Fraser, C. R. Kalmanek, A. E. Kaplan, W. T. Marshall, and R. C. Restrick, "Xunet 2: A nationwide testbed in high–speed networking," in *INFOCOM*, pp. 582–589, 1992.