Are Crossbars Really Dead? The Case for Optical Multiprocessor Interconnect Systems

Andreas G. Nowatzyk
Sun Microsystems Computer Corporation, agn@acm.org
Paul R. Prucnal
Princeton University, prucnal@ivy.princeton.edu

Abstract

Crossbar switches are rarely considered for large, scalable multiprocessor interconnect systems because they require $O(n^2)$ switching elements, are difficult to control efficiently and are hard to implement once their size becomes too large to fit on one integrated circuit. However these problems are technology dependent and a recent innovation in fiber optic devices has led to a new implementation of crossbar switches that does not share these problems while retaining the full advantages of a crossbar switch: low latency, high throughput, complete connectivity and multicast capability. Moreover, this new technology has several characteristics that allow a distributed control system which scales linearly in the number of attached nodes.

The innovation that led to this research is an optical ***and****-gate that can be used to demultiplex multiple high speed data streams that are carried on one common optical medium. Optical time domain multiplexing can combine the data from many nodes and broadcast the result back to all nodes. This paper discusses OTDM technology only to the extent necessary to understand its characteristics and capabilities. The main contribution lies in the description and analysis of interconnect architectures that utilize OTDM to achieve a level performance that is beyond electronic means. It is expected that cost-reduced OTDM systems will become competitive with the next generation of interconnect systems.*

1 Introduction and Motivation

In the absence of implementation constraints, the ideal processor interconnect system is the crossbar switch, because it provides full connectivity at uniformly low latencies and peak throughput that is simply the bandwidth of a node interface multiplied by the number of nodes in the system. Crossbar switches can implement any permutation and support arbitrary multicasting. Unfortunately, crossbar switches implemented by electronic means are costly and do not scale well, which is largely due to the fact that they require $O(n^2)$ switching elements for n nodes. Crossbar switches with up to 32 nodes can be implemented on one integrated circuit [1,2], but this does not solve the scaling problem because of the limited number of data paths that can be connected to one chip. Furthermore, it becomes increasingly difficult to control a crossbar as its size grows.

This situation led to the development of multistage electronic switching fabrics, where many smaller switching elements are interconnected in certain topologies to approximate the functionality of large crossbar switches. Many years of intense research have culminated in interconnect systems that support 1000 (or more) nodes and achieve up to 100 Gbyte/sec bisection bandwidth with mean latencies of about 150 ns (Cray T3D, 0-load latency). However such systems do not achieve full crossbar functionality: the time for a message to traverse the interconnect is variable and depends on many factors, so that the sending node does not know when its data will arrive. Besides lacking a globally visible event ordering, switching fabrics generally do not support broad- or multicasting.

A recent innovation in optics, the Teraherz Optical Asymmetric Demultiplexer (TOAD) is going to challenge electronic interconnect systems [3]. The TOAD device allows the construction of large, high performance crossbar switches in a manner that scales linearly in the number of attached nodes. In the laboratory, a TOAD based data transmission experiment has shown performance characteristics that would allow the construction of a system with a bisection bandwidth of 250 Gbits/sec. The technological limits indicate that operation at rates of more than 5 Tbits/sec are conceivable. Therefore the potential performance of a TOAD based interconnect system exceeds that of the best current electronic switching fabrics by a factor of 10. Moreover, the TOAD based interconnect system offer several capabilities that are not practical in electronic switching fabrics and require fewer components, which should ultimately lead to much lower costs.

This paper will give a brief description of the operation of an optical crossbar switch and its capabilities. It then describes and analyzes a family of architectures that exploit the specific capabilities of such switches and compares these systems to conventional electronic switching fabrics. Finally, the remaining challenges and problems are addressed.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association of Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.
ISCA '95, Santa Margherita Ligure Italy
© 1995 ACM 0-89791-698-0/95/0006...$3.50

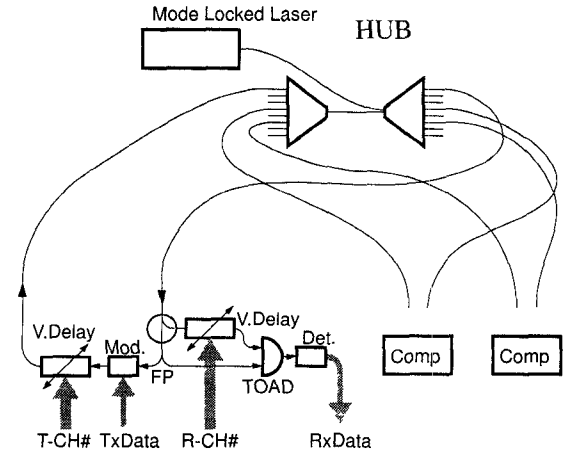
2 Optical Time Division Multiplexing

One way of building a crossbar switch is to take the data that each attached node is trying to send and multiplex it on one common medium that is broadcast to all participating nodes. Each node has a receiver that is capable of extracting its messages out of the combined broadcast. This approach is similar to that of a bus, however the bandwidth of a bus usually matches the node interface speed, while in the case of a shared medium switch, the interface bandwidth is small compared to the total transmission capacity.

It is obvious that this structure is not an interesting approach for an electronic implementation because there is no electronic medium that can broadcast several hundred Gbits/sec. However, an optical fiber has a transmission capacity of more than 10,000 Gbits/sec, which is more than 10x the entire bisection bandwidth of any switching fabric in existence. Given that there is no possibility to match this speed electronically through one interface, the photonics community has concentrated on ways to multiplex multiple, independent electronic data streams onto one fiber, usually in the context of long distance telecommunication. There are fundamentally two different approaches, optical wavelength division multiplexing (OWDM) and optical time-division multiplexing (OTDM) [9]. For the purpose of high performance computer interconnect systems, OTDM is more interesting, because the synchronicity inherent in an OTDM system allows efficient control and arbitration operations. These capabilities will be discussed in detail below. Another reason that currently favors OTDM systems is that the TOAD device allows their implementation with off-the-shelf optoelectronic components while OWDM systems require special lasers with controlled frequencies and special filters to extract channels. Practical lasers suitable for OWDM currently support only a small number of channels (<50) and tunable filters that are capable of selecting the data from a particular OWDM channel are slow to change wavelengths because they typically involve mechanical components (piezo-tuned etalons or surface acoustic wave gratings).

Figure 1 outlines the basic structure of the optical hardware that is the basis for this paper [6]. At the center of the switch is a mode-locked, pulse compressed laser that emits a continuous steam of very short light pulses. The pulse rate is equal to the bit rate that is used by each attached node, for example 1.3 Gbit/sec. The pulse-width of each light burst is much smaller, for example 1ps. This laser will act as a central clock source.

FIGURE 1 : Basic OTDM System



The pulses of the mode locked laser are fed into a passive power splitter that distributes the light equally to all attached nodes. The optical components at the node interface feed the framing pulses to a modulator that either lets the pulse pass through or absorbs it, depending on whether the node wants to send a '1' (= light) or a '0' (= no light). The transmitted data is then sent through a programmable delay element that determines on which channel the data will appear. The output of the transmitters from all nodes is carried back to the central hub where the light is combined and subsequently distributed to all attached nodes.

The receiver section of each node uses a device that separates the data from the framing pulses (for example by virtue of their polarization) and feeds the received data through a programmable delay element that selects the receive channel. The framing pulse is then used to open an optical *and*-gate that isolates the data of the selected channel from all other data. The resulting bit stream is then sent to a photodetector that converts that data back to the electronic domain. The important feature of this transceiver is that all electronic devices (modulator, detector and delay elements) operate only at the channel bit rate.

The key element of the OTDM system described above is the optical *and*-gate. Lacking a fast and practical *and*-gate prevented previous OTDM switch proposals [6] from achieving compelling performance levels. The recent invention of the TOAD device (described below) changes this situation [3].

FIGURE 2 : Optical Time Division Multiplexing

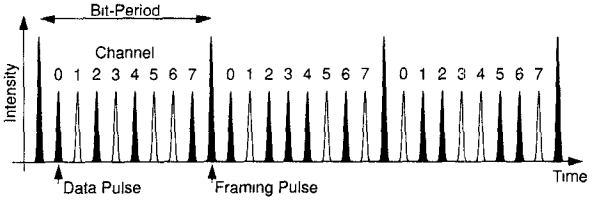
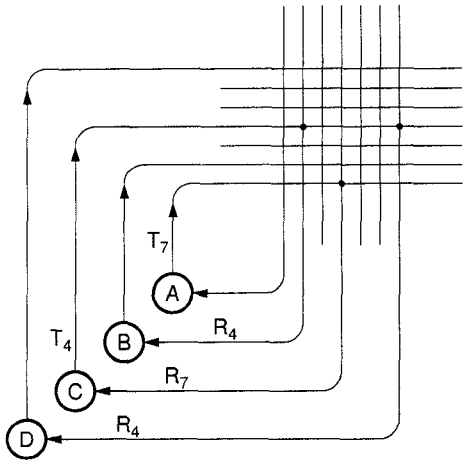


Figure 2 shows the signal that is sent to the receiver of each node. Each node can tune its receiver to any given channel. In particular, nodes are also capable of receiving

their own transmission. This capability allows a node to compensate for the signal propagation delay between its transmitter and the central star-coupler, such that the transmissions of all nodes line up properly. Furthermore, transmitting nodes can detect the presence of multiple transmissions on the same channel because the light from all sources is added, so that the photodetector actually receive the logical *or* of colliding transmission.

FIGURE 3 : Logical Crossbar Structure



Given a collection of nodes, the channel numbers serve as a form of destination address designating the outputs of the switch. For example, the transmissions on channel 7 are normally received by node C in Figure 3. However, this channel to node assignment is not necessarily rigid, rather it is possible to agree on meta-destinations that causes data to be sent to multiple nodes. In the example, all transmissions on channel 4 are received by nodes B and D.

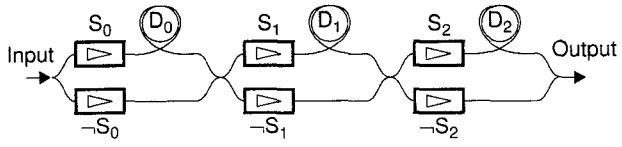
2.1 Ultrashort-Pulse Lasers

There are several practical lasers that can produce extremely short light pulses. For example, Irl Duling's figure 8 laser[5] uses a fiber-optic ring cavity with a section of erbium-doped fiber that serves both as the lasing medium and as an intensity-dependent optical element that causes the light racing around the ring to be concentrated into short bursts. This device has produced light pulses as short as 90 fs (1 fs = 10^{-15} sec.) at repetition rates of up to 10 G pulses/second. Because the light energy is compressed into such a short time, these light bursts are quite intense: a 1 mW pulse laser with a pulse width of 100 fs and a repetition rate of 1 GHz, has a peak power of about 10 KW, which provides good framing pulses even if the output is distributed among many nodes.

2.2 Rapidly Tunable Delay Elements

Both transmitter and receiver rely on a delay element to select which channel to operate on. Given that this delay needs to be stable to less than the channel spacing time, it cannot be realized electronically.

FIGURE 4 : Optical Delay Element



Binary controlled delay elements can be implemented by cascading N switches that direct the light to either of two signal paths with delays that differ by $t_{bp}*2^{-(N-n)}$, where t_{bp} is the bit period and n the index of the control bit [17]. For example, if the system operates at 1 Gbit/sec, the fiber in D2 of the 3 stage delay element in Figure 4, needs to delay light by 0.5 ns, which corresponds to about 9cm. The incoming light is equally split in two parts that are fed to the two solid state amplifiers (SOA) S_2 and $\neg S_2$. SOAs are essentially laser diodes without mirrored ends that have fibers attached to each side. When current is supplied, light entering on one side is amplified before it exits the other end. Without current, the incoming light is attenuated. Depending on bit 2 of the control signal, only one SOA is turned on. SOAs can be turned on or off within about 1 ns, so that the delay element can switch very quickly to a different channel. The delay element also provides some amount of light amplification.

For the large system, the precision of the required delay elements becomes demanding and implementations that use discrete fibers will become costly. However, it is possible to integrate the delay elements via waveguides that are formed via titanium doping of a glass substrate. This allows precise control of the geometry using lithography methods borrowed from the semiconductor industry. Optical integration can also be used for the couplers and elements of the TOAD, as well as the high fan-in/out couplers of the hub [18], resulting in smaller and less expensive nodes.

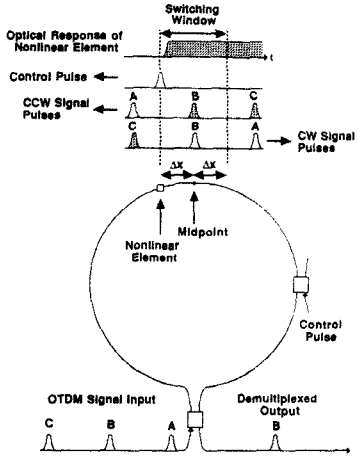
2.3 Controlling Delays

The distance between the hub and the nodes will vary in an operational system because it is not practical to control the fiber length to sub-mm precision. Furthermore, temperature changes and mechanical stress on the fiber may change the propagation delay beyond the channel spacing. To compensate for these effects, each node needs to monitor the delay from its transmitting element to the hub by periodically looking at its own transmission. A piezoelectric fiber stretcher dithers the delay slowly by a fraction of the channel spacing. This results in a slight modulation of the average light intensity that is observed by the receiver. By relating the intensity change to the applied delay, the servo electronics that controls the fiber stretcher keeps the transmitter properly centered on its time-slot relative to the framing pulse. This servo mechanism does not impact ongoing transmission, because the intensity variations are small compared to the digital signal. It turns out that the arbitration mechanism described later, requires that the own transmissions are periodically received. Delays that exceed the channel spacing are controlled digitally.

2.4 The TOAD Device

Building an optical *and*-gate is difficult because light signals tend not to interact with each other. In order to interact, some nonlinear medium is required. Past attempts at building such gates either required very high power levels or very long interaction lengths[4], neither of which were practical in a computer network interface. The TOAD is a new device that is both compact and does not require high power levels. Unlike other attempts at optical and-gates, the TOAD does not try to function like a conventional gate, that can switch quickly at a rate comparable to the switching speed. Rather the TOAD can switch on/off only once before it needs to recover for about one bit period.

FIGURE 5 : THz Optical Asymmetric Demultiplexer



The TOAD consists of a fiber loop connected to a symmetric 2x2 coupler (Figure 5). The other side of the coupler is connected to the signal input and to the photodetector of the receiver. Without the other components, light from the input is split by the coupler into 2 equal parts that traverse the loop in opposing directions. Because each of the two signal components will have traversed exactly the same distance when they meet again in the coupler, constructive interference occurs so that all light is reflected back into the input fiber and no light reaches the detector. This loop-mirror contains a nonlinear element (a SOA) that is located slightly off-center from the half way point. If the control pulse, that is injected elsewhere into the loop, reaches the nonlinear element, it will change its index of refraction such that light traversing it before and after the control pulse will experience slightly different propagation delays. This difference in delay will change the interference condition such that light is directed into the detector for a duration that corresponds to twice the distance of the nonlinear element to the midway point.

FIGURE 6 : An Experimental TOAD

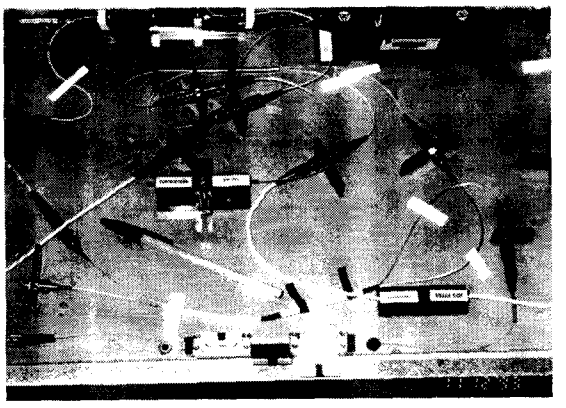
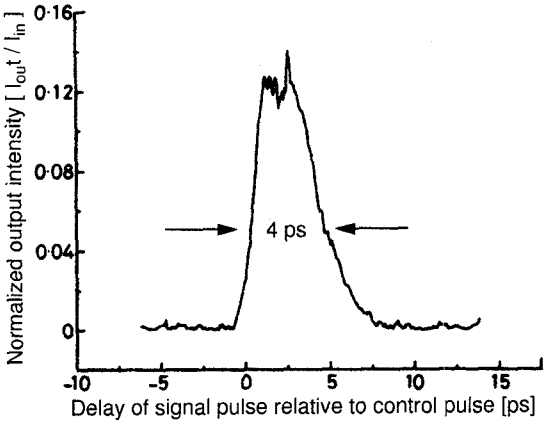


Figure 6 shows an experimental TOAD assembled out of commercially available components in the Princeton Laboratory [7]. Figure 7 shows measured results of the output intensity of the TOAD as a function of the relative positions of the signal and control pulses, where Δx is 100 μm. The data shows that the TOAD passes light for about 4ps, which corresponds to a switch capacity of 250 Gbit/sec.

FIGURE 7 : TOAD Performance



2.5 OTDM Transceivers

Besides the optical components outlined in Figure 1, there is the digital circuitry that is required for a functional OTDM computer interface. Data must be supplied to the transmitter synchronous to the arrival of the framing pulses. This can be achieved by having a separate detector that senses the framing pulses. However this detector would have to operate at twice the transmission data rate. A simpler and more cost-effective method is to rely on the received signal. Since the TOAD is gated by the framing pulses, data is being received with a fixed phase relation to the outbound data stream. Hence the recovered clock from the receiver may be used to control the transmitter, provided that the data encoding method provides sufficient transitions for a conventional clock recovery scheme.

Since data is sent and received strictly synchronously, there is no need to re-establish bit or frame alignment when the receiver switches channels. Hence preamble sequences

common to conventional bit serial communications methods are unnecessary. All of this circuitry can be integrated into one CMOS ASIC, using recently developed serial link technology[8].

The most costly components of an OTDM transceiver are the two delay elements, due to the number of SOAs. However, recent advances in the production of laser arrays and in integrated optics has led to the belief that the optical assembly of an OTDM transceiver could be manufactured in volume for less than $1000. Integration could also reduce its size to about the size of a matchbox.

2.6 Summary: Optical Hardware

The OTDM hardware described above has been demonstrated in the laboratory in a configuration that could supports about 250 channels, each operating at about 1 Gbit/sec[7]. The number of channels is limited by the pulse width of the laser, the available power-levels, photo detector sensitivity and the dispersion of the interconnecting fibers. Ultrashort pulse lasers are practical to support about 5000 channels [5]. The power budget analysis indicates systems with 1000 nodes or more are feasible with current technology[6]. The fiber dispersion will limit the physical distance between the central star-coupler and the attached nodes to about 100m, which is more than sufficient for multiprocessor interconnect systems and ultra-fast local area networks that could support networks of workstations[10] with supercomputer class communication bandwidth. The bandwidth for each channel is limited by the repetition rate of the laser and by the TOAD recovery time. A figure-8 laser has been demonstrated to operate with a repetition rate of up to 10 GHz, while the TOAD switching rate may approach 50 GHz[1].

TABLE 1 : OTDM Characteristics

Property	Current	Future Potential
# of Nodes	250	5000
Bandwidth per Node[a]	1 Gbits/sec	50 Gbits/sec
Channel select time	< 5ns	< 2 ns
Latency	~50 ns + fiber delay	~20 ns + fiber delay
Node to hub distance	< 100 m	< 100 m
Concurrent Send&Rcv.	Yes	Yes
Cost / Node	~$20000	<$1000

a. Using a 1 channel receiver. Receivers that use multiple TOADs can increases this bandwidths by sending multiple bits during one bit time.

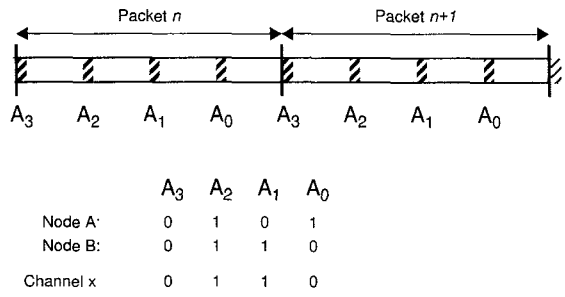
Table 1 summarizes the characteristics of the current lab prototype and the potential of future implementations. The latency is largely determined by the time it takes to serialize data at the sender and deserialize data at the receiver. The actual time to traverse the switch is just the speed of light delay inside the fiber, typically 5.5ns per meter, It should be noted that the current cost per node refers to precision laboratory equipment that is manufactured in very low quantities and hence is much more expensive than industrial components.

1. British Telecom has just demonstrated a 40 Ghz repetition rate TOAD.

3 OTDM Crossbar Control and Arbitration

Given the interconnect hardware described in the previous section, a computer interconnect architecture that is based on OTDM should be able to utilize the switch efficiently and in a manner that preserves the low and uniform latency. Controllers for small, electronic crossbar switches are fairly simple can be co-implemented with the actual switch[1]. However, this approach does not scale to 250 or 5000 nodes because the circuit has essentially $O(n^2 \lg n)$ complexity.

FIGURE 8 : Interleaved Look-Ahead Arbitration



The arbitration mechanism illustrated in Figure 8 exploits the characteristics of the OTDM hardware to achieve near lossless channel allocation. In interleaved look-ahead (ILA) arbitration, data is sent across the system in the form of packets of a fixed size (for example ATM cells). While the transmission of packet n proceeds, nodes that want to transmit data on channel x tune their transmitter and receiver to channel x and begin transmitting their node-ID, which is unique for each node in the system. Transmission of this ID occurs at regular intervals that are interleaved with the ongoing transmission of packet n. This interleaving is possible because all transmission are synchronized to the central framing pulse source, hence it is possible to operate multiple transmitters concurrently on the same channel. By convention, packets are transmitted with '0's in the position of the arbitration bits A_x. The recipient of packet n will simply ignore data in the arbitration bit positions. Both node A and B will only send their ID in the designated bit position so that they do not interfere with the ongoing packet transmission. When the participating nodes have sent one bit of their ID, they will monitor the data to see if the received ID bit matches the one that was sent out. If a discrepancy is detected, the node will not continue sending its ID rather it will retry on the next cycle. Because the received data is the logical *or* of the transmitted data and because node ID's are unique, exactly one node will succeed in sending its ID. This node is subsequently allowed to send its packet in slot $n+1$. This arbitration by successive

elimination of half of the potential requestors is logically similar to techniques used by IBM in the early 60s.

The reason for spreading the arbitration bits across an entire packet is to allow these bits to propagate to the hub and back and to allow sufficient processing time for the ID check processing. Suppose that nodes are located up to 2m away from the hub, and the system operates at 1 Gb/s, then the round trip delay is about 22 bit periods. With margin for processing latencies, arbitration bits need to be separated by 32 bits. In case of ATM traffic (53 byte cells), each cell has room for 13 bits of arbitration, allowing for up to 8192 contending nodes at the expense of a loss of 2.75% in channel bandwidth (actually, one might claim that no loss occurred because the arbitration bits could be used as the source address of the next packet). Larger hub to node distances or shorter packets require that the arbitration takes places more than one packet time in advance (deeper pipelining).

The actual cost for ILA arbitration is larger because it takes about 2-3 bit-times to switch channels, hence each arbitration bit costs about 5-7 actually transmitted bits. Alternatively, a dedicated receiver may be used for arbitration purposes. Since this receiver can share the delay element of the transmitter, which dominates cost, this approach is less expensive than a fully independent receiver.

3.1 Fairness and Priorities

ILA arbitration as described above enforces strict priorities: if the most significant bit of the ID is transmitted first, larger ID numbers will always win. This property is undesirable in systems that try to provide resources equally to all nodes and can lead to life locks. In order to achieve statistical fairness, randomized ILA arbitration scrambles the node IDs in a predictable fashion before using them. For this purpose, each node computes a sequence of pseudorandom numbers that are exclusive *or*-ed to the node ID before arbitration. Given the global synchronization, each node will compute exactly the same PRN sequence, hence the scrambled IDs are still unique. Provided that the PRN sequence has a periodicity of n (for example counters, LFSRs), it is guaranteed that each node will be granted access in no longer than n-1 cycles, hence it is obvious that randomized ILA arbitration is lifelock free.

There are applications where hardware supported priorities simplify the communication system. For example, a distributed shared memory system requires at least 2 levels of priorities to avoid deadlocks in the cache coherency protocols. Randomized ILA arbitration is easily modified to support multiple priority levels, by placing a binary representation of the priority level in front of the scrambled node ID. For example, to support 4 priority levels, 2 extra arbitration bits are needed. Within each priority level, fairness is assured.

3.2 Performance

An OTDM system with randomized ILA arbitration achieves the same performance as an input buffered crossbar switch. This means that in the case of randomly distributed packets, only about 60% of the available bandwidth can be utilized, provided that the outputs of the crossbar are never blocked. The ILA arbitration methods can be modified so that each node trying to send packets is given two (or more) arbitration cycles. If the first arbitration attempt fails due to a collision with another node, the sender tries to send the second pending packet in its outbound queue. In the case where nodes have the ability to queue outbound traffic and are able to deal with out-of-order packet delivery, dual ILA arbitration improves throughput significantly (Figure 9).

FIGURE 9 : Crossbar Saturation Throughput

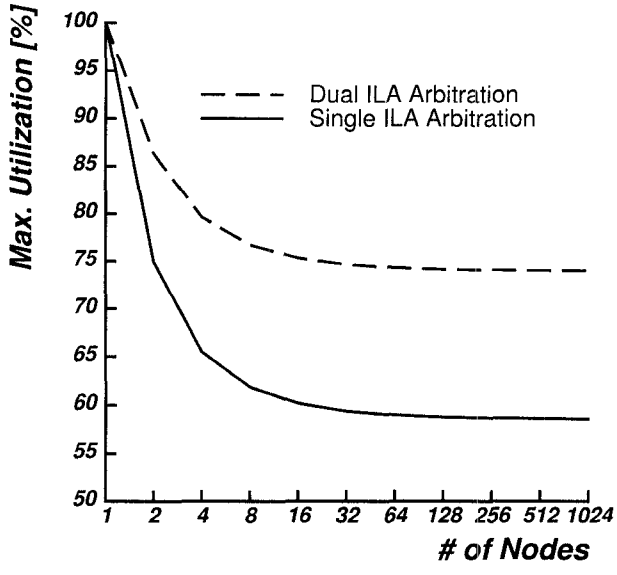


Figure 10 shows the saturation bandwidth of an OTDM interconnect system that uses the current implementation parameters. This number is put in perspective with the raw bisection bandwidth of several electronic switching fabric, where no deductions were made for protocol overhead or resource contention. Given that most of the electronic switches use wormhole-routine, their saturation throughput is typically limited to about 40%.

FIGURE 10 : OTDM Saturation Bandwidth

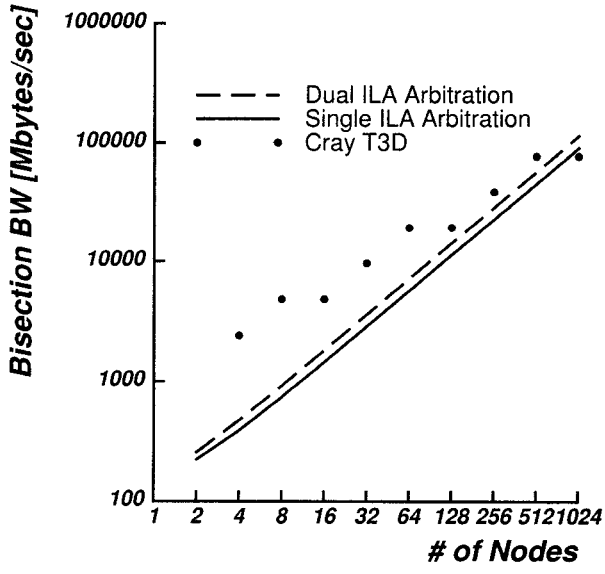
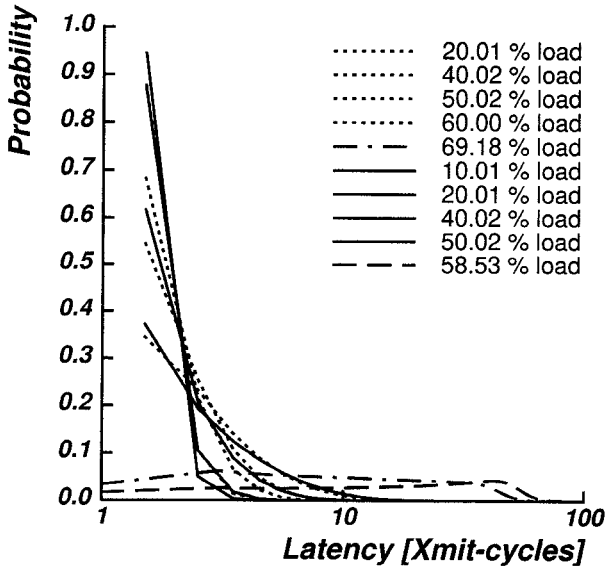


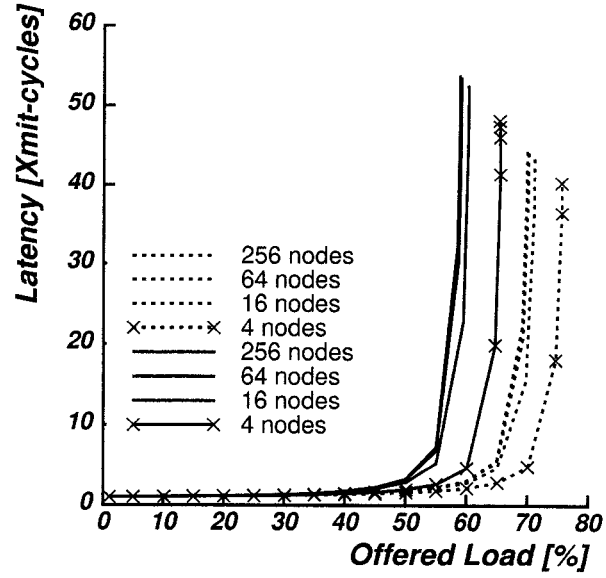
Figure 11 shows the impact of dual ILA arbitration on the latency distribution under relatively high load conditions. It should be noted that priorities can be used to control the latency for a subset of the traffic, for example if real-time applications such as audio and video demand timely delivery of their data. This capability is not present in most existing, electronic switching fabrics.

FIGURE 11 : Latency Distribution



Another important interconnect characteristic is the degradation behavior as the offered load approaches saturation (Figure 12). Given that there is practically no buffering in the OTDM system (all buffering takes place at the packet injection logic in each node), the traffic condition is directly observable by each node, without delay. Hence the data needed for traffic control algorithms is available locally.

FIGURE 12 : Throughput vs. Offered Load



3.3 Flow Control

Since the originator of data can tune his receiver to the channel of the data destination, it can listen to the destination sending a acknowledge signal at a predetermined time slot. Hence acknowledge signals are immediate and do not require that the destination node send a separate acknowledge packet. The destination may also participate in the ILA arbitration cycle. For example, by sending a '1' in all arbitration bit positions, the destination can prevent any node from succeeding (provided that the all-'1' node-ID is reserved for this purpose). In the case of dual ILA arbitration, the second arbitration cycle may succeed so that the originating node does not waste transmission cycles.

Likewise, the sending node may inhibit ILA arbitration by the same means in order to send a message that spans several packets. Since the OTDM system operates more efficiently on fixed size packets, variable length messages need to be sent as a sequence of packets. Inhibiting arbitration will make sure that these packet will be delivered as one burst. While in-order delivery is guaranteed anyway, it is desirable not to mix messages in order to keep the receiver interface simple.

3.4 Multichannel Network Interfaces

The bandwidth of a single transceiver is limited to that of one full-duplex, bit-serial connection. More demanding applications require access to a larger fraction of the total system bandwidth. Rather than using multiple, independent receivers, it is simpler to build a multichannel interface that uses several adjacent channels. As in Figure 1, one programmable delay element is used for each of the transmitter and receiver. However, the output of the delay element is split into several signals, each 1 channel spacing delayed in time. Separate modulators and TOADs are used for each of the these channels, each feeding a separate serial link. All

links can be integrated onto one controller chip [19]. Because the programmable delay elements dominate the transceiver cost, a multichannel interface is more practical than several independent transceivers. Furthermore, all bandwidth is allocated to a single channel that requires only one arbitration and that reduces latency.

In a multichannel system, the number of arbitration steps is also reduced because each step can resolve more than one address bit. For example, if a byte-wide interface is used, a unary encoding resolves 3 address bits in each arbitration cycle.

4 Message Passing Multicomputers

Besides higher throughput and lower latency, building a message passing multicomputer based on an OTDM interconnect system offers a number of capabilities that are difficult to implement in electronic switching fabrics, in particular global synchronization operations and multicasting. The synchronization capabilities of OTDM systems will be discussed in the next section.

Since multiple receivers can share a channel, multicasting simply requires designating a channel for the multicast and to schedule all recipients to listen to the multicast channel. The schedule for each receiver is established when the multicast channel is created and can be broadcast through a common, designated control channel to all I/O interfaces. ILA arbitration coordinates the sender as in the case of the single channel connection.

A simple flow-control mechanism uses a negative acknowledge by all recipients so that the absence of a signal in the designated slot indicates that all receivers have consumed the message. Unfortunately, in this case it is necessary that all recipients decide over successful reception immediately. To avoid this problem and allow more time for the receiver process, the messages that are broadcast need to carry a sequence number. This number is then used in a reverse ILA arbitration, where the receivers communicate the number of the most recent message that was processed by all receivers.

Multicasting through a dedicated channel means that the receiver of a participating node must be tuned to the multicast channel and cannot concurrently listen to messages that are addressed to it. The expensive solution to this problem is to add a dedicated receiver to each node. This does not mean that a node requires extra connections to the central hub because the shared medium carries all data. Furthermore, some optical components (for example the framing pulse extraction) can be shared.

In the case where nodes do not need the bandwidth that multiple receiver could provide, one receiver can be time shared between different traffic types (multicasts and dedicated node-to-node). For example, nodes may be required to listen to a global control channel once ever 10 msec. This control channel would be used to broadcast schedules when nodes are listening to multicasts.

The multicasting support does require that there spare channels, which are a limited resource and that methods are used that manage the network interface bandwidth efficiently. This problems have been studied in the context of ATM switches and many of the ATM solutions are applicable. The extra cost and complexity is justified for applications that intensively use multicasting and that have predictable and slowly changing traffic patters, in particular multimedia and video server applications.

5 Shared Memory Multiprocessors

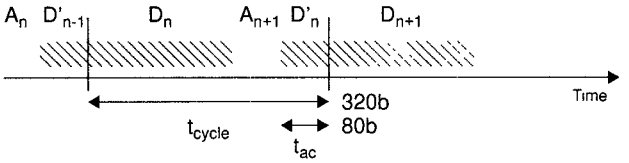
The most demanding application for high performance interconnect networks are shared memory multiprocessors because they tend to need more bandwidth and generate more irregular, fine grained traffic. The performance of shared memory multiprocessors depends critically on the latency. An OTDM system is insensitive towards traffic patterns by virtue of being fully connects. By providing about 250 Mbyte/sec bandwidth to each node with a single interface (multiple transceivers can increase this figure), OTDM is well match to shared memory systems with distributed caches. The latency is dominated by the parallel to serial to parallel conversion, which can be partially hidden by memory designs that supply the critical word first.

The most interesting feature for the architect of the memory hierarchy is the globally visible event ordering in OTDM, which is not present in any electronic multistage switching fabric. Because of this capability, cache coherency protocols can be simplified and synchronization operation can be supported directly.

5.1 Basic Remote Memory Access

Most shared memory multiprocessors collocate the memory units with the processing nodes so that local references do not need to traverse the global interconnect system. The architecture that is described below assumes this organization and considers only the global accesses where data is read from a random, remote node. Conceptually, it is easier to assume that memories and CPU are attached to the interconnect system through separate interfaces, even though in practice the transceiver hardware would be shared.

FIGURE 13 : OTDM Cycle Format



Basic memory access is started by a CPU trying to send the address of the memory location through the OTDM interconnect system to a memory controller. ILA arbitration

will guarantee that only one request will be received by the memory. The OTDM packet cycle matches that of the combined size of granularity of memory accesses (= cache line size) plus a control field that holds the address and an operation code, for example 400 bit (assuming a 32byte cache line, 64bit addresses, ECC code and some room for protocol related information). Address and data fields should be interleaved so that the time from the end of the address field to the time the first datum is sent matches the memory access time. By allowing independent arbitration for the data and address portions, the protocol differs from that of conventional processor/memory busses only by the means of arbitration and by the fact that each memory unit has its own, private bus. The later means that the arbitration for the data portion needs to consider only the small set of pending transactions and can occur during the t_{ac} time. Processors decide based on the address which bus to use, that is the OTDM channel number is part of the address.

TABLE 2 : Basic Remote Memory Access via OTDM

Line size	128 byte	32 byte
Cycle time	1280ns	400ns
Mean read latency[a]	820ns	400ns
R/W bandwidth	100 Mbyte/node	80 Mbyte/node

a. Assumes critical word first delivery of data.

Table 2 gives the performance for remote memory operations in the case of an unloaded system based on the first generation OTDM hardware. Contributing to the read-latency is the time spend waiting for the next arbitration cycle to occur. While the time of an arbitration cycle is predictable, the time of a cache miss is not, hence on average, 1/2 of a cycle is spent waiting. This cycle can be reduced by using shorter arbitration cycles, for example one every 80 bits, at the expense of less bandwidth due to increased overhead.

5.2 Complex Memory Operations

The basic memory operation does not support any form of cache coherency. However, global event ordering is provided: once a write cycle has completed, all subsequent read cycles on any node will return the new value. It also guarantees write atomicity: the order of two writes will be observed by all remote nodes in the same order.

Given that OTDM provides broadcasting of all operations, snooping could be considered as a way to maintain consistency, but this is not practical because the receiver can listen only to one channel at a time. However a large system could partition itself such that snooping is used on small subsets of the machines. Essentially, the multicast facility would be employed.

Synchronization operations are relatively easy to implement directly in the OTDM interface:

- Atomic Test&Set: The memory controller prevents arbitration until the result is returned. In the case of a heavily contended semaphore, the test&set operation combine in the interconnect: nodes that issued the same operation will lose in the ILA arbitration cycle and can see the ID and transaction of the succeeding node by monitoring the transaction, which is necessary to verify that the addresses match.
- Atomic Fetch&Op: for limited operations, namely those that can be realized by *or*-ing data. A complete fetch-and-add is not possible.
- Barrier synchronization: uses the multicast acknowledge method. Can also be implemented via dedicated bit(s) in a control channel that are tested periodically by all participating nodes.

5.3 Coherency Support

Like other scalable interconnect architectures, full cache coherency can be implemented via directory protocols similar to those used in Alewife, DASH and S3.mp [11,12,13]. However the directory protocols can be simplified by relying on the globally visible event ordering[1]:

- Write-update protocols become practical because the new value can be broadcast in one cycle so that the new value becomes visible atomically.
- Three party transactions, such as transferring ownership in an invalidation based protocol allow reflective memory support, where both the home and local node receive the dirty cache line in one transaction cycle.
- Broadcasting becomes a viable solution for directories with limited number of pointers.

6 Discussion and Future Work

The main problem for OTDM systems right now is the transition from the laboratory experiment to a commercial product. The process of cost-reducing the device is partially driven by the demand, which in turn is a function of the cost. The first step is the construction of a small network of 8-16 workstations connected via OTDM running at 1.3 Gbit/sec for each channel. Concurrently, future research will be directed at refining the OTDM cache coherency protocols.

OTDM is competing with optical wavelength division multiplexing (OWDM) in the sense that both systems are capable of sending multiple high-speed data streams over the same medium concurrently, hence both systems have the same high throughput potential can be configured as a virtual crossbar switch. However it is very difficult to control the laser frequency precisely and to filter out frequencies that may differ only by a few Ghz, which is not much given that the operating frequency is near 250 Thz. This has lim-

1. While all coherency transactions are globally visible, it is not possible for a node to snoop all memory transactions due to the limited network interface bandwidth. However, the events that are relevant to a directory based coherency protocol are visible to the participating parties: once an agent wins the arbitration cycle, it has immediate and definite confirmation that both sides have completed the state transition. Hence the protocol does not need to deal with pending messages, race conditions, out of order delivery, etc.

ited OWDM systems to small numbers of channels and to very high channel switching times [14,15].

Another difference is that OWDM channels have no well defined timing relation, so that the OWDM systems are more like a large collection of ethernets. When a receiver switches channels, it needs to reestablish word alignment. Hence it is not possible to *OR* multiple transmissions reliably, as is the case in OTDM, hence ILA arbitration and all functions based thereupon are not feasible in OWDM systems.

7 Conclusion

Optical Time Domain Multiplexing (OTDM) based on the recently developed Thz Optical Asymmetric Demultiplexer will enable the construction of crossbar interconnect systems with ultimate capacities of up to 5 Tbits/sec that interface directly with electronic processing elements. It was shown that the synchronicity inherent in a OTDM system can be exploited to implement efficient arbitration and control methods that scale linearly with the number of attached nodes. OTDM based interconnect systems challenge the current generation of switching fabrics that use electronic routers by offering lower latencies, higher bandwidth and globally visible event ordering, which greatly simplifies synchronization and coherency protocols. Furthermore, nodes are connected to a central switching hub in a starlike fashion where each node requires 2 optical fiber connections, independent of the system size. Nodes may be separated from the hub by up to 300ft, so that the system could be spatially distributed over 1 building while providing a bisection bandwidth of >10x that of a Cray T3D supercomputer.

The TOAD-based OTDM system does not require the development of any new optical components, rather it can be implemented with commercially available components. However before OTDM systems will replace the current interconnect system, it is necessary to integrate the optical components into a compact, economical device.

8 Acknowledgments

Prof. Paul Prucnal, J.P. Sokoloff, I. Glesk and M.Kane achieved the main innovation, the TOAD, that allows the construction of practical OTDM interconnect systems. All research concerning the optical components were carried out in the Advanced Technology Center for Photonics and Optoelectronic Materials, Princeton University. Discussions with Dr. H. Davidson of Sun Laboratories and Dr. C. Gosh of the David Sarnoff Research Center clarified the viability of OTDM. The authors thank the reviewers for their excellent comments and constructive suggestions.

References

[1] Inmos, *IMS C104 Packet Routing Switch*, The T900 Transputer Productes Overview Manual, April 1991

[2] Triquint, Crosspoint Switch Products (*TQS TC8032BM*), 1994, 2300 Owen St., Santa Clara, CA 95054

[3] J.P. Sokoloff, P.R. Prucnal, I. Glesk, M. Kane, *A Teraherz Optical Asymmetric Demultiplexer*, IEEE Photonics Technology Letters, Vol. 5, No. 7, July 1993

[4] Michel Chbat, Ph.D. Thesis, Dept. of Electrical Engineering, Princeton University, 1993, Table 4.1

[5] I. Duling III, *Ultrshort-Pulse Fiber Lasers at 1.5 Micrometers*, Laser Tech Briefs, Spring 1994, Vol. 2, No. 2, p 38.

[6] P.A. Perrier, P.R. Prucnal, *High-Dimensionality Shared-Medium Photonic Switch*, IEEE Transactions on Communications, Vol. 41, No. 1, January 1993, p.224

[7] P.R. Prucnal, J.P. Sokoloff, I. Glesk, *Time-Division Optical Mico-Area Networks*, 27th Annual Hawaiian International Conference on System Sciences, January 1994

[8] D.K. Jeong, et al., *HotPads - Macro-Cells for Gigabit I/O*, Hot Interconnect Symposium II, Stanford CA, August 1994

[9] P.R. Prucnal, M.A. Santoro, S.K. Sehgal, *Ultrafast All-Optical Synchronous Multiple Access Fiber Networks*, IEEE J. Select. Areas Communications, SAC-4 #9, pp1484-1493, AON012

[10]The First Networsk of Workstatins Workshop, October 4, 1994, San Jose, California

[11]Agarwal, A.,Kubiatowicz J., Kranz, D., Lim, B., Yeung, D., D'Souza, G., Parkin, M. *Sparcle: An Evolutionary Processor Design for Large-Scale Multiprocessors*. IEEE Micro, June 1993, pages 48-61.

[12]Lenoski, D. *The Design and Analysis of DASH: A Scalable Directory-Based Multiprocessor*. PhD Dissertation, Stanford University, December 1991.

[13]A. Nowatzyk, G. Aybay, M. Browne, E. Kelly, M. Parkin, W. Radke, S. Vishin, *S3.mp: Current Status and Future Directions*, 4 Workshop on Shared Memory Multiprocessors, ISCA 1994, Chicago, Illinois

[14]M. Borella, B. Mukherjee, F. Jia, S. Ramamurthy, D. Banerjee, J. Iness, *Optical Interconnects for Multiprocessor Architectures Using Wavelength-Division Multiplexing*, 27th Annual Hawaiian International Conference on System Sciences, January 1994

[15]J.C. Lu, L. Kleinrock, *A Wavelength division multiple access protocol for high-speed local area networsk with a passive star topology*, Performance Evaluation Journal, Vol. 16, No. 1-3, pp. 223-239, Nov. 1992

[16]P.W. Dowd, *Random Access Protocols for High Speed Interprocessor Communication based on an Optical Passive Star Topology*, Journal of Lightwave Technology, Vol. 9, pp. 799-808, June 1991

[17]P.R. Prucnal, M.F. Krol, J.L. Stacy, *Demonstration of a Rapidly Tunable Optical Time-Division Multiple Access Coder*, Photonics Tech. Letters, Vol. 3, #2, pp 170-172, OTV015

[18]IOT product information on integrated optical components, IOT GmbH, P.O.Box 1252, D-68744 Waghausel-Kirrlach, Germany

[19]A.G. Nowatzyk, M. Parkin, The S3.mp Interconnect System and TIC chip, Hot Interconnect Symposium, Stanford CA, August 1993