CIXB-1: Combined Input-One-cell-Crosspoint Buffered Switch

Roberto Rojas-Cessa,* Eiji Oki,** Zhigang Jing, and H. Jonathan Chao
Department of Electrical Engineering
Polytechnic University
6 Metrotech Center, Brooklyn, New York 11201, USA
email: {rrojas, eoki, zgjing, chao}@kings.poly.edu

Abstract—
Buffered crossbars have been considered as an alternative for non-buffered crossbars to improve switching throughput. The drawback of a buffered crossbar is the memory amount that is proportional to the square of the number of ports ($O(N^2)$). This is not the main limitation when the buffer size is kept to a minimum size such that implementation is feasible. For a small buffer size, the number of ports of a switch module is not limited by the memory amount but by the pin count. We propose a novel architecture: a Combined Input-One-cell-Crosspoint Buffer crossbar (CIXB-1) with Virtual Output Queues (VOQs) at the inputs and round-robin arbitration. We show that the proposed architecture can provide 100% throughput under uniform traffic. A CIXB-1 offers several advantages for a feasible implementation such as scalability and timing relaxation. With the currently available memory technology, a one-cell crosspoint buffered switch is feasible for a 32×32 fabric module.

I. Introduction

Crossbar switching fabrics are very popular for switch implementation because of their non-blocking capability, simplicity, and their market availability. A switch with a crossbar fabric and queues at the output ports to store those cells that could not be sent to the output lines is called Output Buffered (OB). In an OB switch, all cells coming to an input are forwarded to the destined outputs as they arrive. This architecture is not scalable because the required internal bandwidth or speedup (S) —defined as the number of times that the switch core works faster than the input line rate— is equal to the number of ports ($S = N$); the working speed of the switch core and the output memory make its implementation infeasible for even a medium sized switch. However, this architecture has been used as a comparison reference for any switch model because of its desirable characteristics such as high throughput and low delay.

Crossbars could have input queues to store those cells (or packets) that could not go through because of contention for an output; this architecture is known as Input Buffered (IB). An IB architecture is scalable and its implementation does not have the restrictions of an OB model because the core fabric works at the input line rate ($S = 1$). However, IB switches need to resolve input and output contention by means of arbiters at the inputs and outputs. The requirements for such arbiters are (a) low complexity, (b) fast contention resolution and, (c) high efficiency to provide a high performance. Low complexity is needed to make implementation feasible. For a high-capacity switch, a fast resolution is necessary so that the arbiter can select a cell among those eligible in the allotted time.

Head of Line (HOL) blocking is a well-known problem for a crossbar with FIFOs at the inputs [1]. This problem is overcome by using separate queues at the inputs, one for each output. This queuing system is called Virtual Output Queuing (VOQ). For a crossbar with VOQs, maximum matching algorithms have been proposed to achieve 100% throughput. Maximum matching algorithms are efficient but with such a high complexity [2] that implementation is infeasible for high-speed systems. Schemes based on a maximum size or weight matching, like Longest Port Queuing (LPQ), Oldest Cell First (OCF), and Longest Port First (LPF) [3], have been proposed [2].

Maximal matching schemes have been considered as an alternative to maximum matching schemes; iSLIP [4], Dual Round-Robin Matching (DRRM) [5], and Longest Output Occupancy First Algorithm (LOOFA) [6] are examples. To make up for the lack of efficiency that a maximal scheme has (compared to a maximum type), a number of iterations —where the number of iterations is the number of times that an algorithm is performed to obtain a cumulative result, speedup, or a combination of both is used, as in LOOFA. iSLIP is a good example of an iterative matching scheme. Although iSLIP provides 100% throughput for uniform independent traffic, because of the arbitration time and connection state amount of this arbitration scheme, it has been proposed for a small number of ports [7] due to its centralized implementation (i.e., 32 for iSLIP). Transmission of phases: request, grant, and acknowledge are performed within a cell slot between input and output arbiters. This transmission of information reduces the available time for arbitration because these

*Corresponding author. Phone:(718)-260-3496, fax:(718)-260-3906
**Research Engineer at NTT Network Service Systems Laboratories. This work was done while he was a Visiting Scholar at Polytechnic University.

transmission phases are performed during the cell slot in serial with input and output arbitration, even when the transmissions are done within a single chip (so that the off-chip delay is avoided). Another drawback with the proposed single-chip centralized implementation is that the pin count limits the number of ports.

A switch architecture using speedup, such that $1 < S < N$, is called Combined Input and Output Buffered (CIOB), where queues are placed at the inputs and outputs. As the demand for high switching rates increases, this speedup becomes a bottleneck since the available time for arbitration is inversely proportional to the cell slot duration divided by S. The DRRM scheme considers speedup instead of a number of iterations to improve the matching performance. Although the overhead information exchanged between input and output arbitration is reduced in this scheme, the arbitration time becomes insufficient for a switch with a large number of ports and with a high port speed.

For a long time, buffered crossbars have been considered as a solution to improve switching throughput instead of non-buffered crossbars. However, it is known that the number of buffers would grow in the same order as the number of crosspoints ($O(N^2)$, where N is the number of ports), making implementation infeasible for the memory amount required with a large buffer size or a large N. With currently available standard cell technology, a large memory amount can be implemented in a chip (i.e., 1 Mb with a 0.18 μm, [8]). It is interesting to note that for small sized crosspoint buffers, the size limitation in terms of number of ports for a buffered crossbar is set by the number of pins and not by the memory amount.

In pure buffered crossbars —we call a pure crossbar to the architecture that only has buffering at the crosspoints and none in any other place, a large crosspoint buffer has been utilized to minimize cell loss. The number of ports is limited by the memory amount that can be implemented in a module chip. An example of this architecture was proposed in [9], where a 2×2 switch module with a crosspoint memory of 16 kbytes each was implemented. In this architecture, a large crosspoint buffer is needed to store all those cells that could not be switched to the output port to comply with the required cell loss rate.

To reduce the memory amount in the crosspoint buffer, input queues are used. FIFO queues have been proposed, where HOL blocking, as in a non-buffered crossbar, remains in this architecture. Examples of these architectures were presented in [10], [11], and [12]. A buffered crossbar with a single-cell buffer was proposed in [10], and [11], together with a FIFO input buffer at the input ports. This architecture provides an improvement over non-buffered crossbars with FIFO input buffers. The well-known limited throughput of a FIFO input-buffered architecture of about 58% was improved to 91% with a priority scheme (also called HOL blocking scheme by the same author). However, the FIFO buffers at the inputs limit the maximum throughput performance in this architecture because the HOL blocking can not be completely eliminated. In [12] a similar architecture with a 4-cell crosspoint buffer is considered. This buffered crossbar, used with 32-cell input FIFOs, achieves an acceptable cell loss (10^{-8}). In this architecture, a flow control mechanism is also used to avoid cell loss at the core. All cell loss occurs at the input FIFO for a very congested output. This study shows that with input FIFOs and a small sized crosspoint buffer together with a flow control mechanism, the cell loss rate can be kept small and the HOL blocking diminished to a certain degree.

As with maximal matching schemes as in non-buffered crossbars, the HOL blocking problem for FIFO buffers can be overcome in a buffered crossbar with the consideration of VOQs.

100% throughput is obviously achieved for a buffered crossbar with infinite crosspoint buffer sizes [13], [14], and [15]. To our knowledge, no maximum or minimum finite memory size has been specifically proved to provide a 100% throughput for a buffered crossbar.

In this paper, we propose a novel architecture: a one-cell crosspoint buffer crossbar with VOQ at the inputs (CIXB-1) and simple round-robin for input and output arbitration. We show that the combination of input and crosspoint buffer, with a single-cell buffer and round-robin arbitration scheme provides 100% throughput under uniform traffic. In this architecture input and output arbitration are more independent than in a non-buffered crossbar model, simplifying arbitration time complexity. The property of a buffered crossbar allows a simplification of the arbiter design and the adoption of distributed-fashion arbiters.

We show an improvement on delay performance of this architecture for uniform and unbalanced traffic compared with a non-buffered crossbar with iSLIP arbitration, the effect of burstiness with on-off traffic, and the impact of the switch size with uniform traffic.

The organization of this paper is as follows. In Section II, we present our switch model and the properties of a buffered crossbar with a single-cell crosspoint buffer such that combined with round-robin arbitration, it offers 100% throughput under uniform traffic.

In Section III, we discuss the advantages of CIXB-1 in terms of time complexity. In Section IV, we present a simulation study of delay performance of this architecture. In Section V, we present our conclusions.

II. Combined Input and One-cell Crosspoint Buffered Switch (CIXB-1)

In this section, we present a number of properties of the CIXB-1 architecture. We show that that 100% throughput is achieved with a simple round-robin arbitration for inde-

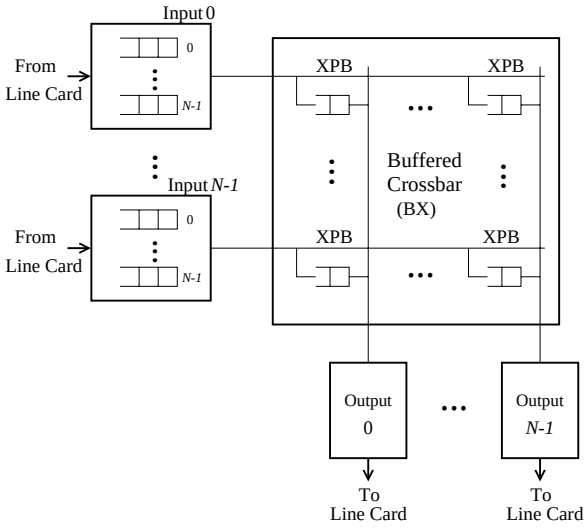
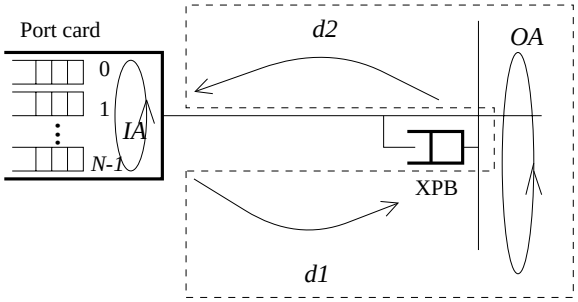


Fig. 1. Buffered crossbar architecture



OA: Output Arbitration time
IA: Input Arbitration time
d1 Transmission delay from portcard to crossbar
d2 Transmission delay from crossbar to portcard

Fig. 2. Round trip between port and crossbar cards

pendent uniform traffic.

We make some notes about real delays such as transmission and arbitration. These delays need to be absorbed by the crossbar buffers to keep the characteristics of CIXB-1 intact.

A. Switch Model

In this section, we introduce the proposed architecture and the terminology used in the rest of the paper.

A Buffered Crossbar (BX) has N input and output ports. A crosspoint (XP) element in the BX that connects input i, where $0 \leq i \leq N-1$, to output j, where $0 \leq j \leq N-1$, is denoted as $XP_{i,j}$. The XP Buffer (XPB) of $XP_{i,j}$ is denoted as $XPB_{i,j}$.

We consider fixed size packets, named cells. A variable length packet can be segmented into cells for internal switching and re-assembled before it leaves the switch. The transmission time has a fixed length, called cell or time slot.

Our switch model has a structure as described below and shown in Figure 1:

- *Input Queue.* There are VOQs at the input ports. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$.
- *Crosspoint Buffer XPB.* Each crosspoint has a one-cell buffer. Only those inputs with a cell in the crosspoint buffer are considered for output arbitration.
- *Flow Control.* A flow control mechanism tells the input port i which $XPB_{i,j}$ is occupied, so that the $VOQ_{i,j}$ is inhibited (a non-empty and non-inhibited VOQ is considered eligible for input arbitration). In this paper, we assume a credit-base flow control mechanism, unless otherwise specified.
- *Round Trip (RT).* We define RT as standing at the input port. RT is the composite time of the transmission cell delay from a portcard to the crossbar ($d1$) plus the Output Arbitration time (OA) and the tranmission of the flow control information back to the portcard ($d2$), as shown in Figure 2. RT is given in a number of cell cycles. Cell and data alignments are included in the transmission times. As a general case:

$$RT = d1 + OA + d2 \leq C_{XPB} - IA \qquad (1)$$

  where IA is the input arbitration time in time slots and C_{XPB} (i.e., CIXB-1 means $C_{XPB} = 1$) is the crosspoint buffer size.
  The constraints for IA and OA are:

$$IA < 1, \qquad (2)$$

  and

$$OA < 1. \qquad (3)$$

- *Arbitration.* Round-robin arbitration is used at the input and output ports. The input arbiter selects a VOQ if there is at least a single eligible VOQ. The eligibility of VOQs is determined by the flow control mechanism. Selection of a crosspoint per output is performed similarly in a round robin fashion, where only non-empty crosspoint buffers are considered.

B. Properties of CIXB-1

CIXB-1 reaches a state such that the number of cells entering the system be equal to the ones leaving it (i.e., defining 100% throughput). We prove 100% throughput in this system by showing that no input is totally inhibited and observing the service of the VOQs.

Traffic Model. In this section, we consider the following traffic model: All inputs have uniform traffic; where each

$VOQ_{i,j}$ receives cells at rate $\rho_{i,j}$ such that $\sum_{i=0}^{N-1} \rho_{i,j} = 1.0$, $\sum_{j=0}^{N-1} \rho_{i,j} = 1.0$, and each $VOQ_{i,j}$ has an occupancy $\sigma_{i,j}(t)$, such that $\sigma_{i,j}(t) < \infty$ and $\sigma_{i,j}(t) > 0$ [1].

With these switch and traffic models, we found that CIXB-1 achieves 100% throughput under uniform traffic.

III. Arbitration Time Complexity

We have selected round-robin as the arbitration scheme in this architecture because of its fairness and simple implementation. Different from a maximal matching scheme, input arbitration and output arbitration in CIXB-1 are performed more independently. The information exchanges such as request, grant, and accept are not used. Input arbitration, as output arbitration, can be performed for the complete cell slot time. The advantages of this architecture are as follows:

- *Timing relaxation.* Speedup and iterations are not necessary to achieve 100% throughput. A cell slot is better utilized for arbitration. This property allows a relaxed cell-alignment and clocking. Furthermore, these features allow switch scalability.
- *Fast arbitration.* The cell transmission is separated from the arbitration time as well as the flow control information exchange. The arbitration complexity is $O(N)$, which can be reduced to $O(\log N)$ with a suitable encoding logic. Also, the information exchange between the crosspoints and the arbiters is done in-chip for a distributed arbiter implementation.

IV. Delay Performance

We simulated a CIXB-1 32×32 switch model and compared it to an architecture with a non-buffered crossbar with the iSLIP arbitration scheme, where we considered the cases for 1 and 4 iterations. As originally presented in [4], iSLIP with more than 4 iterations offers no measurable improvement for uniform traffic. Also, more iterations take longer to perform and the allotted arbitration time for a large switch size is limited. The traffic patterns studied in this section are uniform and non-uniform traffic with Bernoulli arrivals. We also studied the effect of burstiness with an on-off traffic model. We show a comparison with the average delay of the OB switch model and the performance of the BX for different switch sizes. Our simulation results are obtained with 95% confidence interval, not greater than 5% for the average cell delay.

A. Uniform Traffic

The average delay results of the simulation of CIXB-1, iSLIP with 1 and 4 iterations (1-SLIP and 4-SLIP, respectively) and OB for traffic uniformly distributed to all output ports with Bernoulli arrivals are shown in Figure 3.

[1] We assume this condition as the initial state for simplicity in the demonstration; however, we can start from an empty queue as well.

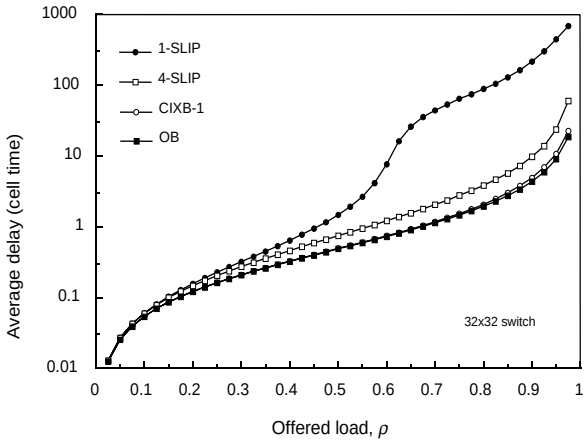


Fig. 3. Performance of CIXB-1, iSLIP, and OB under Bernoulli uniform traffic

Under independent uniform traffic, CIXB-1 has a smaller average delay than 4-SLIP. This result is achieved even when CIXB-1 uses a simpler scheduling scheme (i.e., no iterations) with no speedup, so that arbitration time is relaxed. We can also see that CIXB-1 has comparable average delay performance to OB.

B. Non-uniform Traffic

We define the non-uniform traffic by using an unbalanced probability w. Let us consider input port s, output port d, and the offered input load ρ for each input port. The traffic load from input port s to output port d, $\rho_{s,d}$ is given by,

$$\rho_{s,d} = \begin{cases} \rho\left(w + \frac{1-w}{N}\right) & \text{if } s = d \\ \rho\frac{1-w}{N} & \text{otherwise.} \end{cases} \quad (4)$$

where N is the switch size. Here, the aggregate offered load that goes to output d from all input ports, ρ_d is given by,

$$\rho_d = \sum_s \rho_{s,d} = \rho\left(w + N \times \frac{1-w}{N}\right) = \rho. \quad (5)$$

When $w = 0$, the offered traffic is uniform. On the other hand, when $w = 1$, the traffic is completely unbalanced. This means that all the traffic of input port s is destined for output port d only, where $s = d$.

The average delay of CIXB-1 and iSLIP for unbalanced traffic is shown in Figure 4. We can see that the performance of CIXB-1 is always better than 1-SLIP and 4-SLIP. The throughput of CIXB-1 is almost 100% when the w is about zero (i.e., uniform traffic) and about one (i.e., totally directional traffic).

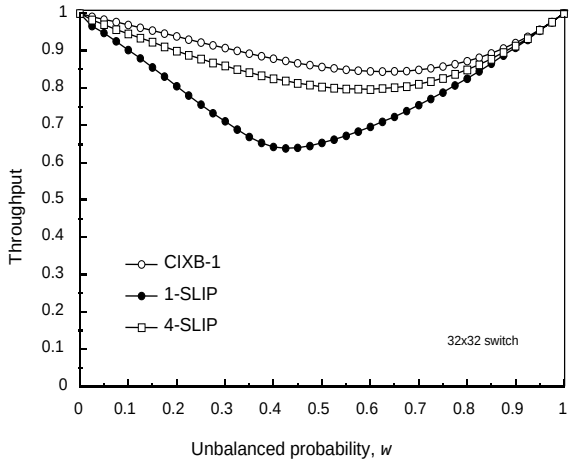


Fig. 4. Performance of CIXB-1, 1-SLIP, and 4-SLIP under unbalanced traffic

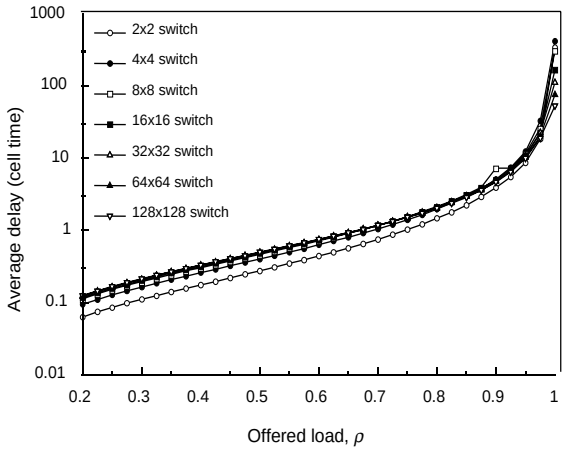


Fig. 6. Performance of CIXB-1 for different switch sizes

C. Burstiness and switch size effect in CIXB-1

Figure 5 shows the average delay performance of CIXB-1 and OB for on-off arrival uniform traffic. Bernoulli traffic and trains of average burst lengths (l) of 10 and 100 cells are used. We can see that the average delay is proportional to the burst length and that the throughput is unaffected at any load. The average delay of CIXB-1 is close to that of OB for any case. In Figure 6, we show the average latency of a CIXB-1 for different sizes of $N \times N$ switches with Bernoulli arrival traffic. We can see that the average latency difference is almost negligible for different switch sizes.

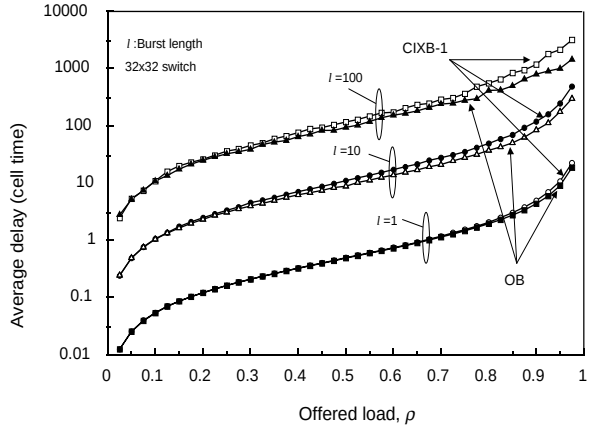


Fig. 5. Performance with Bernoulli arrivals and burst lengths of 10 and 100 cells in CIXB-1 and OB

V. Conclusions

We have introduced a novel switch architecture that presents a high-speed arbitration and scalability features, a Combined Input-One-Cell-Crosspoint Buffered (CIXB-1) switch model. We have also presented a series of properties in the CIXB-1 such that 100% throughput and arbiter simplification are achieved. The stability property of a buffered crossbar helps to simplify the complexity of the arbitration scheme. We have shown that CIXB-1 presents several advantages for implementation, such as timing relaxation, simplification of arbitration, scalability, and a better delay performance than a non-buffered crossbar architecture with iSLIP. Timing relaxation is a very important feature for switch implementation to supply time for cell alignment and a better utilization of a cell slot for arbitration. A crossbar with a single-cell crosspoint buffer is not limited by the memory amount; the pin number remains the limitation as in the case for a non-buffered crossbar. The memory amount grows in the same order as the number of crosspoints. This is a disadvantage, but for a small buffer size as in CIXB-1 and medium-sized switch module, the implementation is feasible. A 32×32 CIXB-1 switch module requires 512 kbits for 64-byte cell size, and up to 1 Mbit memory can be embedded in a chip using available 0.18-μm CMOS technology. CIXB-1 offers a very close average delay to that of an output buffered switch model for uniform traffic with Bernoulli and bursty arrivals.

References

[1] M. Karol, M. Hluchyj, "Queuing in High-performance Packet-switching," *IEEE J. Selected Area Communications*, vol. 6, pp. 1587-1597, December 1988.

[2] N. McKeown, A. Mekkittikul, V. Anantharam, V., and J. Walrand, "Achieving 100% Throughput in an Input-queued Switch," *IEEE Trans. On Comm.*, vol. 47, no. 8, pp. 1260-1267, August 1999.

[3] A. Mekkittikul and N. McKeown, "A Practical Scheduling Algorithm to Achieve 100% throughput in input-queued switches," *INFOCOM '98*, vol. 2 , pp. 792 -799, March 1998.
[4] N. McKeown, "The *i*SLIP Scheduling Algorithm for Input-Queue Switches", *IEEE Trans. Networking*, vol. 7, no. 2, pp. 188-200, April 1999.
[5] H. J. Chao and J-S. Park, "Centralized Contention Resolution Schemes for a Large-capacity Optical ATM Switch," *Proc. IEEE ATM Workshop*, Fairfax, VA, pp. 10-11, May 1998.
[6] P. Krishna, N. S. Patel, A. Charny, R. Simcoe, "On the Speedup Required for Work-Conserving Crossbar Switches," *IEEE Selected Areas in Communications*, vol. 27, no. 6, pp. 1052-1066, June 1999.
[7] N. McKeown, M. Izzard, A. Mekkittikul, W. Ellersick, and M. Horowitz, "Tiny Tera: A Packet Switch Core," *IEEE Micro*, vol. 17, no. 1, pp. 26 -33, Jan.-Feb. 1997.
[8] GS40 0.15-μm CMOS, Standard Cell/Gate Array, Texas Instruments, http://www.ti.com/, version 0.2, May 2000.
[9] S. Nojima, E. Tsutsui, H. Fukuda, and M. Hashimmoto, "Integrated Packet Network Using Bus Matrix," *IEEE JSAC*, vol. SAC-5, no. 8, pp. 1284-1291, October 1987.
[10] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, "16 x 16 Limited Intermediate Buffer Switch Module for ATM Networks," *GLOBECOM '91*, pp. 939-943, December 1991.
[11] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, "Limited Intermediate Buffer Switch Modules and their Interconnection Networks for B-ISDN," *ICC '92*, pp. 1646-1650, June 1992.
[12] Y. Doi and N. Yamanaka, "A High-Speed ATM Switch with Input and Cross-Point Buffers," *IEICE TRANS. COMMUN., VOL. E76, NO.3*, pp. 310-314, March 1993.
[13] F. A. Tobagi, "Fast Packet Switch Architectures For Broadband Integrated Services Digital Networks," *Proceedings of the IEEE*, vol. 78, No. 1, pp. 133-167, January 1990.
[14] H. Ahmadi and W. E. Denzel, "A Survey of Modern High-Performance Switching Techniques," *IEEE JSAC*, vol. 7, no. 7, pp. 1091-1103, September 1989.
[15] R. Y. Awdeh and H. T. Mouftah, "Survey of ATM Switch Architectures," *Computer Networks and ISDN Systems*, vol. 27, pp. 47-93, 1995.
[16] K. H. Rosen, "Discrete Mathematics and Its Applications," McGraw-Hill, 1995.