

# Two-Dimensional Crossbar Matrix Switch Architecture

\*Jong Arm Jun, \*Sung Hyuk Byun, \*Byung Jun Ahn, \*\*Seung Yeob Nam and \*\*Dan Keun Sung  
\*Dept. of Internet Technology, ETRI, 161, Gajeong-dong, Yusong-gu, Taejeon, 305-350, KOREA  
\*\*Dept. of EECS, KAIST, 373-1, Kusong-dong, Yusong-gu, Taejeon, 305-701, KOREA

E-mail: jajun@etri.re.kr, shbyun@etri.re.kr, bjahn@etri.re.kr;  
synam@cnr.kaist.ac.kr, dksung@ee.kaist.ac.kr

## Abstract

*Explosive growth of Internet traffic causes a new challenge in the design of high-speed switches. One of main design issues for high-speed switches is a scalability problem. This paper proposes a scalable two-dimensional crossbar matrix switch (CMS) architecture, which is composed of multiple crossbar switch units with virtual output queues (VOQs) at the inputs and single-cell scheduling decomposition buffers (SDBs) at the outputs of each crossbar switch unit. We propose a hierarchical scheduling algorithm for the proposed crossbar switch architecture, and show that the proposed switch architecture and hierarchical scheduling algorithm can provide 100% throughput under i.i.d uniform traffic.*

**Keywords:** Crossbar Switch

## 1. INTRODUCTION

A switch with a crossbar switch fabric and queues at the output ports is called the output buffered switch. In  $N \times N$  output buffered switches, the required operation speed of the internal switching core is  $N$  times faster than the input line rate to forward all incoming cells to the destined output ports as they arrive. The main reason that this architecture can not be used for high-speed switches is the speed-up requirement. Even if this architecture is not applicable to high-speed switches, it has been used as a performance reference for any switch model because of its high throughput and low delay.

A switch with a crossbar switch fabric and queues at the input ports is called the input buffered switch. In input buffered switches, a complicated arbitration scheme is required to resolve contentions at input and output ports. There have been a number of studies on arbitration algorithms: parallel iterative matching (PIM)[1], round-robin matching (RRM), iSLIP[2], FCFS in round-robin matching(FIRM)[3], wave front arbitration (WFA)[4], 2-dimensional round-robin(2DRR)[5], and Dual Round Robin Matching (DRRM) [6], etc.

High-speed switches with a crossbar switch fabric and VOQs at the inputs have become very popular because of its feasibility for high-speed switch. For a crossbar switch with VOQs, many maximal matching algorithms have been proposed to achieve 100% throughput under uniform traffic, such as iSLIP[3]. Although such algorithms can provide 100% throughput for uniform and independent traffic, because of limitation in their

arbitration time, they have been used for a small number of ports (i.e., 32 for iSLIP) [7].

We have an inherent problem in implementing a single crossbar switch fabric with a large number of ports. For example, we need a 512x512 crossbar switch fabric with 2.5 Gbps port speed and 1.28 Tbps switching capacity. Thus, we need a scalable crossbar switch architecture that can be scaled up to more than 1 Tbps switching capacity without performance degradation. Decomposed crossbar switches with multiple input and output buffers have been proposed to decrease the input and output contention probabilities for high-speed large-scale switching systems [8].

We propose a scalable two-dimensional crossbar matrix switch architecture, which is composed of multiple crossbar switch units with VOQs at the inputs and single-cell SDBs at the outputs of each crossbar switch unit. High-speed switches with more than 1 Tbps switching capacity can be built up by scaling-up the proposed crossbar switch units.

We propose a hierarchical scheduling algorithm for the proposed crossbar switch architecture and show that the proposed architecture and scheduling algorithm can provide 100% throughput under uniform traffic without limitation in arbitration time.

The organization of this paper is as follows. In Section II, we describe a CMS switch architecture and propose a hierarchical scheduling algorithm for the described crossbar switch architecture. In Section III, we show the performance evaluation of the described architecture for various traffic patterns. In Section IV, we present our conclusions.

## 2. CMS SWITCH

We describe a two-dimensional CMS switch architecture and propose a hierarchical scheduling algorithm for the proposed switch architecture.

### A. Switch Architecture

Figure 1 shows the switch architecture. The terminology used in this figure is listed as follows:

<i>EOB</i>	Equivalent Output Buffer
<i>SDB</i>	Scheduling Decomposition Buffer
<i>XSU</i>	Crossbar Switch Unit
<i>XSM</i>	Crossbar Switch Module
<i>n</i>	Number of input ports / output ports in each <i>XSU</i>
<i>l</i>	Decomposition index, $l = N/n$
<i>VOQ (i,j)</i>	Virtual Output Queue at input <i>i</i> that stores cells destined to output <i>j</i> , $0 \leq i \leq N-1, 0 \leq j \leq N-1$
<i>XSM (m)</i>	<i>m</i> th <i>XSM</i> , $0 \leq m \leq l-1$
<i>XSU (k,m)</i>	<i>k</i> th <i>XSU</i> in <i>XSM (m)</i> , $0 \leq k \leq l-1$
<i>SDB (h,k,m)</i>	<i>h</i> th <i>SDB</i> in <i>XSU (k,m)</i> , $0 \leq h \leq n-1$

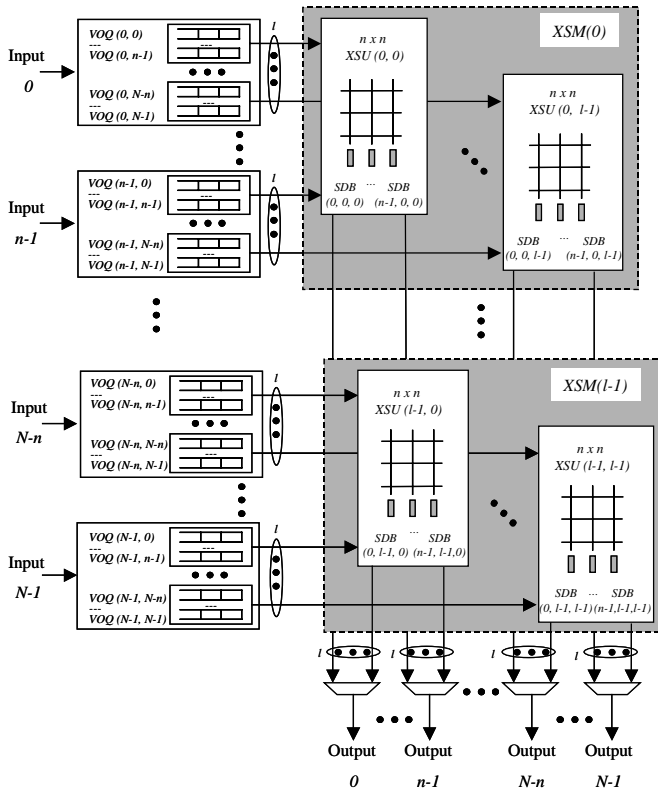


Fig. 1.  $N \times N$  Crossbar Matrix Switch

A CMS has  $N$  input ports and  $N$  output ports. An input has  $N$  VOQs to eliminate a Head-Of-Line

(HOL) blocking problem. An input has  $l$  switch interface ports, each of which is connected to a dedicated *XSU*. An *XSM* has  $l$  *XSUs*, each of which is an  $n \times n$  crossbar switch unit with single-cell scheduling decomposition buffers (*SDBs*) at the outputs. An *XSM* is connected to  $n$  input ports. Each output is connected to  $l$  *XSUs* and receives one cell from *XSUs* in one cell time slot. Equivalent output buffer for each output is given by

$$EOB [j] = \sum_{k=0}^{l-1} SDB(j \bmod n, k, \left\lfloor \frac{j}{n} \right\rfloor), \quad 0 \leq j \leq N-1 \quad (1)$$

Total output buffer for  $N \times N$  switch with decomposition index  $l$  is given by

$$\sum_{j=0}^{N-1} EOB [j] = \binom{N}{n}^2 \times n = \frac{N^2}{n}, \quad 1 \leq n \leq N \quad (2)$$

### B. Scheduling

Figure 2 shows an example of hierarchical arbitration mechanism for an  $N \times N$  switch that is composed of  $2 \times 2$  *XSU* of which size is  $N/2 \times N/2$  crossbar switch. The scheduling is based on a round-robin matching algorithm for implementation feasibility.

In each time slot the scheduling consists of two separate arbitrations: (1) 1<sup>st</sup> tier arbitration, and (2) 2<sup>nd</sup> tier arbitration. Each *XSU* performs 1<sup>st</sup> tier arbitration and each output performs 2<sup>nd</sup> tier arbitration. Those two arbitrations can be run concurrently without internal speed-up.

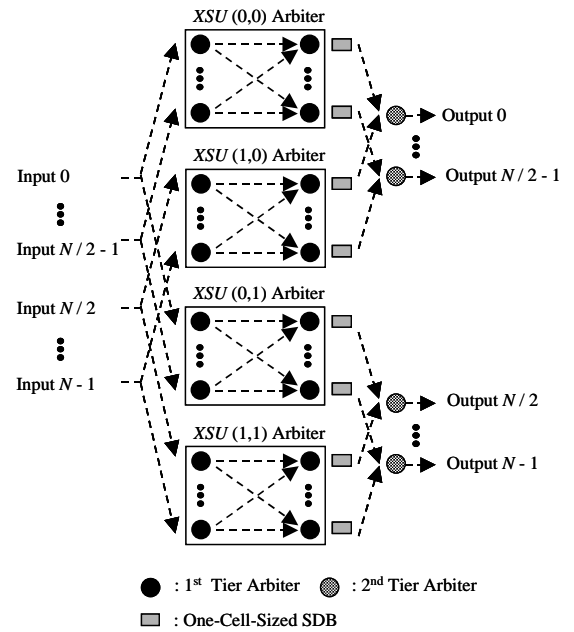


Fig. 2. An example of Hierarchical Arbitration

The 1<sup>st</sup> tier arbitration is a variation on iSLIP[1]. A form of credit-based SLIP (C-SLIP) is used for the 1<sup>st</sup> tier arbitration. Each output of XSU has an associated credit, which is used as a flag for the state of the SDB (1=full, 0=empty). The operations of C-SLIP are as follows:

**Step 1. Request.** In each time slot, non-empty VOQs send a request to every output link arbiter.

**Step 2. Grant.** Each output link arbiter chooses one request in a round-robin fashion starting from the highest priority elements if its credit is 0. It then sends the grant to the selected VOQ.

**Step 3. Accept.** If an input receives grants, it accepts the one in a round-robin fashion starting from the highest priority elements.

The 2<sup>nd</sup> tier arbitration is based on a pure round robin selection. The operations are as follows:

**Step 1. Request.** In each time slot, non-empty SDBs send a request to every output arbiter.

**Step 2. Accept.** Each output arbiter chooses one request in a round-robin fashion starting from the highest priority elements.

With the decomposition effect, each XSU can have more arbitration time margin than an  $N \times N$  crossbar switch, and the contention probability at inputs and outputs can be decreased.

For high-speed switch fabric, distributed arbitration is one of the key design issues to reduce arbitration complexity. The CMS architecture can provide a novel distributed arbitration mechanism because the arbitration process is distributed into XSUs and outputs, and those two arbitrations can be run in parallel..

### 3. SIMULATION RESULTS

The performance of CMS is evaluated through simulations for various switch size. The notation used in this section is as follows.

$l \times l$ CMS	Crossbar Matrix Switch with decomposition index $l$
OB	Output Buffered Switch

We are interested in a case of single iteration or no speed-up arbitration, for example 1-SLIP. However, the results of 4-SLIP simulation are used as a comparison reference for the proposed algorithm. In this section, we show the performance evaluation for uniform i.i.d Bernoulli traffic, burst traffic, and unbalanced traffic.

#### A. Uniform Traffic

The traffic pattern is uniform i.i.d Bernoulli arrivals. Figure 3 shows the comparison for average latency performance for scale-up CMS. The performance of 2x2 CMS is much better than that of 1-SLIP even if the switch size is 64x64. The performance of 4x4 CMS with a size of 128x128 is better than that of 2x2 CMS with a size of 64x64 for any case. The 4x4 CMS provides delay performance similar to OB switch.

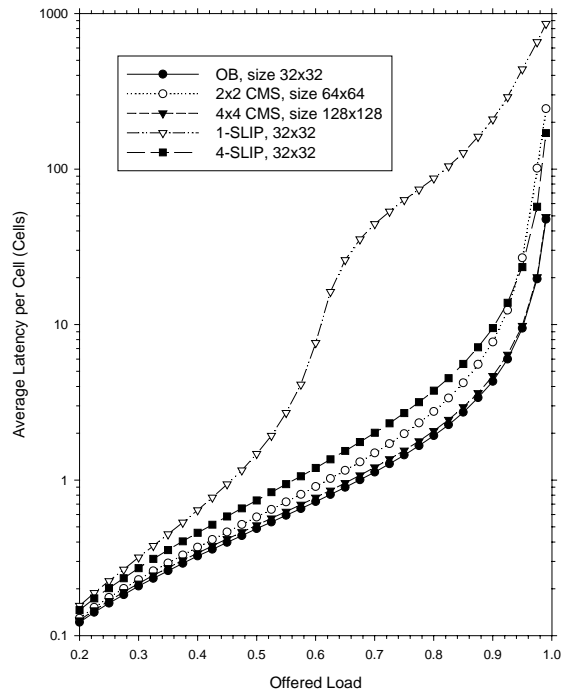


Fig. 3. Comparison of average latency for scale-up CMS

Figure 4 shows the comparison for average latency performance in the scale-down CMS case. The performance of 2x2 CMS is much better than for 1-SLIP with the same size. The performance of 4x4 CMS is better than for 2x2 CMS for any case.

From the results of scale-up and scale-down CMS simulations, we can find that decomposition index is a main factor that affects the total average latency. We can observe that the throughput is unaffected for any case.

The reason for the performance improvement is that the number of paths between inputs / outputs and switch fabric increases in proportion to the decomposition index  $l$ , and also each XSU output can have maximum  $l$  (decomposition index) times more chances for arbitration than for 1-SLIP.

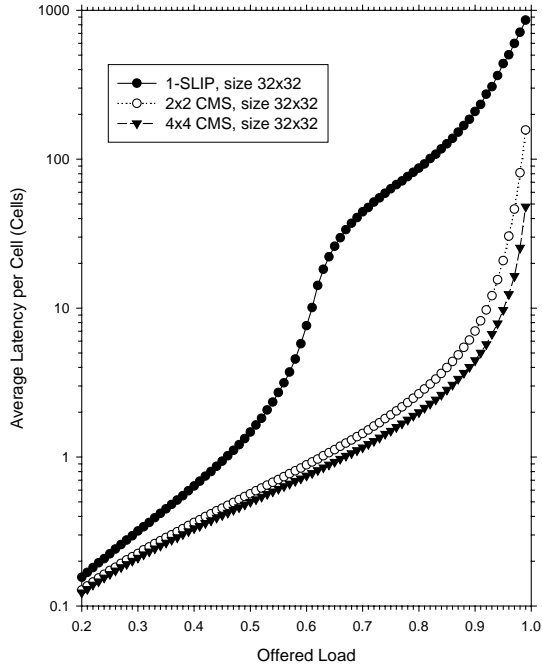


Fig. 4. Comparison of average latency for scale-down CMS

### B. Burst Traffic

Burst traffic is modeled as an *on-off* arrival process where the *on* and *off* interval lengths are exponentially distributed with different parameters. The source alternately generates a burst of cells followed by an idle period of no cells. During the *on* period cells are generated at the link rate and the destined output ports of the cells are identical. The average burst length is set to 16 cells.

Figure 5 shows the average latency performance of 2x2 CMS, 1-SLIP, and OB for burst traffic. The average delay is close to that of OB and much better than that of 1-SLIP under burst traffic at any load, and it is slightly longer than that of 1-SLIP under Bernoulli traffic for  $\rho \geq 0.7$ . The throughput is not affected at any load.

### C. Unbalanced Traffic

We use the same model as given in [9]. We define non-uniform traffic by using an unbalanced probability  $w$ . Let us consider input port  $i$ , output port  $j$ , and the offered load  $\rho$  for each input port. The traffic load from input port  $i$  to output port  $j$ ,  $\rho_{i,j}$  is given by,

$$\rho_{i,j} = \begin{cases} \rho(w + \frac{1-w}{N}) & \text{if } i = j \\ \rho \frac{1-w}{N} & \text{otherwise} \end{cases} \quad (3)$$

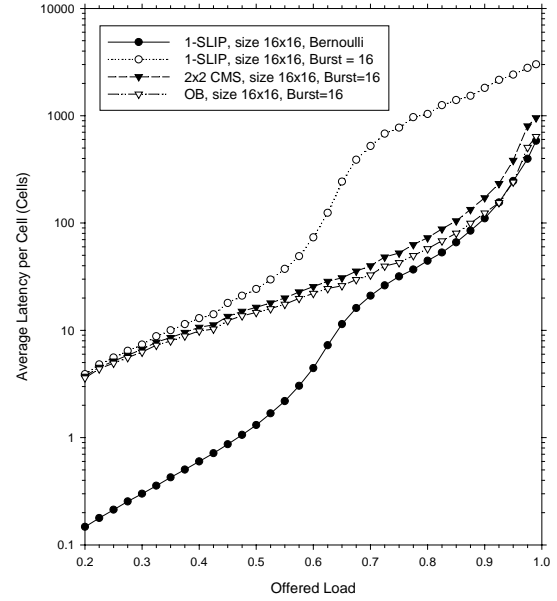


Fig. 5. Comparison of average latency under burst traffic

where  $N$  is the switch size. Here, the aggregate offered load that goes to output  $j$  from all input ports,  $\rho_j$  is given by ,

$$\rho_j = \sum_{i=0}^{N-1} \rho_{i,j} = \rho(w + N \times \frac{1-w}{N}) = \rho \quad (4)$$

For  $w = 0$ , the offered load is uniform. On the other hand, for  $w = 1$ , the traffic is completely directional from input  $i$  to output  $j$ . This means that all the traffic of input port  $i$  is destined to output port  $j$  only, where  $i = j$ .

The throughput of CMS and iSLIP for unbalanced traffic is shown in Figure 6. The throughput of 2x2 CMS, which consists of 4 XSUs of size 16x16, is always better than that of 1-SLIP and 4-SLIP. The throughput of 4x4 CMS, which consists of 16 XSUs of size 16x16, is much better than that of 4-SLIP. The throughput is almost 100% when the  $w$  is about zero or one. Speed-up or maximum weighted matching algorithm is necessary to provide 100% throughput for unbalanced traffic.

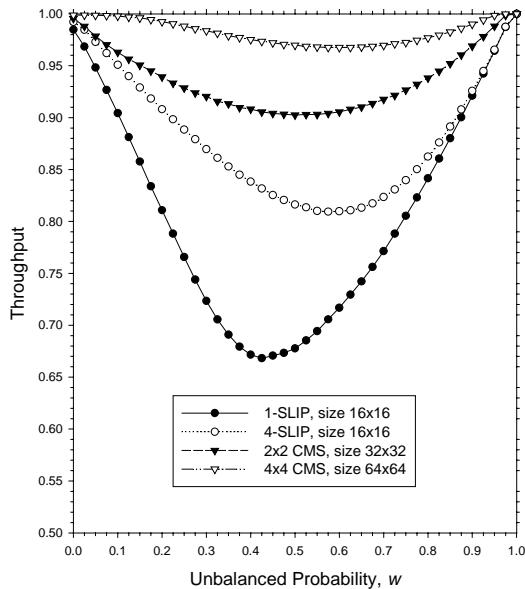


Fig. 6. Comparison of throughput under unbalanced traffic

#### 4. CONCLUSION

We have introduced a scalable two-dimensional crossbar matrix switch architecture and its hierarchical scheduling algorithm. We have shown that CMS yields several advantages for implementation feasibility, such as arbitration timing relaxation, distributed arbitration, scalability, and a better delay performance than a crossbar switch with iSLIP. The feature for distributed arbitration and scalability is a very important requirement for high-speed switch implementation.

The two-dimensional CMS switch can offer a very close average latency to that of output buffered switch under uniform and burst traffic. We also observe that decomposition index is a main factor that affects the total average latency.

A crossbar with a single-cell SDB at each output port is a very feasible solution because the memory size of an SDB is very small and the number of pins remains the same as the case for a non-buffered crossbar.

#### Acknowledgement

We gratefully acknowledge Seong-Joon Kim for his comments and help in performance simulation.

#### 5. REFERENCES

- [1] T. Anderson, S. Owicki, J. Saxe, and C. Thacker, "High speed switch scheduling for local area networks," *ACM Trans. Comput. Syst.*, vol.11, no. 4, pp. 319-352, Nov. 1993.
- [2] N. McKeown, "The iSLIP Scheduling Algorithm for Input-Queued Switches," *IEEE/ACM Trans. Networking*, vol. 7, no. 2, pp. 188-200, Apr. 1999.
- [3] D. N. Serpanos and P. I. Antoniadis, "FIRM: A Class of Distributed Scheduling Algorithms for High-speed ATM Switches with Multiple Input Queues," in *Proc. IEEE INFOCOM 2000*, vol. 2, pp. 548-555, 2000.
- [4] Hsin-Chou Chi and Yuval Tamir, "Decomposed Arbiters for Large Crossbars with Multi-Queue Input Buffers," *IEEE International Conference on Computer Design: VLSI in Computers and Processors*, pp.233-238, 1991.
- [5] Richard O. LaMaire and N. Serpanos, "Two-Dimensional Round-Robin Schedulers for Packet Switches with Multiple Input Queues," *IEEE/ACM Trans. Networking*, vol. 2, no. 5, Oct. 1994.
- [6] H. J. Chao and J-S Park, "Centralized Contention Resolution Schemes for a Large-Capacity Optical ATM switch," *Proc. IEEE ATM Workshop '97*, Fairfax, VA, pp.10-11, May 1998.
- [7] N. McKeown, M. Izzard, A. Mekkittikul, B. Ellersick, and M.Horowitz, "The tiny tera: A small high-bandwidth packet switch core," *IEEE Micro*, vol. 17, pp. 26-33, Jan.-Feb. 1997.
- [8] Seung Yeob Nam and Dan Keun Sung, "Decomposed Crossbar Switches with Multiple Input and Output Buffers," *GLOBECOM 2001. IEEE*, vol 4, pp. 2661 -2665, 2001.
- [9] R. Rojas-Cessa, E. Oki, Z. Jing, and H. J. Chao, "CIXB-1: Combined Input-One-Cell-Crosspoint Buffered Switch," *Proc. 2001 IEEE Workshop on High Performance Switching and Routing (HPSR)*, pp. 324-329, May 2001.