

# A Reconfigurable Crossbar Switch with Adaptive Bandwidth Control for Networks-on-Chip

Donghyun Kim, Kangmin Lee, Se-joong Lee, and Hoi-Jun Yoo

Semiconductor System Laboratory, Department of Electronic Engineering and Computer Science  
Korea Advanced Institute of Science and Technology (KAIST)

Daejeon, Republic of Korea  
donghyun53@eeinfo.kaist.ac.kr

**Abstract** — We propose a new crossbar switch structure with adaptive bandwidth control. In a complex SoC design, the proposed crossbar switch efficiently incorporates various IPs with different bandwidth requirements. Simulation under various traffic scenarios shows that the throughput of the proposed crossbar switch is as high as that of conventional switch operating at twice the speed. The proposed crossbar switch shows maximum 27% improvement in throughput and maximum 41% improvement in latency compared to the conventional one. The proposed crossbar switch is implemented using verilog HDL, synthesized with 0.18um process library, and verified on FPGAs. The area and power overhead of the proposed crossbar switch is 21% and 15%, respectively, when compared to the conventional crossbar switch.

## I. INTRODUCTION

Before the end of this decade, by using 50nm process technology, a complex System-on-Chip (SoC) will integrate 4 billion transistors running at 10 GHz [1]. As a solution to the interconnection among numerous Intellectual Properties (IPs) on an SoC, a new design paradigm called Networks-on-Chip (NoC) has been proposed [2-3]. NoC design interconnects various IPs by structured on-chip packet switched network, instead of ad-hoc dedicated routing wires. Because electrical parameters of each modular network block are well defined, it is possible to increase operating frequency and reduce power consumption.

Crossbar switches are key building blocks of the NoC, which enable structured design of a complex SoC. In an SoC, the traffic condition of a crossbar switch is different from that of traditional computer networks: each input port of a crossbar switch requires different amount of bandwidth because each IP operates different function with different operating frequency. Although power and area issues of the crossbar switch are being actively studied for cost-effective NoC implementation [4-6], performance of the crossbar switch under such traffic condition is rarely studied yet. Our work is motivated by observing traffic conditions described above, which are unique to the SoC. In this paper we propose a new crossbar switch with adaptive bandwidth control for

efficient implementation of NoC under the distinctive bandwidth requirement of various IPs.

The rest of the paper is organized as follows: section II explains concept of the proposed crossbar switch. And then, section III describes detailed structure and operation of the proposed crossbar switch. In section IV, the throughput and latency of the proposed and conventional crossbar switch are evaluated. Section V presents implementation result. Finally, conclusion and summary is made in section VI.

## II. CONCEPT OF THE PROPOSED CROSSBAR SWITCH

Fig. 1 shows a structure of a conventional crossbar switch. When an input port transmits a packet to an output port, the input and output ports are connected through the cross wires and junction switches. All the input ports and the output ports have the same bit width and operation frequency, i.e. the same bandwidth, so that any input port can be connected to any of the output ports. With this simple and regular structure, the crossbar switch provides simultaneous multiple connections between the input and output ports. Providing identical bandwidth for every input port is

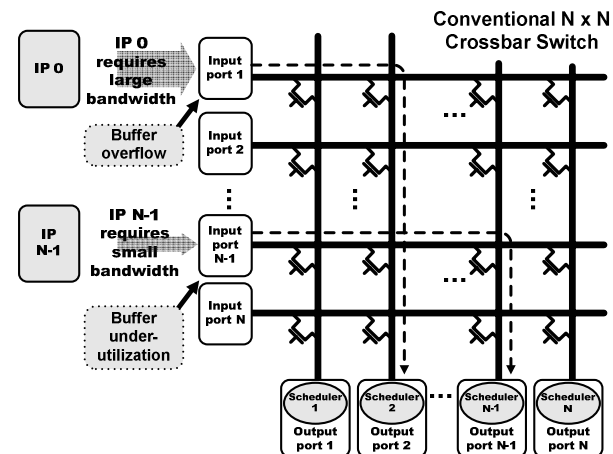


Figure 1. A conventional crossbar switch.

adequate to traditional network systems such as Ethernet and ATM, because maximum bandwidth of the input traffic is fixed by network technology adopted and traffic is regulated by statistical multiplexing [7]. However, in an on-chip situation, each IP has its own bandwidth requirement, and the number of IPs is limited to a few tens. Therefore, statistical multiplexing is not expected and considerable bandwidth variation of offered traffic is observed. As shown in Fig. 1, the structure that provides identical input port bandwidth is not adequate to on-chip network switches. When a large amount of bandwidth is required for the input port 1, input buffer overflows due to insufficient bandwidth provided by switch fabric. For the input port N-1, switch fabric is under utilized because the amount of bandwidth required is smaller than the bandwidth provided by the switch fabric. Therefore, when a crossbar switch is designed to meet maximum bandwidth requirement, waste of power or silicon area occurs. On the other hand, if a crossbar switch is designed for minimum required bandwidth, performance of the system is degraded significantly.

To resolve such a problem of conventional crossbar switch, we propose a crossbar switch structure with an additional bus. The structure of the proposed crossbar switch is motivated by observing traffic characteristics of an SoC presented in Fig. 2. Fig. 2-(a) shows conceptual representation of bandwidth requirement versus time for three IPs, and Fig. 2-(b) shows average bandwidth requirement of the IPs. The point of the proposed idea is as follow: for various IPs, crossbar switch fabric provides the common minimum required bandwidth. And then, remaining portions of required bandwidth, such as peak bandwidth, is provided by time sharing of the additional bus. Because IPs on an SoC do not continuously utilize their maximum bandwidth as shown in Fig. 2-(a) [8], time-sharing of the additional bus compensates variation of required bandwidth of various IPs. In the following sections we will detail the implementation of the design concept and evaluate the benefits.

### III. CROSSBAR SWITCH DETAIL

Fig. 3 shows the structure of the proposed  $N \times N$  crossbar switch. The difference of the proposed crossbar switch compared to the conventional one is an additional bus which can be scheduled to any of input ports to provide extra bandwidth dynamically. Because the bus has the same bitwidth as other switch fabric ports, total bandwidth of the proposed  $N \times N$  crossbar switch is  $(N+1)/N$  times of the conventional  $N \times N$  crossbar switch. A work conserving bus scheduler grants the additional bus to one of the input ports in a way such that the extra bandwidth is distributed according to the amount of required bandwidth.

The operation of the proposed crossbar switch is also shown in Fig. 3. Before describing operation of the proposed crossbar switch, we assume that the bandwidth of the switch fabric is designed to meet the average required bandwidth of

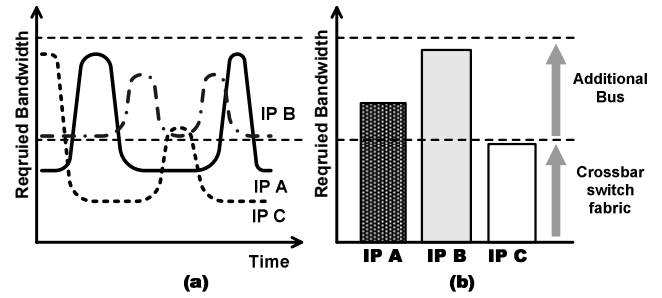


Figure 2. Traffic characteristics of an SoC: (a) Bandwidth utilization of each IP versus time, (b) Average bandwidth requirement of each IP.

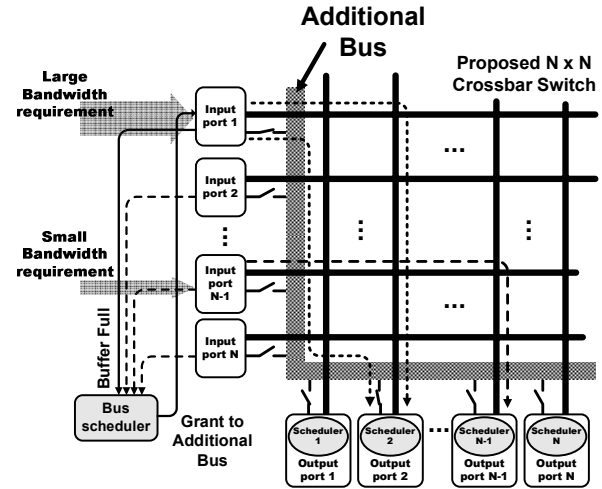


Figure 3. The proposed crossbar switch: the bandwidth of the input port 1 is doubled temporarily.

the input ports. If the input port 1 in the Fig. 3 is heavily loaded, input buffers in the input port 1 become full because the rate of incoming packets is higher than the rate of outgoing packets. In this case, the bus scheduler immediately detects the congestion and schedules the additional bus to the input port 1 to accelerate the packet transfer. If the IP which is connected to port 1 is idle or not using its full bandwidth, the additional bus can be scheduled to the other ports. When more than two ports request the additional bus, the bus scheduler grants them in round-robin fashion. In case of the input port gets a grant from the bus and a grant from the switch fabric too, the crossbar switch transfers packets for that connection using both switch fabric and the additional bus. Therefore, while possessing a grant from the bus, the bandwidth between the input and output ports becomes twice the usual bandwidth because the available number of wires is doubled. For each port, effective bandwidth is determined by the utilization of the additional bus.

The proposed crossbar switch dynamically increases the bandwidth of the each input port only for the necessary period of time. Therefore the bandwidth of the proposed crossbar switch is effectively doubled, even though the total bandwidth of  $N \times N$  proposed crossbar switch is only increased to  $(N+1)/N$  times the conventional crossbar switch.

#### IV. PERFORMANCE EVALUATION

To evaluate the performance of the proposed crossbar switch, we perform simulation under various traffic conditions. The evaluation is performed for 8 x 8 crossbar switches. The first simulation is carried with different combinations of traffic generators representing various traffic conditions on an SoC. For the second, we apply the proposed crossbar switches to the MPEG-4 decoding system which is mapped on 3 x 5 mesh NoC.

In the simulation, we used two kinds of traffic generators,  $G_{fast}$  and  $G_{slow}$ . The  $G_{fast}$  operates at twice the speed of  $G_{slow}$ , therefore generate packets in twice the rate. The destined output ports of the generated packets are randomly and uniformly distributed. The combinations of traffic generators for the performance evaluation are listed in Table I. From the case 1 to the case 3, total bandwidth required by the traffic generators is increased in equal steps. In the case 4, operating frequencies of the eight traffic generators are spreaded in equal step, in the range between the operating frequencies of  $G_{slow}$  and  $G_{fast}$ . The total required bandwidth for case 4 is the same as in the case 2. These different combinations of traffic generators with different required bandwidth present various traffic conditions on an SoC. For the traffic conditions, the throughput and latency of three kinds of crossbar switches are evaluated. The three crossbar switches are as follows: (1)  $SW_{conv1X}$  - The conventional crossbar switch operating at the speed of  $G_{slow}$ . (2)  $SW_{conv2X}$  - The conventional crossbar switch operating at the speed of  $G_{fast}$ . (3)  $SW_{prop}$  - The proposed crossbar switch operating at the speed of  $G_{slow}$ .

Fig. 4 shows the throughput comparison of the three switches. If the offered load is lower than 50%, without regard to the traffic patterns,  $SW_{prop}$  outperforms the  $SW_{conv1X}$ , and shows similar performance with  $SW_{conv2X}$ . With the offered load higher than 50%, the throughput of the proposed crossbar begins to saturate because the total bandwidth required by the traffic generators exceeds the maximum sustainable bandwidth of the proposed crossbar switch. However, the proposed crossbar switch still outperforms the conventional crossbar switch operating in the same speed. The proposed crossbar switch shows maximum 27% and average 20% improvement in throughput compared to the convention crossbar switch of same operating frequency.

TABLE I. COMBINATIONS OF TRAFFIC GENERATORS FOR SIMULATION

Case	Num. of $G_{slow}$	Num. of $G_{fast}$	Total required bandwidth normalized to case 1
1	6	2	1
2	4	4	1.2
3	2	6	1.4
4	$G_{slow}, 1.14G_{slow}, \dots, 1.86G_{slow}, 2G_{slow} (=G_{fast})$		1.2

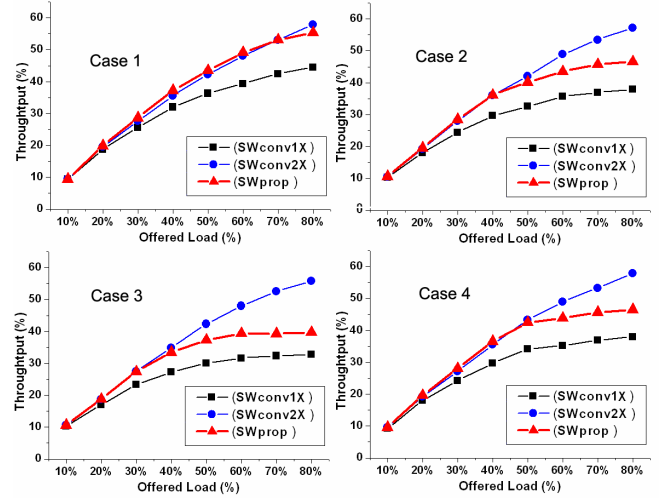


Figure 4. Throughput comparison for 3 different crossbar switches with the cases listed in table I.

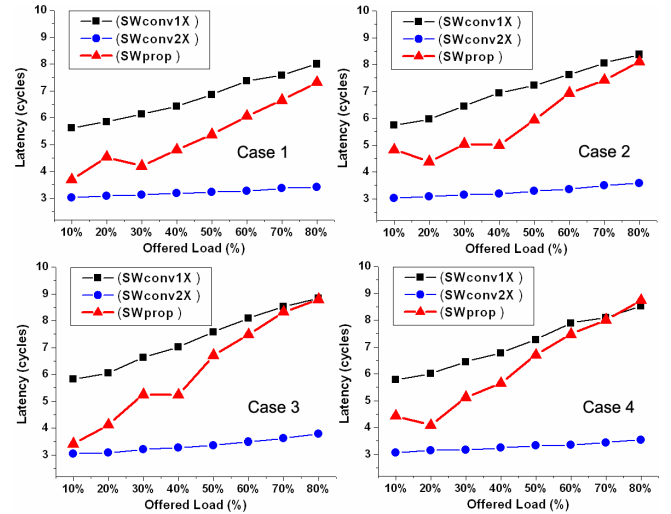


Figure 5. Latency comparison for 3 different crossbar switches with the cases listed in table I.

Fig. 5 shows the average latency comparison of the three switches. The improvement in average latency is 41% in maximum and 15% in average. Although plots are not shown in here, the simulation with packets of burst length eight is also performed. In the burst packet simulation, proposed crossbar switch achieves average 22% and 25% improvement in throughput and average latency, respectively. For burst packets, the amount of the performance improvement is slightly higher than that of random packets because the overhead of granting bus is reduced.

We also apply the proposed crossbar switch to the MPEG-4 decoder system shown in Fig. 6 [9]. MPEG-4 decoder system is modeled and mapped on 5x3 mesh network-on-chip.

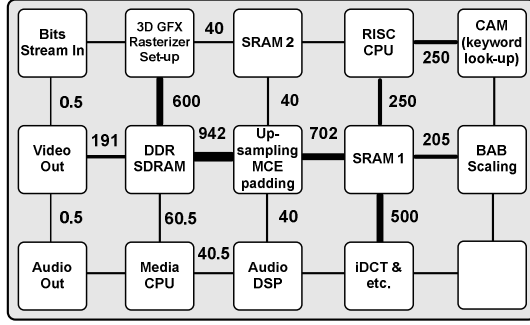


Figure 6. Mapping of MPEG-4 decoding system on a mesh NoC: Average bandwidths between to nodes are presented. (MB/s)

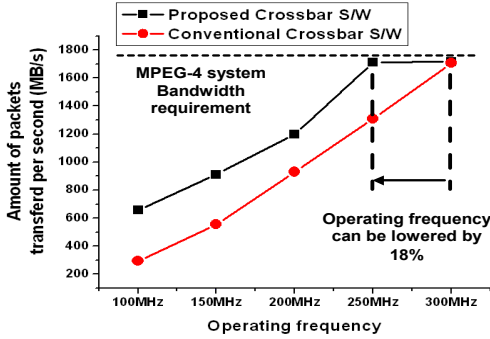


Figure 7. Performance evaluation of MPEG-4 system with the proposed and conventional crossbar switch.

In the system of Fig. 6, each NoC node has 5x5 crossbar switch; one port of the crossbar switch is connected to the functional block and other four ports are used to interconnect neighboring modules. In Fig. 6, the numbers above the link connecting two nodes present average bandwidth between them. In this case we compared the average amount of packet transferred through the crossbar switch in the every NoC node with increasing operating frequency. As an example, Fig.7 shows the result evaluated for the node of DDR SDRAM. The proposed crossbar switch satisfies bandwidth required by MPEG-4 system at 250MHz of operating frequency. On the other hand, the conventional crossbar switch provides the same bandwidth at operating frequency of 300MHz. By applying the proposed crossbar switch to the NoC the operating frequency of the overall system is lowered by 18%.

## V. IMPLEMENTATION RESULT

We implement the proposed crossbar switch using verilog HDL. The proposed crossbar switch including synchronization units to interconnect IPs of different operating speed is verified on the ALTERA STRATIX series FPGAs. To compare the power consumption and area we also synthesized the proposed crossbar switch using 1-poly, 6-metal 0.18um process library. The synthesized results for the 8 x 8 proposed crossbar switch and the conventional crossbar switch are shown in Table II. The proposed crossbar

TABLE II. SYNTHESIZED RESULT COMPARISON

	Conventional Crossbar switch	Proposed crossbar switch
Estimated Area (unit inverters)	70524	85401
Maximum Operating frequency	418 MHz	406 MHz
Power Consumption (at 400MHz)	66.8 mW	76.7 mW

switch shows 21% area overhead when compared to conventional crossbar switch. Under the maximum utilization of the switch fabric, power overhead of the proposed crossbar switch is 15% for the proposed crossbar switch. The maximum operating frequency of the proposed crossbar switch is only 3% lower than the conventional crossbar switch.

## VI. CONCLUSION

We proposed the crossbar switch with adaptive bandwidth control for NoC design. The proposed crossbar switch dynamically schedules the extra bandwidth obtained by the additional bus to the port of congestion. Therefore, the IPs with different bandwidth requirement can be efficiently interconnected. Simulations under various traffic conditions show that the proposed crossbar switch performs effectively as much as conventional crossbar switch operating at twice the speed. The maximum improvement in throughput and latency of the proposed crossbar switch over the conventional switch is 27% and 41% respectively.

## REFERENCES

- [1] ITRS, International Technology Roadmap for Semiconductors, update 2003, <http://public.itrs.net>.
- [2] Luca Benini and Giovanni De Micheli, "Networks on Chips : A New SoC Paradigm," IEEE computer, vol. 35, pp. 70-78, 2002.
- [3] Jorg Henkel, Wayne Wolf, and Srimat Chakradhar, "On-chip networks : A scalable, communication-centric embedded system design paradigm," IEEE Proceedings of 17<sup>th</sup> international Conference on VLSI, pp. 845-851, 2004.
- [4] Kangmin Lee, Se-Joong Lee, and Hoi-Jun Yoo, "A Distributed Crossbar Switch Scheduler for On-Chip Networks," IEEE Proceedings of Custom Integrated Circuits Conference, pp. 671-674, 2003.
- [5] Kangmin Lee, et al., "A 51mW 1.6GHz On-Chip Network for Low-power Heterogeneous SoC Platform," IEEE Digest of International Solid State Circuits Conference, vol. 1, pp. 152-153, 2004.
- [6] Se-Joong Lee, et al., "An 800MHz Star-Connected On-Chip Network for Application to Systems on a Chip," IEEE Digest of International Solid State Circuits Conference, vol. 1, pp. 468-469, 2003.
- [7] Wu-chun Feng, "Networks Traffic Characterization of TCP," Military Communications Conference proceedings, vol. 1, pp. 22-25, 2000.
- [8] Girish Varatkar, Radu Marculescu, "Traffic Analysis for On-chip Networks Design of Multimedia Applications," 39<sup>th</sup> Proceedings of Design Automation Conference, pp. 795-800, 2002.
- [9] E.B. Van der Tol, E.G.T. Jaspers, "Mapping of MPEG-4 Decoding on a Flexible Architecture Platform," SPIE, 2002, pp. 1-13, 2002.