# A 118.4GB/s Multi-Casting Network-on-Chip for Real-Time Object Recognition Processor

Joo-Young Kim, Kwanho Kim, Seungjin Lee, Minsu Kim, Jinwook Oh, and Hoi-Jun Yoo
School of EE&CS, Korea Advanced Institute of Science and Technology (KAIST)
373-1, Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea
E-mail: trample7@eeinfo.kaist.ac.kr

***Abstract*— A 118.4GB/s multi-casting network-on-chip (MC-NoC) is developed as communication platform for a real-time object recognition processor. To support application-specific data transactions, the MC-NoC adopts the combination of hierarchical star and ring topology with the multi-casting capability. As a result, the proposed MC-NoC improves data transaction time and energy consumption by 20% and 23%, respectively, under target object recognition traffic. The 350k gates MC-NoC, fabricated in a 0.13μm CMOS process, consumes 48mW at 400MHz, 1.2V.**

## I. INTRODUCTION

Recently, Network-on-Chip (NoC) has become widely applied to communication architectures of multi-core System-on-Chips (SoCs) with more than 10 IP blocks [1]. Object recognition [2], which is a key technology for artificial visions, also adopts a multi-core approach to satisfy its real-time requirements. Some of previous multi-core processors [3-4] start to use the NoC as communication architecture for object recognition applications. However, these NoCs have just provided basic interconnection functionalities to overall systems, without considerations for application-specific data transactions of object recognition.

Fig. 1 shows the data transaction flows of the 3-stage object recognition processor with more than 20 IP blocks including 16 processing elements (PEs) [5]. At the pre-processing stage, a pre-processor computes expected interest regions from the input image and generates image tile tasks. And a scheduler distributes >0.5Mb of instruction kernels and >2Mb of pixel data to the 16 PEs. In the main stage, the 16 PEs generate descriptor vectors and transfer them to a post processor. In this process, the PEs communicate >200Kb intermediate data among themselves and generate >100Kb data transactions to the post processor. In the post-processing stage, the post processor performs vector matching for the aggregated descriptor vectors with object database for the final recognition. To achieve efficient data flows in the proposed object recognition processor, 5 special requirements should be satisfied. First, low latency data transactions are required for real-time processing. Second, 1-to-N data transactions are required in cases the pre-processor distributes instruction kernels, pixel data, and image tile tasks to the 16 PEs. Third, sufficient communication channels are required among the PEs for their intensive data communications. Fourth, N-to-1 data transactions are required for aggregation of data from the 16 PEs. Lastly, the overall NoC should support data synchronizations among different clock domains of IP blocks.

The conventional mesh architecture, depicted in Fig 2 (a), is not suitable for above NoC requirements because its long latency cannot provide fast data transactions among IP blocks. In addition, its large area and power overhead can be a significant burden to overall system [6]. On the other hand, the hierarchical star (H-star) topology of Fig 2 (b) is more suitable for object recognition data flows than mesh topology because of its short latency. It is adopted in the object recognition processor of [7], which includes sequential object recognition data flows. However, this conventional H-star topology is not sufficient for the proposed object recognition processor that requires concurrent data transactions between the neighbor

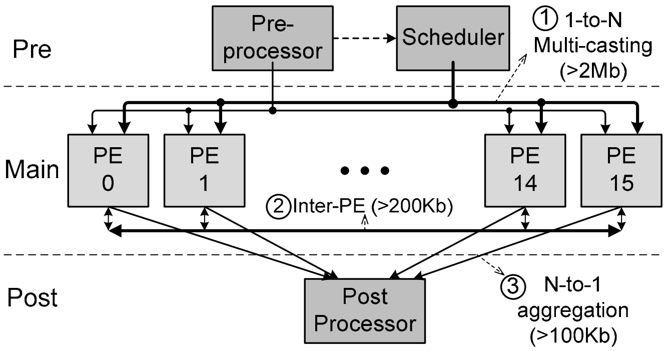


Fig. 1 Data transaction flows of 3-stage recognition processor

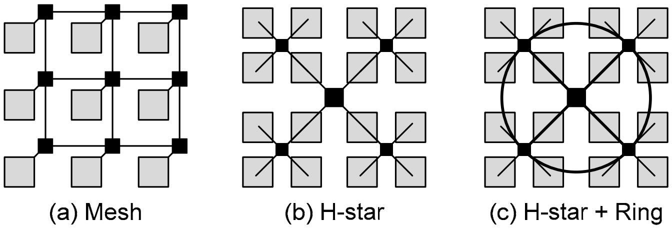

(a) Mesh
(b) H-star
(c) H-star + Ring

Fig. 2 Conceptual diagram of NoC topologies

978-1-4244-4353-6/09/$25.00 ©2009 IEEE

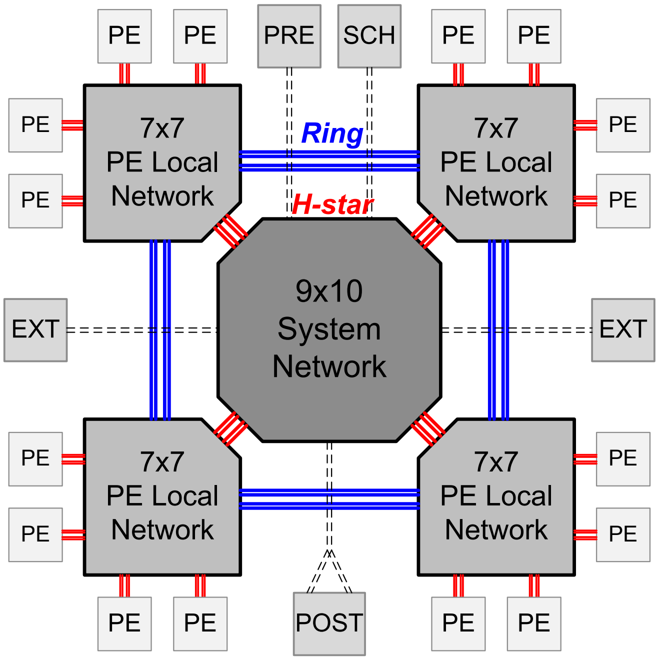


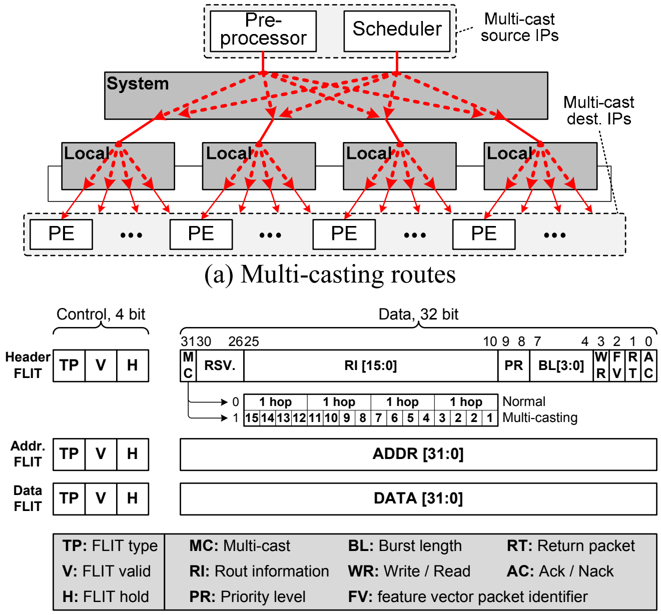
Fig. 3 Proposed MC-NoC Architecture

stages, due to its pipelined operations. It also cannot provide sufficient communication channels among the 16 PEs.

In this paper, we propose a multi-casting Network-on-Chip (MC-NoC) to resolve data transaction requirements of the real-time object recognition processor with pipelined operation. In the MC-NoC, the combination of the H-star and ring topology, whose concept diagram is shown in Fig. 2 (c), is adopted. While the H-star topology provides low latency data transactions, additional ring networks facilitate data communications among the 16 PEs. The multi-casting capability is proposed to accelerate 1-to-N data transactions such as program kernel and image data distribution. Each input port of a switch includes a heterogeneous clock interface to support multiple clock domains, and an output port of one switch is customized for the N-to-1 data transactions.

## II. MC-NoC Arhictecture

Fig. 3 shows the proposed MC-NoC architecture that consists of a 9x10 system network and four 7x7 PE local networks. The 16 PEs are connected to the system network through the 4 local networks while the rest of IP blocks such as the pre-processor, scheduler, post-processor, and 2 external interfaces are directly connected to the system network. The proposed MC-NoC adopts a hierarchical star topology as basic starting topology for low latency data communications, and then, supplements a ring topology to the local networks for high speed circular data transactions. The additional ring network links for the combined topology provide a total of 25.6 GB/s data bandwidth between the local networks, and allow each PE to access the PEs in neighbor local networks in 2 switch hops. In addition, in the system network, the output port to the post-processor is customized to dual ports for the N-to-1 data transactions. One port is dedicated to aggregate specially notified packets from the PEs. In overall, topology-combined architecture of the proposed MC-NoC provides 118.4GB/s total bandwidth with the switch hop latency of less than 3. In the MC-NoC, two multi-casting ports are supported for the pre-processor and scheduler to provide 1-to-N data transaction capability to the 16 PEs. Since the mentioned 3-stage object recognition data flows designate the multi-casting source and destination IP blocks to the pre-processor and scheduler, and the 16 PEs, respectively, the number of multi-casting ports in the MC-NoC is also set to two for hardware efficiency and protocol compactness. Fig. 4 (a) shows the multi-casting routes in the MC-NoC and Fig. 4 (b) shows the packet format of the MC-NoC based on wormhole routing protocol. The packet is composed of header, address, and data flow control units (FLITs) where each of them consists of 4-bit control and 32-bit data signals. The header FLIT contains all information for the entire packet routing including 4-bit burst length for burst data transaction up to 8 FLITs and 2-bit priority for quality-of-service. The 16-bit source defined routing information (RI) indicates where the packet should be routed to go to the final destination in each network switch. It allows 4 switch traversals for normal packets, and multi-



(a) Multi-casting routes

| | | | |
|---|---|---|---|
| **TP:** FLIT type | **MC:** Multi-cast | **BL:** Burst length | **RT:** Return packet |
| **V:** FLIT valid | **RI:** Rout information | **WR:** Write / Read | **AC:** Ack / Nack |
| **H:** FLIT hold | **PR:** Priority level | **FV:** feature vector packet identifier | |

(b) Packet format

Fig. 4 Multi-casting NoC

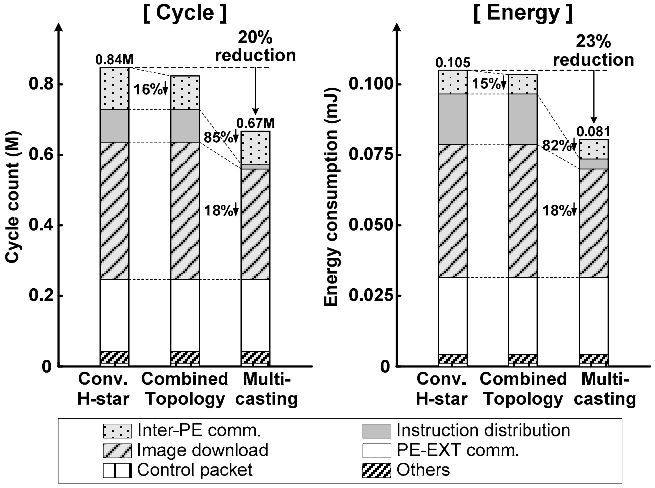


Fig. 5 Experimental results

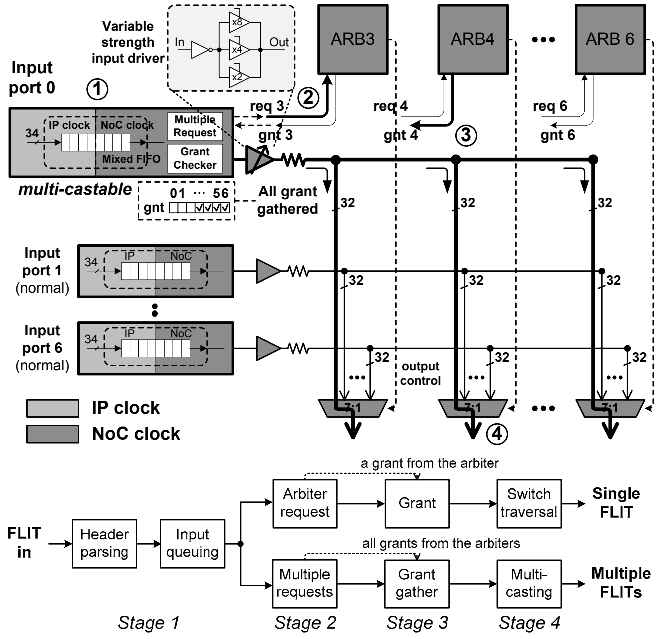


Fig. 6 4-stage pipelined multi-casting switch

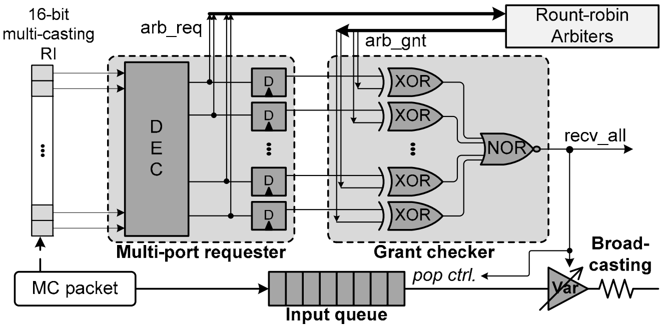


Fig. 7 Multi-port requester and grant checker

casting to arbitrary PEs for multi-casting packets. In the case of multi-casting packets, each bit of 16-bit RI indicates the destination PE. Since the multi-casting paths are determined according to the destination PEs, each multi-casting port decodes the multi-casting RI and generates the request signals for correct multi-routing. To evaluate the proposed MC-NoC, cycle count and energy consumption are measured when the target object recognition task is running for a one frame of VGA (640x480) sized video input. Fig. 5 shows the cycle count and energy reduction effect by the proposed combined architecture and multi-casting capability. Specifically, the combined architecture contributes to inter-PE communication, and the multi-casting contributes to program distribution and image download. As a result, the overall cycle count and energy consumption is reduced by 20% and 23%, respectively. Thanks to the short routing channels and packet multi-casting operations, redundant packet buffering and transmissions in network interconnections are removed.

## III. DETAILED CIRCUITS DESIGN

### A. Multi-Casting Crossbar Switch

Fig. 6 shows the block diagram of the 4-stage pipelined multi-casting crossbar switch. The incoming FLITs are buffered at the 8-depth FIFO queue that contains synchronization interface for heterogeneous clock domain conversion. Then, each active input port sends a request signal to the destination arbiter to get a grant signal to traverse the crossbar switch. For scheduling of grant signals, the arbiters perform a simple round-robin scheduling according to the priority levels of request packets. In case of the multi-casting packet, the input port sends multiple requests to all destination arbiters at the same time and waits until all the grant signals are returned. To this end, multi-port requester and grant checker are employed in multi-casting port as shown in Fig. 7. The multi-port requester decodes the 16-bit RI and generates corresponding request signals, and the grant checker holds the multi-casting packet in the FIFO not to be sent until all latched request signals become equal to the received grant signals. After all grants are gathered, the multi-casting is performed using the existing broad-casted wires of mux-based crossbar fabric without requiring any additional wires. A variable strength driver is specially employed in the multi-casting port to provide sufficient driving strength for multi-casting.

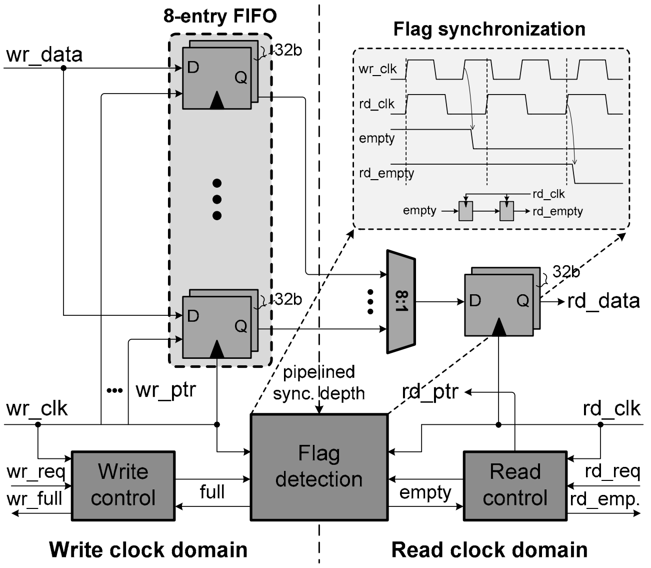


Fig. 8 Heterogeneous clock interface

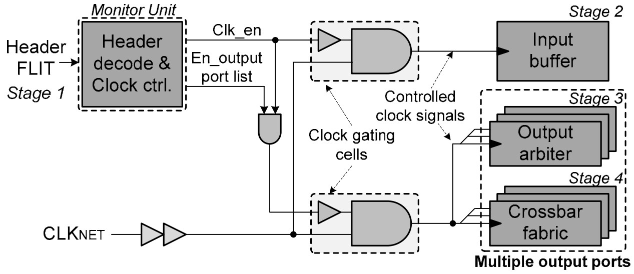
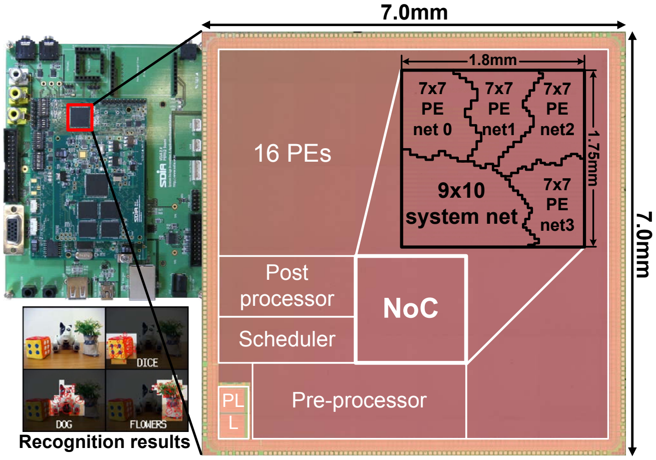


Fig. 9 Fine-grained clock gating

### B. Heterogeneous Clock Interface

To allow the different clock frequencies between the IP blocks and the MC-NoC, the first-in-first-out (FIFO) based synchronization is performed in each input port of the crossbar switch as shown in Fig. 8. It includes 8-depth FIFO buffers whose data are managed by write and read pointers. The



(a) Micrograph

| | |
|---|---|
| Process Technology | 0.13mm 1P 8M CMOS |
| Die Size | 7mm x 7mm |
| Power Supply | 1.2V core, 2.5V I/O |
| Operating Frequency | 200MHz IPs / 400MHz NoC |
| Transistor Counts | 36.4M transistors<br>3.73M gates / 396KB SRAM |
| Power Consumption | 496mW (average) |
| Topology | Hierarchical star + Ring |
| Transistor Counts | 350k gates |
| Total Bandwidth | 118.4 GB/s (H-star:92.8GB/s, Ring:25.6GB/s) |
| Latency | Less than 3 switch hops |
| Operating Frequency | 400MHz |
| Power dissipation | 48mW |
| Protocol | Multi-casting support / Wormhole routing<br>Burst packet transmission (up to 8)<br>2 priority levels |
| Queuing model | Input queuing (depth 8) |
| Interface | Heterogeneous clock interface<br>(FIFO based synchronization) |

(b) Summary

Fig. 10 Chip implementation results

empty and full flag are generated by combinational flag detector using relative locations between the write and read pointer. When the detected flag signals are employed in an either domain, they are synchronized with the clock of that domain by a simple pipelined synchronization method [8].

### C. *Fine-grained Clock Gating*

For low power consumption, fine-grained clock gating is applied to each component of the switch as shown in Fig. 9. The always turn-on monitor unit decodes the incoming header packet and generates the output port list to activate the clock of the input buffer, output arbiter, and crossbar fabric sequentially. By activating only required paths of the routing switch in packet transmission, the fine-grained clock gating achieves 28% power saving on average.

## IV. Implementation Results

The proposed MC-NoC for real-time object recognition processor is fabricated in a 0.13µm 8 metal CMOS process. The overall recognition processor has 49mm$^2$ die area including 3.73M gates and 396KB on-chip SRAM, while the MC-NoC accounts for 350k gates. The chip micrograph and the evaluation board are shown in Fig. 10 (a). The operating frequency is 200MHz for IP blocks and 400MHz for MC-NoC. The MC-NoC provides 118.4GB/s total bandwidth (92.8 GB/s from hierarchical star topology, 25.6 GB/s from ring topology) at 400MHz frequency and supports multi-casting to the 16 PEs. The total power consumption of overall processor is 496mW at 1.2V when the object recognition task is running at 60fps, where the MC-NoC dissipates 48mW. The table of Fig. 10 (b) summarizes the chip and MC-NoC features. Fig. 11 shows the measured waveforms of the multi-casting operation in the MC-NoC. It can be shown that the multi-casting packet waits until all grant signals are gathered from output arbiters and is routed after all output paths are reserved.

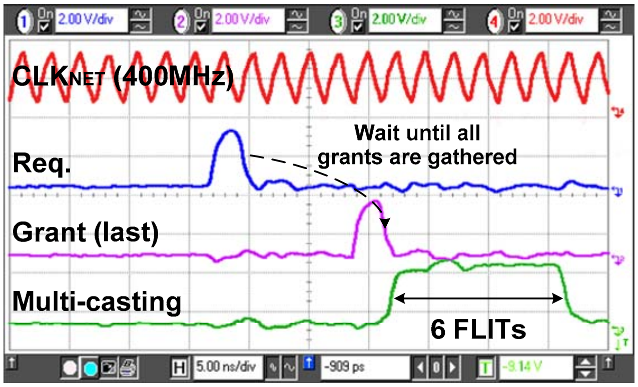


Fig. 11 Measured waveforms

## V. Conclusion

A 118.4GB/s multi-casting network-on-chip (MC-NoC) is developed as communication platform for a real-time object recognition processor. The proposed MC-NoC architecture supports less than 3 switch hop latency and multi-casting capability. It reduces data transaction time and energy consumption by 20% and 23%, respectively, under the object recognition traffic. The MC-NoC is fabricated in a 0.13µm CMOS process and consumes 48mW at 400MHz, 1.2V.

## References

[1] S.-J. Lee, et. al. “An 800MHz Star-Connected On-Chip Network for Application to Systems on a chip”, IEEE ISSCC Dig. of Tech. Papers, pp. 468-489, Feb 2003.

[2] D. Kim, et al., “An 81.6 GOPS object recognition processor based on NoC and Visual Image Processing Memory,” IEEE CICC, pp. 443-446, Sep 2007.

[3] R. R. Rojas, et al., "Object recognition SoC using the support vector machines," Journal on Applied Signal Processing, pp.993-1004, 2005.

[4] H.-C. Lai, et al., "Communication-aware face detection using NoC architecture,"International Conference on Computer Vision Systems (ICVS), pp.181-189, 2008.

[5] J.-Y. Kim et al., “A 201.4 GOPS 496mW real-time multi-object recognition processor with bio-inspired neural perception engine,” IEEE ISSCC Dig. of Tech. Papers, pp.150–151, Feb 2009.

[6] S. Vangal et al., “An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS,” IEEE ISSCC Dig. of Tech. Papers, pp.98–99, Feb 2007.

[7] K. Kim, et al., “A 125GOPS 583mW Network-on-Chip based parallel processor with bio-inspired visual attention engine,” IEEE ISSCC Dig. of Tech. Papers, pp.308–309, Feb 2008.

[8] Jakov N. Seizovic, “Pipeline synchronization,” Proc. of IEEE ASYNC, pp. 87–96, Nov 1994.