Deadlock-free XY-YX router for on-chip interconnection network

Yeong Seob Jeong and Seung Eun Lee

Dept. of Electronic Engineering, Seoul National Univ. of Science and Technology
232 Gongreung-ro, Nowon-gu, Seoul, 139-743, Korea
a) seung.lee@seoultech.ac.kr

Abstract: Specifically, based on the observation that a response is always preceded by a request in multi-processor SoCs, this letter proposes a novel deadlock-free XY-YX router for on-chip network performance improvement. In order to avoid deadlock, we add additional physical channels in the horizontal direction and optimize the priority of output channel allocation. Simulation results show the enhancement in the throughput of an NoC.

References

[1] Xiaohui Li, Yang Cao, Liwei Wang, and Tian Cai: Fault-Tolerant Routing Algorithm for Network-on-Chip Based on Dynamic XY Routing, Wuhan University Journal of Natural Sciences, August 2009, Vol. 14, Issue 4, pp 343-348

[2] Ghosal, P. and Das, T.S.: Network-on-chip routing using Structural Diametrical 2D mesh architecture, Emerging Applications of Information Technology (EAIT), Kolkata, November 2012, pp 471-474

[3] Seung Eun Lee, Jun Ho Bahn, and Nader Bagherzadeh: Design of a Feasible On-Chip Interconnection Network for a Chip Multiprocessor (CMP), International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD), pp.211-218, 24-27 Oct. 2007.

[4] Zhen Fang, Erik G. Hallnor, Bin Li, Michael Leddige, Donglai Dai, Seung Eun Lee, Srihari Makineni, and Ravi Iyer: Boomerang: Reducing Power Consumption of Response Packets in NoCs with Minimal Performance Impact, IEEE Computer Architecture Letters, July 2010, Vol. 9, No. 2.

[5] D. Seo, A. Ali, W. Lim, N. Rafique, and M. Thottehodi: Near-optimal worst-case throughput routing for two dimensional mesh networks, International Symposium on Computer Architecture (ISCA), pp.432-443, 2005.

[6] Roman Gindin, Israel Cidon, and Idit Keidar: NoC-Based FPGA: Architecture and Routing, In First International Symposium on Networks-on-Chips (NOCS), 2007, pp 253-264.

1 Introduction

Network-on-chips (NoCs) have been studied as an alternative for bus architecture to communicate among large-scale IPs efficiently. While designing a NoC structure, a wide variety of topologies should be considered that can comprise a network and a routing algorithm to select a path, from a source to a destination, for effective communication. Traditional XY routing, which is often used in torus and mesh topology, has the advantage of deadlock- and livelock-free characteristics and it is easy to design. However, there is a drawback that the traffic is not distributed evenly during communication, resulting in congestion. Therefore, modified XY routing techniques that switch between deterministic or adaptive mode depending on the circumstances of the congested network have been proposed [1-3]. Research combining XY routing and YX routing has recently been performed [4-6]. The objective of these studies is to determine the ratio of the two optimal routing techniques considering the traffic circumstances in order to obtain better performance when implementing the combined XY-YX routing algorithm. This letter proposes a novel deadlock-free XY-YX router architecture combining the XY and YX routing in accordance with the packet information and analyzes the performance of our proposal according to the XY and YX packet ratios.

2 Conventional XY-YX routing architectures

XY-YX routing can be used in various applications. As an example, it has the benefits of performance improvement and power reduction by predicting response packet paths in advance and eliminating wake-up delay [4]. However, designers of XY-YX combined routing networks should consider the fact that its performance varies depending on the degree of the traffic congestion. In other words, efforts to adjust the transmission ratios of the XY and YX packets are required in order to combine the two routing techniques. Moreover, deadlock, which occurs when combining the two routing techniques, also has to be prevented in the network. The algorithms in [5, 6] try to control the transmission ratios of XY and YX packets to maintain optimal load balancing. This technique requires multiple virtual channels per physical channel to resolve the deadlock. Our proposal avoids deadlock by adding two disjoint horizontal channels instead of using virtual channels. Though our approach for deadlock freedom requires additional resources to build two physical channels, it can reduce the complexity in a routing algorithm, which should control multiplexing of virtual channels to escape a deadlock situation if virtual channels are utilized for this purpose. Additionally, the overhead to add physical channels can counterbalance the cost to allocate virtual channel buffers and associated control logic. Therefore, our approach of providing physical channels in the horizontal direction can be beneficial in its own way. The use of horizontal channels is constrained by the direction of delivered data. That is, each horizontal channel is exclusively used depending on XY or YX routing of the delivered packet.
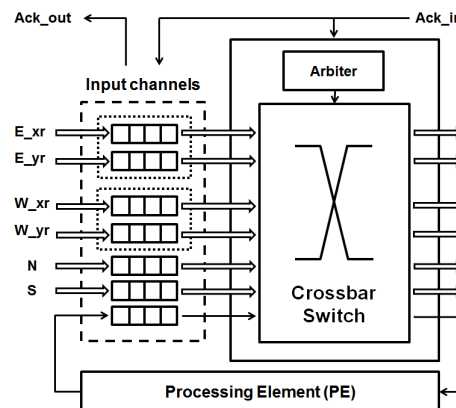
**Fig. 1.** Deadlock-free XY-YX router architecture

3 Deadlock free XY-YX router design

Figure 1 illustrates a single router with seven input and six output channel ports employed in every node of the network. There are two east and two west channel pairs reaching each router, providing two disjoint sub-networks for deadlock freedom. The proposed deadlock-free XY-YX router consists of an arbiter and a crossbar switch and supports wormhole flow control for packet transmission in a mesh topology. There is an input FIFO queue per each input channel. All incoming packets into the router have a single-bit signal to determine whether XY or YX routing is applied. The arbiter allocates the outgoing channel for incoming packets and sends signals to the crossbar switch. The proposed router is easy to implement based on an existing XY router and has higher performance with less overhead. As shown in [4], the XY-YX router can also be used as a low-power architecture so that it has a power-efficient feature at the system level.

For the outgoing channel allocation, the router applies a fixed priority scheme to reduce the complexity of the router. Initially, the possible incoming channels have a descending order of priority in a clock-wise direction for each outgoing channel (referred to as origin hereafter). However, we have observed that vertical channels suffer from high congestion due to the additional channels in the horizontal direction. Thus, the output selection policy is modified to have higher priority in vertical channels than in horizontal channels (referred to as modified hereafter), forwarding packets coming from the north and south, firstly. In addition, in order to prevent a situation in which certain packets remain in the input queue for a long time, we adopt an aging counter, which increases the priority of the packet over time.

4 Experimental results

In order to evaluate the performance of our XY-YX routing algorithm, we have developed a router written in Verilog[TM] HDL. For the measurement of throughput and adjusting incoming traffic, we adopted a standard interconnection network measurement setup where the packet generation is placed in front of an infinite depth source queue, and the input timing of each packet is measured whenever it is
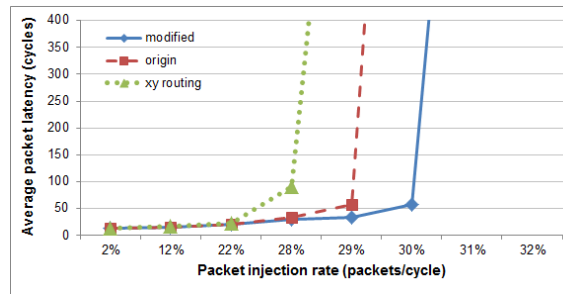
IEICE Electronics Express, Vol.* No.*,*-*
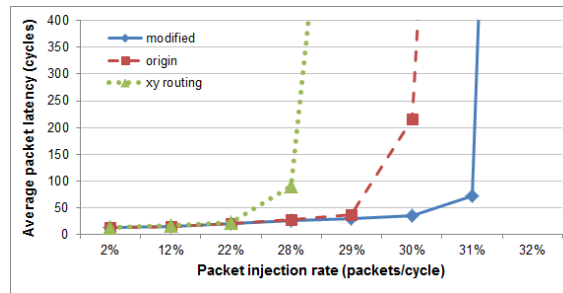
a. xy-50% and yx-50%
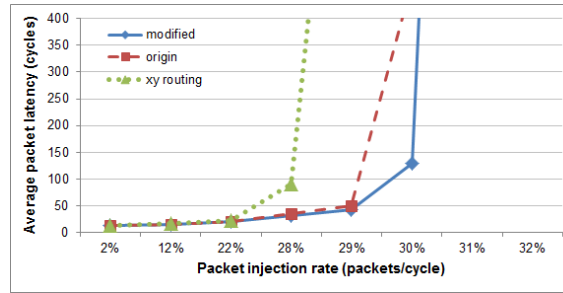
b. xy-10% and yx-90%
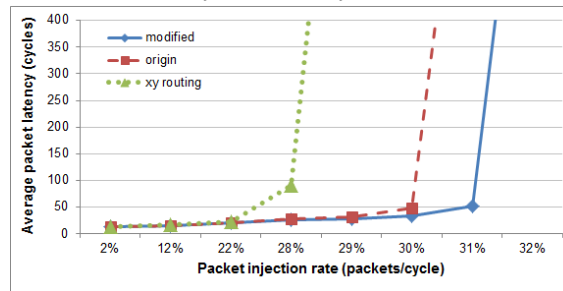
c. xy-30% and yx-70%

d. xy-70% and yx-30%

Fig. 2. The 8×8 mesh network comparison based on XY-YX ratios: xy-50% vs. yx-50% (Fig. 2a), xy-10% vs. yx-90% (Fig. 2b), xy-30% vs. yx-70% (Fig. 2c), and xy-70% vs. yx-30% (Fig. 2d).

generated. In this simulation, the packet length is fixed to eight flits (one head flit and seven body flits) even though the packet format supports packets of various sizes. Each graph represents offered traffic (flit/node/cycle) on the X-axis and average latency (cycles) on the Y-axis.

Figure 2 illustrates the performance of the origin priority of XY-YX routing (origin), the modified priority of XY-YX routing (modified), and the existing XY routing according to XY-YX data ratios in 8×8 mesh topology. The origin router

4

exhibits better performance than existing XY routing and the modified router outperforms in terms of throughput for all traffic patterns, demonstrating the feasibility and superiority of our proposal.

The work in [5] tried to balance the XY and YX traffic in order to improve the performance of the XY-YX router. Splitting the traffic evenly between XY and YX routes in toggle XY (TXY) routing does not always achieve optimal load balancing. Moreover, TXY leads to out-of-order arrivals, requiring large re-order buffers. In order to ensure the in-order arrivals, [6] proposed the weighted ordered toggle (WOT) algorithm, which assigns XY and YX routes to source-destination pairs in a way that reduces the maximum network capacity for a given traffic pattern. In the routing algorithm in [6], the routes are calculated by using the information of the actual traffic pattern. Algorithms in [5, 6] try to balance the traffic of XY and YX packets to maintain optimal load balancing in XY-YX routing in order to improve the performance of the router. In our experiment, we extracted the throughput of the router varying the ratios of XY and YX packets in the network. As shown in Figure 2, our modified routing algorithm outperforms the origin XY-YX routing algorithm over all transmission ratios of XY and YX packets, demonstrating better performance than in [5, 6]. Moreover, our router supports in-order arrivals without virtual channels, reducing the hardware cost to avoid deadlock. In summary, the proposed router architecture enhances the throughput of the existing XY-YX routing algorithms, reducing the additional hardware cost in avoiding deadlock.

6 Conclusions

In this letter, an XY-YX router architecture for flexible on-chip interconnection is proposed and implemented. The main idea is the use of XY and YX routing for different packets and avoiding deadlock by adding two disjointed horizontal channels instead of using virtual channels, thereby reducing hardware cost. Furthermore, the priority of outgoing channel allocation was optimized in order to increase the throughput of the NoC, thereby increasing the accepted load. Experimental results demonstrated the feasibility and superiority of our proposal.

Acknowledgments