Optimized Port Allocation Algorithm for Deflection Router with Minimal Buffering

Igor Stojanovic[1], Milica Jovanovic[2], Sandra Djosic[3] and Goran Djordjevic[4]

Abstract – **In this work we present solutions for improving performance of deflection router with minimal buffering. Proposed solutions modify the both, port allocation algorithm and architecture of the baseline deflection router. In the modified architecture, the buffer inject stage is placed at the output, instead of input of the port allocation stage. The optimized algorithm uses prioritized instead of randomized selection while choosing deflected flit for in-router buffering. Evaluations show that the proposed modifications yield an improvement of 11% in network saturation throughput under uniform traffic pattern.**

Keywords – **Network-On-Chip (NoC), Deflection routing, Side buffering, Multi-core.**

I. Introduction

With the current VLSI technology that supports extensive integration of transistors, the modern System on Chip (SoC) architecture complexity is rapidly growing. Traditional bus-based interconnection fabric faces with many limitations, mostly in term of scalability and bandwidth, which make it inefficient and difficult to meet future hardware demands [1]. A *network-on-chip* (NoC) has been proposed as an alternative option for on-chip communication system that successfully copes with growing complexity by reducing the both, hardware design effort and time to market [2], [3].

NoC is a communication system that includes a set of routers interconnected in a structured way using point-to-point physical channels (links) [3]. Each router also has additional link to a local processing element (PE), implemented as an IP core, that exchanges data with other IP cores in a NoC. Among many NoC topologies, the 2D mesh NoC has emerged due to its simplicity, regularity, and scalability. The NoC routers traditionally employ wormhole switching technique as the robust technique that successfully handles high traffic loads. In order to avoid deadlocks and achieve better channel utilization, it usually adopts virtual channel flow control mechanism [2]. Beside its superior performance, wormhole switching technique is not cost effective for many SoC designs because it relays on buffers for temporarily storing flits (elementary units of exchange data) on the paths to its final destination. Buffers take the large silicon area, which reflects to higher end price, and consume significant amount of power.

The growing demand for low-cost, low-power SoC designs made a new branch in the NoC research area – Bufferless Network-on-Chip [6]. In a difference to buffered NoC, bufferless NoC does not incorporate buffers for temporarily flit storage. Because of a lack of buffers, each flit that reached a router in a current network cycle has to leave it in the next cycle. Each flit of a packet is routed independently through the network and any port contention between multiple arriving flits over the same productive port (i.e. port that brings flit closer to its destination) results in misrouting, where one of the flits is directed to desired port (i.e. productive port) and the rest are deflected, i.e. directed to another output ports. By removing the buffers, bufferless NoC significantly reduces power consumption and silicon costs, because of the smaller chip die area. On the other hand, under high traffic load, performance of the bufferless NoC rapidly drops [4]. This performance degradation is caused by the fact that under high traffic load deflections occurs more frequently. Each deflection puts the flit on the nonproductive way that brings it further from the destination. Therefore, the benefits of the bufferless NoC are only applicable under low to medium traffic load.

Many previous works proposes different mechanisms that try to improve bufferless NoC performance under high traffic load. AFC [7] proposes hybrid NoC that can switch between buffered and bufferless NoC, respective to the traffic state, thus trying to combine the benefits of the both, buffered and bufferless NoC. However, switching between the two modes of operation requires substantial time in order to power up the buffers. Therefore, high frequency of switching under intensive load degrades network performance, while rising power consumption. There is also a line of research that cope with high traffic load by incorporating a small, flit-sized buffer (i.e. side buffer) for storing deflected flits in a bufferless NoC router. The idea is to prevent some of deflected flits to make unproductive hop by storing the flit in the side buffer of the current router and give it the chance in the next cycle to allocate productive port. The most prominent bufferless NoC router with a side buffer that is used as a baseline router in many succeeding works is MinBD [9].

In this work, we present solutions for improving performance of the deflection routers with minimal buffering. The proposed solutions modify the both, router architecture and port allocation algorithm. Modified router architecture balances overall traffic load, at the same time improving injection fairness. In the other hand, proposed port allocation algorithm improves allocation strategy of conventional

[1]Igor Stojanovic is teaching associate at Faculty of Electronic Engineering, University of Nis, A. Medvedeva 14, 18000 Nis, Serbia, E-mail: igor.stojanovic@elfak.ni.ac.rs

[2] Milica Jovanovic is teaching assistent at Faculty of Electronic Engineering, University of Nis, A. Medvedeva 14, 18000 Nis, Serbia, E-mail: milica.jovanovic@elfak.ni.ac.rs

[3] Sandra Djosic is teaching assistent at Faculty of Electronic Engineering, University of Nis, A. Medvedeva 14, 18000 Nis, Serbia, E-mail: sandra.djosic@elfak.ni.ac.rs

[4] Goran Djordjevc is professor at Faculty of Electronic Engineering, University of Nis, A. Medvedeva 14, 18000 Nis, Serbia, E-mail: goran.lj.djordjevic@elfak.ni.ac.rs

deflection routers by prioritizing selection and storing deflected flits that are most likely to be directed to the productive port in the next cycle.

The rest of this paper is organized as follows: Section II gives overview of the bufferless deflection routing. Section III introduces side buffer technique as extension to bufferless deflection routing. The proposed solution is elaborated in the section IV. Section V describes simulation set up, defines and compares measured metrics for the two conventional deflection router architectures and the router with proposed solutions.

II. Bufferless Deflection Routing

Deflection routing technique, used in bufferless NoC, requires that all the flits reached a given router in one cycle, must leave it the next cycle. This restriction is due to the fact that bufferless NoC does not include buffer for flit storing, so there is no room for new incoming flits unless the previous flits leave it. In a situation when two or more incoming flits have one productive port (port that brings it closer to its destination), only one flit can be directed to productive port, and the remaining are deflected to non-productive, free ports. The deflection routing can be implemented to any network topology as long as there are routers with the same number of input and output ports. In the 2D mesh topology, which is used in this work, each router has five bidirectional links, where four of them are used for exchanging flits with neighboring routers and the fifth is used by the local IP core for flit inject/eject.

Fig. 1 shows a set of routers connected in the 2D mesh topology.

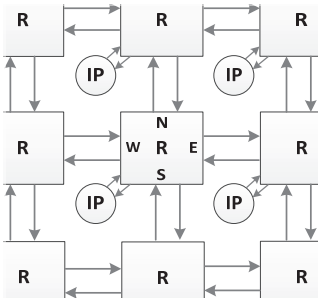


Fig. 1. 2D Mesh NoC topology

In general, internal bufferless NoC architecture consists of three main stages: Eject, Inject, and Port Allocation and Switching (PAS) stage (Fig. 2.).

The Eject stage takes one of the locally addressed flits (flit addressed to IP core attached to a current router), if present, and brings it to the local IP core. The Inject stage takes the flit from the local IP core and injects it to the network, if there are free input ports. The PAS stage firstly determines productive ports for each incoming flit by comparing the addresses of the current router and the router of the destination IP core.

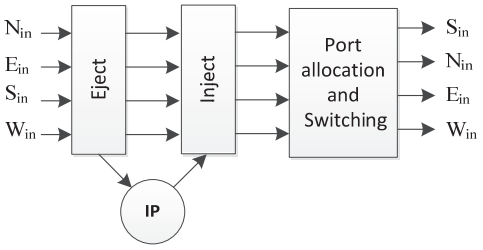


Fig. 2. Architecture of baseline deflection router

Then, the PAS stage moves the input flits to the output ports according to the information obtained in previous activity. Note that PAS stage has to resolve situation when multiple input flits content over the same productive output port. The implemented contention resolution strategy significantly influences overall NoC performance. In other words, PAS stage makes the important difference between various bufferless NoC designs.

Among several bufferless NoC router architectures, BLESS [6] and CHIPPER [8] has emerged as baseline routers for many future works. The only difference between BLESS and CHIPPER is in PAS stage. BLESS incorporates 4x4 crossbar switch controlled by the allocator unit. Port allocation is priority based where the highest priority is assigned to the oldest flit in the network. This assures that each flit reach it destination in a finite period of time. Although priority based allocation strategy achieves superior performances, it relays on complex, sequential hardware with high latency. On the other hand, CHIPPER incorporates much simpler PAS stage that is composed of two-stage permutation network with four arbiter blocks. Each arbiter block implements allocator unit and 2x2 crossbar switch. Crossbar switch is configured by the randomly selected flit's productive ports. This simplification significantly reduces hardware footprint and latency, but with the performances penalty.

III. Deflection Routing with Side Buffer

Among many prior works that try to compensate bufferless deflection router performance loss, deflection routing with side buffer has emerged as promising solution. Introduced in MinBD [9], this solution attracts wide attention due to its simple, cost-effective extension to the traditional bufferless routers, such as BLESS and CHIPPER. In difference to baseline bufferless routers, it implements small side buffer attached to each router in the network. The idea is to preserve up to one deflected flit in a cycle from being misrouted by storing it temporarily in a side buffer. The flit stays in the current router until the next network cycle when it is given a chance to contend again for a productive output port. By preventing some of deflected flits to be misrouted, side buffer implementation reduces misrouting rate, at the same time improving network performances.

Fig. 3 shows architecture of baseline deflection router with side buffer.

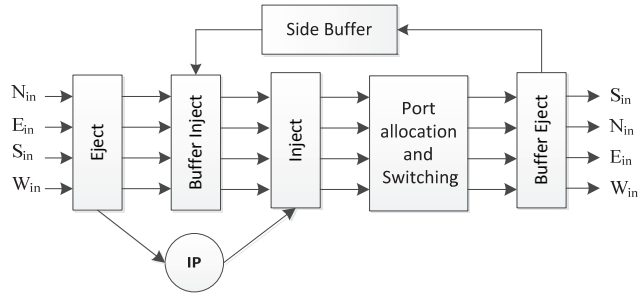


Fig. 3. Architecture of baseline deflection router with side buffer

A side buffer is attached to the deflection router via two additional stages – buffer eject and buffer inject stage. Buffer eject stage randomly takes one of deflected flits at the output port and puts it to the side buffer if it is empty. This flit will be injected to the network in some future network cycle via buffer inject stage. Although it significantly improves network performance, this router architecture reduces injection fairness [4][10]. Preceding the inject stage of a local IP core, buffer inject stage gets higher priority over inject stage thus lowering chance of a local IP core to inject the flits in the network.

IV. Optimized Bufferless Deflection Router Architecture

In this work we propose solutions that modify the both, router architecture and port allocation algorithm of the traditional bufferless NoC router with side buffer. Fig. 4 shows proposed router architecture.

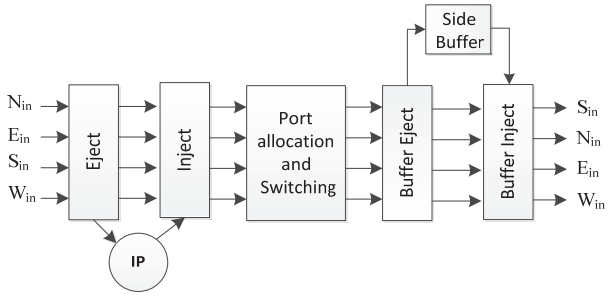


Fig. 4. Proposed deflection router architecture

As it can be observed, in the proposed router architecture the buffer inject stage is moved from the input to the output of the PAS stage. This organization raises probability for a local IP core to inject flits in the network, thus balancing overall traffic and improving injection fairness.

Traditional bufferless NoC router with side buffer implements simple allocation algorithm that randomly selects one of the deflected flit for storing in a side buffer. The proposed solution modifies this algorithm by enabling additional options when choosing one of deflected flits for buffering. The modified algorithm includes several activities. At the first, it identifies candidate flits for storing in the side buffer. A candidate flit is any non-locally addressed deflected flit, i.e. any flit that is directed to a non-productive port at the output of PAS stage except those that are addressed to the local IP, but not ejected from the router in the current network cycle. At the second step, the algorithm selects one of candidate flits (if any) for storing in the side buffer. The highest priority is given to the candidate flit deflected to the output port that is productive for the flit which currently resides in the side buffer. The rationale for such selection criterion is to prevent misrouting of the buffered flit due to the lack of free productive ports. Namely, injection of a deflected flit into the side buffer forces the buffered flit to leave the router. If all its productive ports are occupied, the buffered flit will inevitably be misrouted to an unwanted direction. However, if the selected candidate flit and the buffered flit share the same productive port, the buffered flit can always be productively routed through the port which is freed up after the candidate flit is injected into the side buffer. If more than one candidate flits share productive port with the buffered flit, the priority is given to the candidate flits with two productive ports. Such selection criterion is motivated by the fact that the probability to allocate a productive port for the buffered flit is higher if the flit has two productive ports.

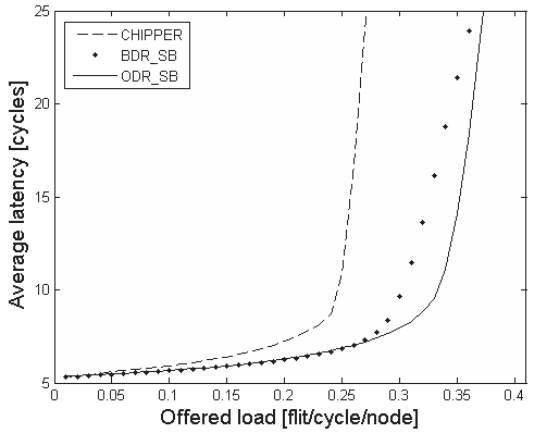
Finally, if none of candidate flits share productive port with the buffered flit, or the side buffer is empty, the algorithm gives priority to candidate flits with two productive ports, also. If there are no such flits, the algorithm reduces to the allocation policy of the baseline bufferless deflection routers with side buffer – i.e., it randomly selects a candidate flit for injecting into the side buffer.

Coupled with the optimized router architecture with side buffer, the proposed algorithm improves buffer utilization by favoring storage of the deflected flits that have the highest probability to get productive output port in the following network cycles. Together with the modified router architecture, proposed solutions also improve overall injection fairness.

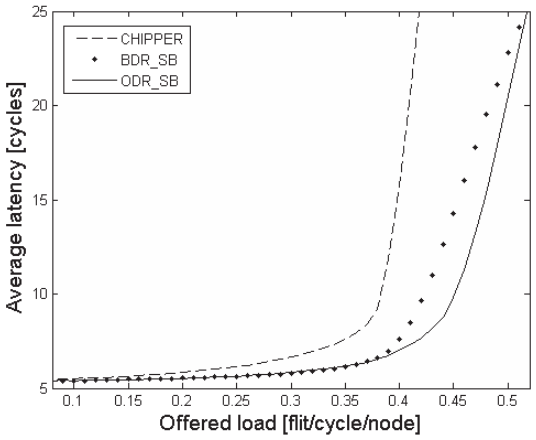
V. Evaluation

In order to evaluate performance of the proposed solutions we use an in-house cycle-accurate NoC simulator developed in SystemC. We have simulated two conventional router architectures: CHIPPER (as a representative of true bufferless deflection router), MinBD (as the baseline deflection router with a side buffer), and the router with a side buffer that implements our architecture and port allocation algorithm. Routers are organized within the 2D mesh network with dimension of 8x8 routers. The simulation is conducted under synthetic traffic with flits addressed in the both uniform and transpose traffic patterns. Flits are generated and injected into the network following a Poisson distribution. In all our simulations, the flit injection rate (i.e. the average inter-arrival time of flits at injection port of each router) is swept from zero to network saturation.

Fig. 5 shows simulation results. Latency numbers presented in these graphs are measured from the time the flit was generated at the source node to the time it arrives at the destination node, including the time the flit spends in the IP core's queue. As can be observed from these figures, for both traffic patterns, all the router architecture achieve almost the same performance at low traffic load. As the traffic load increases, the flit latency dramatically increases due to the network congestion.



(*a*)



(*b*)

Fig. 5. Load-latency graphs comparing deflection router architectures (CHIPPER, Baseline deflection router with side buffer – BDR_SB, Optimized deflection router with side buffer – ODR_SB): (*a*) Uniform traffic pattern; (*b*) Transpose traffic pattern

Table 1 compares throughputs (i.e. maximum traffic accepted by the network measured in flits per node and per clock cycle) of the network based on conventional deflection router architectures with the throughput of the network based on deflection router with side buffer that implements proposed solutions. For each network, throughput is measured under saturation traffic load. As it can be observed, the proposed solutions improve network throughput for 11%, under uniform traffic load, and 12%, under transpose traffic load, compared to baseline bufferless deflection router with side buffer.

TABLE I
NOC THROUGHPUT COMPARISON

Traffic pattern	CHIPPER	Baseline deflection router with SB	Optimized deflection router with SB
Uniform	0.242	0.332	0.366
Transpose	0.375	0.40	0.45

VI. CONCLUSION

In this paper we have proposed a modified architecture of minimally-buffered deflection router in which the position of the buffer inject stage is moved from input to the output of the PAS stage. Also, we have proposed a port allocation algorithm for deflection router with side buffer. The proposed algorithm modifies conventional random based buffering of deflected flits by prioritizing the buffering of the deflected flits that are most likely to get productive port in the next network cycle. In conjunction, these two modifications improve overall performances of deflection-routing based network on chip, in terms of misrouting rate, latency and injection fairness. By replacing conventional deflection router with the proposed optimized router, the saturation throughput of 2D mesh network is increased for 11%.

REFERENCES

[1] S. Borkar, "Future of Interconnect Fabric: A Contrarian View", *12th ACM/IEEE International Workshop on System Level Interconnect Prediction*, pp. 1-2, 2010.

[2] W. Dally, W. James, and B. Towles, *Principles and Practices of Interconnection Networks*, Elsevier, 2004.

[3] F. Gebali, H. Elmiligi, and MW. El-Kharashi, *Networks-on-chips: Theory and Practice*, Taylor & Francis Group, LLC, 2006.

[4] I. Stojanovic, M. Jovanovic, and G. Djordjevic, "Dual-mode Inter-router Communication Channel for Deflection-routed Networks-on-chip", *Journal of Supercomputing*, pp. 2597-2613, 2015.

[5] G. Michelogiannakis, D. Sanchez, W.J. Dally, and C. Kozyrakis, "Evaluating Bufferless Flow Control for On-chip Networks", *Proc. 4th ACM/IEEE International Symposium on Networks-On-Chip*, pp. 9-16, 2010.

[6] T. Moscibroda, and O. Mutlu, "A Case for Bufferless Routing in On-Chip Networks", *Proc. 36th International Symposium on Computer Architecture*, pp. 196-207, 2009.

[7] S. Jafri, Y. Hong, M. Thottethodi, and T. Vijaykumar, "Adaptive Flow Control for Robust Performance and Energy", *Proc. 43rd Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 433-444, 2010.

[8] C. Fallin, C. Craik, O. Mutlu, "CHIPPER: A Low-complexity Bufferless Deflection Router", *Proc. 17th International Symposium on High Performance Computer Architecture (HPCA)*, pp. 144-155, 2011.

[9] C. Fallin, G. Nazario, X. Yu, K. Chang, R. Ausavarungnirun, and O. Mutlu, "MinBD: Minimally-Buffered Deflection Routing for Energy-Efficient Interconnect", *Proc. 6th IEEE/ACM International Symposium on Networks on Chip*, pp. 1-10, 2012.

[10] I. Stojanovic, and G. Djordjevic, "In-channel Misrouting Suppression Technique for Deflection-Routed Networks on Chip", *FACTA UNIVERSITATIS, Electronics and Energetics,* pp. 309-323, 2016.