

# An Empirical Evaluation of Topologies for Large Scale NoC

Mehdi Baboli\*<sup>1</sup>, Nasir Shaikh-Husin<sup>2</sup>

Department of Electronics & Computer Engineering, Faculty of Electrical Engineering,  
Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia

\*Corresponding author, e-mail: mehdi.baboli@fkegraduate.utm.my<sup>1</sup>, nasirsh@utm.my<sup>2</sup>

## Abstract

*In the past decades, processing power has achieved considerable gains. Researchers proposed faster uniprocessors that are capable of improving the instruction level parallelism through out-of-order implementation to increase the performance quality of the existing network-on-chip (NoC). Diminishing returns of the performance of uniprocessor architecture caused multiprocessors to be integrated on a chip. In this paper, we selected a popular NoC topology, i.e., mesh, and evaluated it in terms of latency, maximum delay, average throughput, and total energy under different routing algorithms, number of router buffers, and random traffic model. We selected two sizes of NoC, 12×12 and 16×16, to represent large scale NoC. We investigated all characteristics and measured latency, maximum delay, and total energy by Noxim simulator. In this paper, we demonstrate that when the network size is large and number of buffers is insufficient, popular routing algorithms cannot ensure good network performance and almost all routing algorithms have the same performance for the large scale NoCs.*

**Keywords:** network-on-chip, network topology, network routing algorithm

**Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.**

## 1. Introduction

One of the most important trends in computer architecture in the past decade has been the move towards the use of multiple CPU cores on a single chip. The system integration has developed such that a complete system can be placed on one chip. As more processors are on a chip, improved bandwidth will be available, and its efficiency for sharing the on-chip communication architecture would be also improved. Hardware developers considered communication on a chip as the main bottleneck in multi-core era [1]. The system-on-chip (SoC) design is expanded to accommodate an increasing number of resources in order to satisfy the high performance requirements. Conventional bus-based systems cannot provide the efficiency and scalability in interconnecting many cores on one chip. NoC has the capacity for meeting the challenges. NoC is an on-chip communication infrastructure consisting of interconnected routers within a regular topology (e.g., a mesh), enabling integration of the memories, computational processors, and the Intellectual Property (IP) components. The method of communication for the resources within an NoC-based system is by utilizing packets through routers within a network.

NoC has some advantages over the conventional bus; for instance, NoC makes a distinction between the computation and communication and, consequently, it can increase the simplicity of design of the communication system. Systems that are based on NoC have a modular approach with clear distinction among the components. NoC has some desired features, including flexibility, reusability, and capability for quick prototyping during the construction of an SoC.

### 1.1. Network Topology

Network topology determines the way the IP cores are laid out physically and how they are interconnected to each other via the links existing within the network. Many different topologies have been proposed [4], such as mesh, torus, binary tree, octagon, mixed, and custom topology. The general purpose network topologies such as rings [2] and meshes [3] are popular selections for the on-chip networks due to ease of physical layout, the router complexity and wire length. The most common topology is 2D mesh due to its grid-type shape and regular

structure which is the most appropriate for the two dimensional layout on a chip. It can be easily expanded by adding new nodes and links without any modification to the existing node structure. Another reason behind mesh's popularity is its capability to be partitioned into smaller meshes, which is a desirable feature for parallel applications. According to some comparative studies, a number of spatial topologies were shown to have better performance than mesh and torus [5]. In [6], they introduce a novel network alignment algorithm which is based only on network topology architecture. In [7], a mesh-based interconnect architecture called CLICHE was developed by placing computational resources along with the switches arranged in an m-by-n mesh. SPIN [8] is a generic interconnect template that uses fat-tree architecture to interconnect IP blocks. In SPIN, every node has four children, and the parent is replicated four times at any level of the tree. In [9], the authors proposed a 2D torus as an NoC architecture. The torus architecture is basically the same as a regular mesh except that there are wrap-around channels connecting the edge switches. OCTAGON [10] utilizes a basic octagon unit consisting of eight nodes and bidirectional links. Each node is associated with a processing element (PE) and a switch. Communication between any pair of nodes takes at most two hops within the basic octagonal unit. Butterfly fat-tree topology was adopted as an interconnect template where the IPs are placed at the leaves and switches are placed at the vertices [11]. In [27], They compare two popular NoC topologies, i.e., mesh and torus, in terms of different figures of merit e.g., latency, power consumption, and power/throughput ratio under different routing algorithms and two common traffic models, uniform and hotspot. Some researchers have proposed application-specific topology that can offer superior performance while minimizing area and energy consumption [12].

## 1.2. Routing Algorithm

For a network topology, the paths through which the packets are transmitted between the source node and the destination node are determined by the routing algorithm. With oblivious routing, minimum delay and optimal throughput are not always achievable because the routing paths would be oblivious about the state of network congestion [13]. In the situation in which the optimal throughput is not attainable with minimal routing, then the non-minimal routing paths may be employed to balance latency and throughput. For adaptive routing algorithms, the routing paths can be adapted to the existing traffic conditions through routing around the links that are heavily congested. These adaptive routing algorithms apply more complicated control hardware for sensing and reacting to the network congestion [14, 15]. In [16], the authors proposed a virtual channel routing algorithm to achieve load equality. The algorithm directs packets to different virtual channels in different virtual networks to shun livelock and deadlock. Their results illustrated that the throughput and latency can be improved by selecting the appropriate random factor for different traffic models.

Dimension-Ordered Routing (DOR) [17] is one of deterministic routing algorithms, which routes the packets, first, in one dimension and then along the next dimension. For this reason, it is known also as the XY routing algorithm (first, X direction, then, Y direction). It is one of the popular algorithms because of its simplicity that enables it to be implemented with a low cost. Randomized Oblivious Multi-phase Minimal routing (ROMM) [13] chooses an intermediate node randomly in a rectangle that is defined by the source and destination nodes, and then the packets are routed by intermediate node. A strong point of ROMM is that a better load balance can be achieved and its randomization results in path diversity. Similarly, in VALIANT routing [18], the routing is separated into two steps. In deviation from ROMM, in VALIANT the intermediate node is allowed to be selected among all nodes existing within the networks. In Orthogonal one-TURN routing (O1TURN) [19], the packets are routed by means of one of at most two dimension-ordered, minimal paths by selecting the first traversal dimension randomly. It has been found that the O1TURN routing has a worst-case near-optimal throughput. Turn model [20] has been proposed to develop deadlock-free adaptive routing algorithms by prohibiting some turns at routers to break cycles in resource dependence graph. This approach utilizes three routing algorithms that are partially adaptive, including Negative First, North Last, and West First. Unlike turn model in which, at all routers, some particular turns are prohibited, Odd-Even turn model [21] devises the adaptive routing through confining the positions wherein some types of turns can be taken. The result of this technique is that the degree of adaptation can be more evenly distributed in routing. As a hybrid approach, dynamically switching between Adaptive and Deterministic routing (DyAD) [22] has been proposed. It uses both adaptive and

deterministic routing. In cases that the network is not congested, DyAD route packets in its deterministic mode in order to guarantee a low level of latency. In cases where the network is congested, DyAD routers would be switched automatically to the adaptive mode, which enables it to avoid the congested links.

## 2. Large Scale Network-on-Chip

During the past decade, the application of multiple CPU cores to a single chip has been one of the main trends in computer architecture. There exist real chips with 100 cores, and even with 1000 cores for a research prototype at University of Glasgow [26]. While increased core count has allowed processor chips to scale without experiencing complexity and power dissipation problems inherent in larger individual cores, new challenges have come to exist.

Considering the dramatic increase in the number of nodes inside the NoC and also the increase in transaction between nodes, the rate of transmission of the data inside the links rises. Some links are likely to be used more excessively than other links which can lead to lack of load balancing inside the NoC. This causes some packets inside the large NoCs to have long paths to reach the destination. Long packet path causes an increase in certain parameters such as latency, packet loss, and power consumption, and reduction in throughput. One of the ways to remove the aforementioned problems is to use proper routing algorithm. However, the topologies that currently exist are good for small-sized networks only. Therefore, it is necessary to design and develop a new topology that can be applied to large-sized NoCs. The topology defines how routers are connected with each other and the network endpoints. Furthermore, a routing algorithm has to be optimized to suit the suggested topology. For a large-scale system, the topology has a major impact on the performance and cost of the network. For example, in [24], large NoCs were evaluated for 16 to 4096 cores. They showed that an increase in the number of connected cores caused the latency to be increased and the throughput to be decreased. Problems such as routing algorithm, QoS, power consumption, and topology for small NoCs have been addressed. Some researchers are working on large NoCs to obtain a solution for the problems associated with the large size. Due to this congestion, load balancing and throughput are not solved completely. Additionally, adequate studies have not been conducted to propose new topologies for large NoC in order to improve the issues related to performance and cost.

## 3. Research Method

This paper reports on overall assessment on the impact of routing algorithms and number of buffer on large-scale NoCs. This evaluation was derived from simulations using Network-on-Chip Simulator (Noxim) for 12×12 and 16×16 mesh topology. The Noxim simulator [25] was developed using SystemC, a system description language based on C++. Noxim has a command line interface for defining several parameters of a NoC. This simulation was done for XY, West First, North Last, Negative First, Odd-Even, and fully adaptive routing algorithms, using random traffic. Noxim uses DyAD algorithm for fully adaptive routing. Additionally, it was done for different number of buffers for each input port, which are 2, 4, 8 and 12. This simulation was performed under warmup for 500 cycles. Warmup is an option in Noxim that users can set before the simulator starts to collect statistics. Each simulation was performed for 100000 cycles. From the simulation, global average delay (GAD), maximum delay, global average throughput (GAT), and total energy, for both router and link, were evaluated. Noxim assumes certain energy parameters in order for it to determine energy consumption. For example, average energy expended by a flit to go through a switch is estimated to be 0.151 nJ for XY routing.

## 4. Experimental Results and Analysis

In the first set of experiments, we tested the performance of 12x12 mesh topology. The global average delay for this topology for various routing algorithms and number of buffers is shown in Figure 1. From the graph, it can be seen that all routing algorithms almost have similar performance. When the number of buffer is 2, Odd-Even routing algorithm has significantly higher latency than other routing algorithms. Figure 2 shows the maximum delay metric performance. Again, all routing algorithms have similar performance; the only difference is

shown when the number of buffer is 2. In this case, fully adaptive algorithm is much worse than the other algorithms. For global average throughput, all routing algorithms have similar performance except the Negative First routing algorithm, as shown in Figure 3. Figure 4 plots the total energy, and it is obvious that fully adaptive routing algorithm outperformed the others.

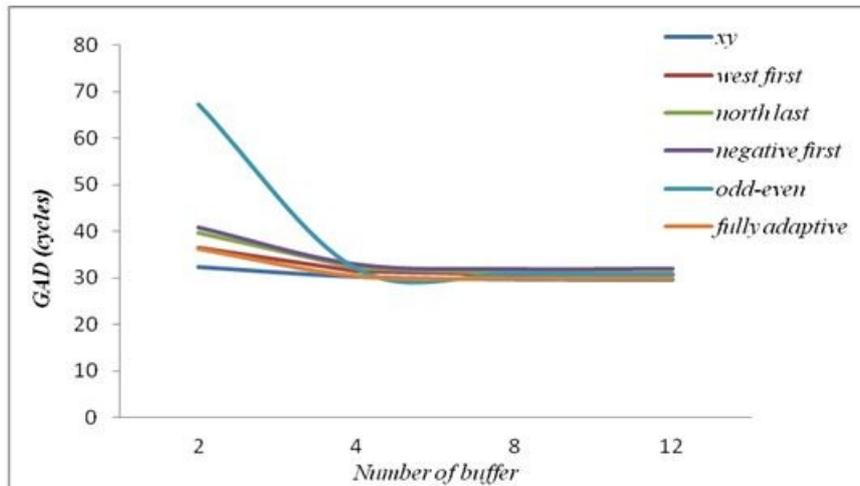


Figure 1: Global average delay (cycles) for 12x12 mesh topology with different number of buffers and routing algorithms

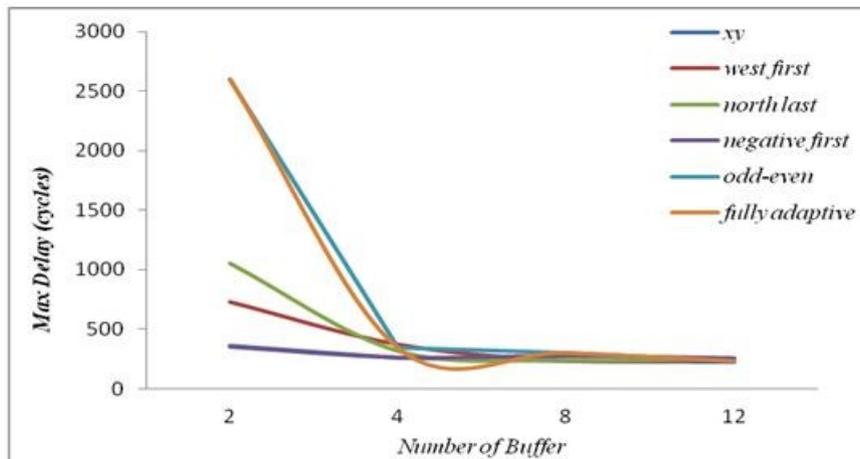


Figure 2: Maximum delay (cycles) for 12x12 mesh topology with different number of buffers and routing algorithms

We have repeated again all simulations for mesh topology in dimension 16x16. As can be seen in Figure 5 and Figure 6, which show global average delay and maximum delay respectively, West First, North Last, and Odd-Even routing algorithms do not have suitable performance when number of buffers is small. They are not suitable routing algorithms for large size NoCs if number of buffers is inadequate, because when they are used, the latency rises steeply. On the other hand, XY, Negative First, and fully adaptive routing algorithms have negligible difference about latency with each other. Figure 7 shows global average throughput. Odd-Even routing algorithm performance degrades rapidly when number of buffer is 12. Figure 8 illustrates total energy for mesh in dimension 16x16. The fully adaptive routing algorithm has better result than other routing algorithms. In fact, all routing algorithms have similar performance except fully adaptive.

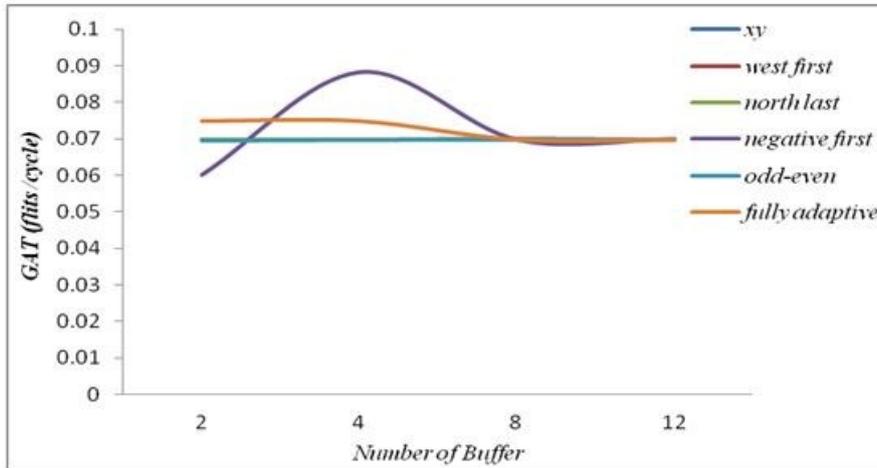


Figure 3: Global average throughput (flits/cycle) for 12x12 mesh topology with different number of buffers

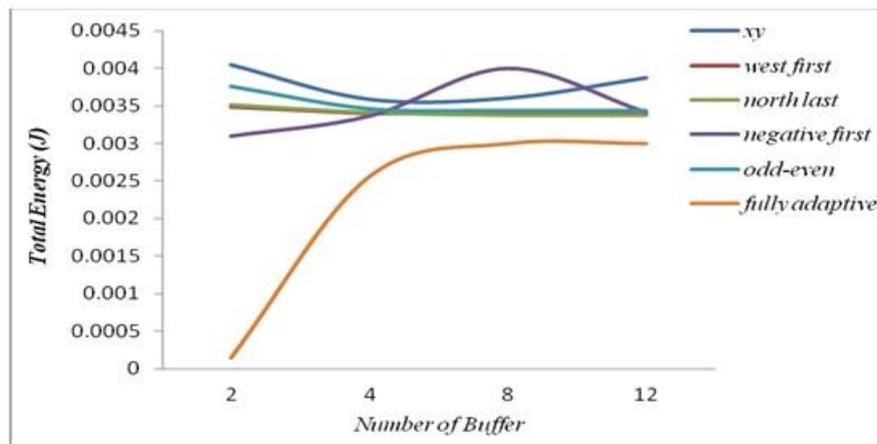


Figure 4: Total Energy (J) for 12x12 mesh topology with different number of buffers

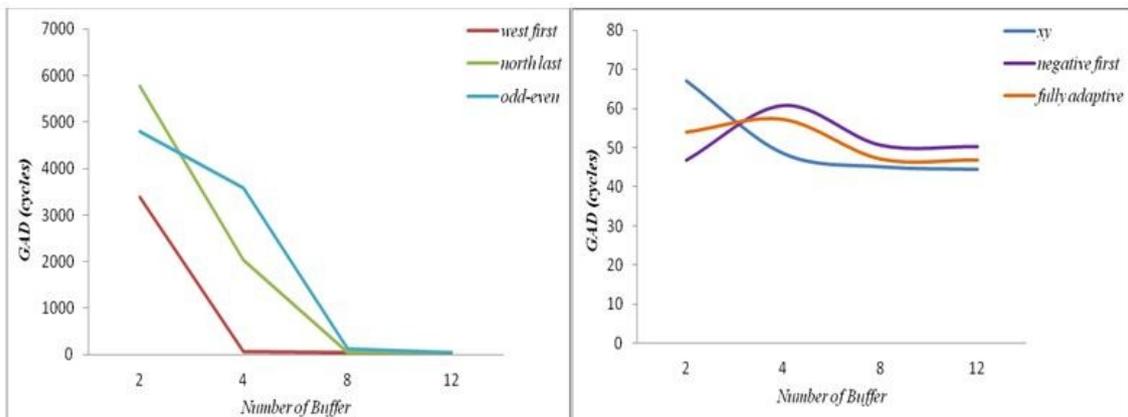


Figure 5: Global Average Delay (cycles) for 16x16 mesh topology with different number of buffers

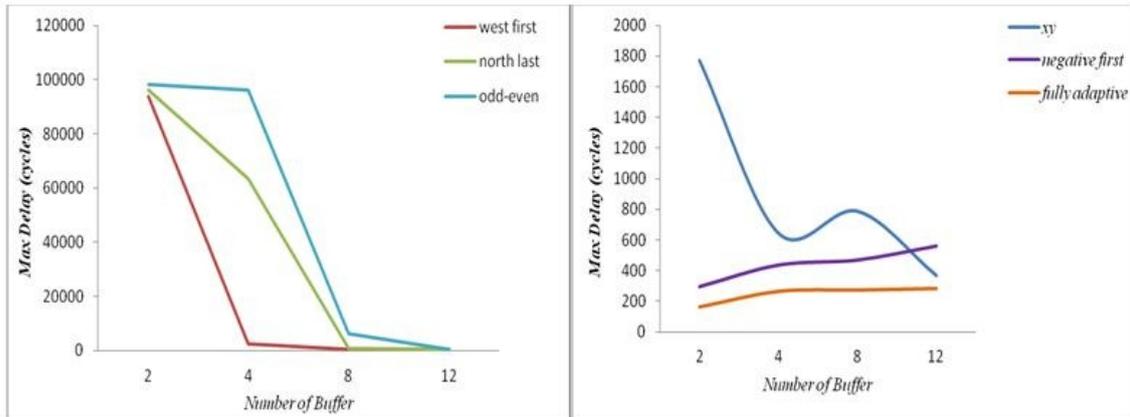


Figure 6: Max Delay (cycles) for 16x16 mesh topology with different number of buffers

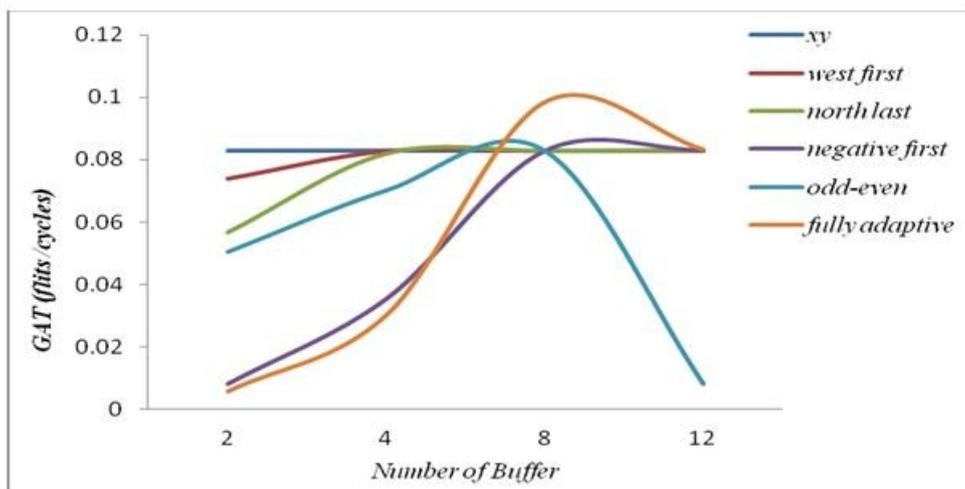


Figure 7: Global Average Throughput for 16x16 mesh topology with different number of buffers

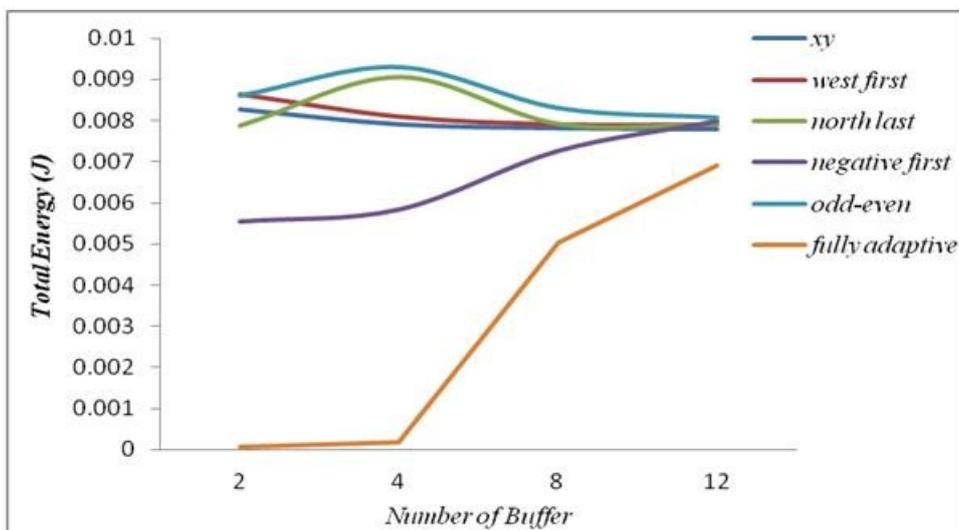


Figure 8: Total Energy for 16x16 mesh topology with different number of buffers

## 5. Conclusion and Future Work

Researchers have solved some problems such as routing algorithm, QoS, power consumption, and topology for small size NoCs and some are currently working on large size NoCs to achieve a solution for problems. Because of this congestion, load balancing and throughput have not been solved completely. Furthermore, sufficient research has not been carried out to propose a new topology for large NoC in order to improve issues such as the performance and cost. We should not however forget that, for a large-scale NoC, the topology has a major impact on the performance and cost of the network. Therefore, in this paper, we proved that routing algorithms could not improve the performance and cost in large size NoCs. In order to achieve good performance in a large network on chip, both the topology and routing algorithms should be provided or there should be a new design. In future, we will focus on designing a novel topology and new routing algorithm for large NoCs in order to improve the performance.

### Acknowledgment:

The authors would like to express their gratitude to Universiti Teknologi Malaysia (UTM) and the Ministry of Higher Education (MOHE), Malaysia for supporting this research work under Research Grant No. R.J130000.7923.4S093.

### References

- [1] Chrysostomos AN, Dongkook P, Jongman K, Vijaykrishnan N, Mazin SY, Das CR. *ViChar: A Dynamic Virtual Channel Regulator for Network-on-Chip Routers*. IEEE/ACM International Symposium on Microarchitecture. Orland. 2006; 39: 333–346.
- [2] Seiler L et al. A Many-Core x86 Architecture for Visual Computing. *ACM Transactions on Graphics*. 2008; 27(3):1–15.
- [3] Howard J et al. *A 48-Core IA-32 Message-Passing Processor with DVFS in 45nm CMOS*. IEEE International Solid-State Circuits Conference Digest of Technical Papers, ISSCC '10. San Francisco. 2010; 10:108 –109.
- [4] Salminen E, Kulmala A, Hamalainen T. *Survey of Network-on-Chip Proposals*. OCP-IP white paper. 2008; 1–13.
- [5] Baboli M, Shaikh Husin N, Marsono MN. *A Comprehensive Evaluation of Direct and Indirect Network-on-Chip Topologies*. Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management. Bali. 2014; 2081-2090.
- [6] Zou Q, Liu F, Hou T, Jiang Y. A Topology-Based Algorithm for Directed Network Alignment. *TELKOMNIKA*. 2013; 11 (10): 6202 – 6208.
- [7] Kumar S, Jantsch A, Soininen JP, Forsell M, Millberg M, Oberg J, Tiensyrja K, Hemani A. *A Network on Chip Architecture and Design Methodology*. IEEE Computer Society VLSI Annual Symposium. Pittsburgh, 2002; 105 – 112.
- [8] Guerrier P, Greiner A. *A Generic Architecture for on-Chip Packet-Switched Interconnections*. Design, Automation and Test in Europe Conference and Exhibition. 2000; 250 -256.
- [9] Dally W, Towles B. *Route Packets, Not Wires: on-Chip Interconnection Networks*. Design Automation Conference. 2001; 684 – 689.
- [10] Karim F, Nguyen A, Dey S. An Interconnect Architecture for Networking Systems on Chips. *IEEE Micro*. 2002; 22(5): 36 – 45.
- [11] Pande P, Grecu C, Ivanov A, Saleh R. *Design of a Switch for Network on Chip Applications*. IEEE Circuits and Systems, ISCAS '03; 2003; 5: 217-220.
- [12] Marculescu R, Umit Y, Peh L-S, Natalie E J, Yatin H. Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*; 2009; 28(1): 3-21.
- [13] Sunkam R, Bill L. *Near-Optimal Oblivious Routing on Three Dimensional Mesh Networks*. IEEE International Conference on Computer Design. 2008; 134 –141.
- [14] Daniel S, George M, Christos K. An Analysis of On-Chip Interconnection Network for Large-Scale Chip Multiprocessors. *ACM Transactions on Architecture and Code Optimization (TACO)*. La Jolla, 2010; 7(1):1-28.
- [15] Sunkam R, Lin B. *Destination-Based Adaptive Routing on 2D Mesh Networks*. Proceedings of the 6th ACM/IEEE Architectures for Networking and Communications Systems (ANCS); New York; 2010; pp 1–12.

- 
- [16] Tao H, Zhang X, and Qin T. Improving Negative First Routing Algorithm with Load Equalization of Virtual Channel in NoC. *TELKOMNIKA*. 2013; 11 (11): 6460 – 6467.
- [17] Sullivan H, Bashkow T R. A Large Scale, Homogeneous, Fully Distributed Parallel Machine. *Proceedings of the 4th Annual Symposium on Computer Architecture, ISCA '77*. New York, 1997; 5(7): 105–117.
- [18] Borkar S. *Thousand Core Chips: a Technology Perspective*. New York, 2007; 746-749.
- [19] Seok E, Cao C, Shim D, Arenas D, Tanner D, Hung CM. *A 410 Hz CMOS Push-Push Oscillator with an on-Chip Patch Antenna*. IEEE Solid-State Circuits Conference (ISSCC); 2008; 472 –629.
- [20] Glass C, Ni L. The Turn Model for Adaptive Routing. *Journal of the ACM (JACM)* 1994; 41(5): 874–902.
- [21] Bell S, Edwards B, Amann J, Conlin R, Joyce K, Leung V, Mackay J, Reif M, Bao L. *TILE64 Processor: A 64-Core SoC with Mesh Interconnect*. IEEE International Solid-State Circuits Conference. Los Alamitos, 2008; pp 88-598.
- [22] Vangal S et al. *An 80-Tile 1.28 TFLOPS Network-on-Chip in 65 nm CMOS*. IEEE International Solid-State Circuits Conference. 2007; pp 98 -589.
- [23] Owens J, Dally W, Keckler W. Research Challenges for on-Chip Interconnection Networks. *IEEE Micro*. 2007; 27(5): 96 - 108.
- [24] Javier N, Behram K, Salman K, Paolo F, Mikel L. *Reservation-Based Network-on-Chip Timing Models for Large-Scale Architectural Simulation*. Sixth IEEE/ACM International Symposium on Networks-on-Chip. 2012;18: 91 – 98.
- [25] Noxim: Network-on-Chip simulator, [Online], Available: <http://sourceforge.net/projects/noxim>.
- [26] Scientists Squeeze More Than 1,000 Cores on to Computer Chip, [Online], Available: <http://goo.gl/KdBbW>.
- [27] Mirza-Aghatabar M, Koochi S, Hessabi S, Pedram M. *An empirical investigation of Mesh and Torus NoC topologies under different routing algorithms and traffic models*. IEEE Proceedings of the 10th Euromicro Conference on Digital System Design Architectures. 2007: 10: 19 – 26.