ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 1, No.7 , January 2014
ISSN : 2322-5157
www.ACSIJ.org

ACSIJ
WWW.ACSIJ.ORG

Contention-aware virtual channel assignment in application specific Networks-on-chip

Mona Soleymani[1] , Fatemeh Safinia[2] , Somayyeh Jafarali Jassbi[3] and Midia Reshadi[4]

[1,2]Department of Computer Engineering Science and Research branch, Islamic Azad University, Tehran, Iran
[1]Mona.soleymani@gmail.com
[2]f.safinia1@gmail.com

[3,4] Faculty Member of Computer Department, Science and Research Branch, Islamic Azad University, Tehran, Iran
[3]s.jassbi@srbiau.ac.ir
[4]reshadi@srbiau.ac.ir

Abstract

Nowadays lots of processing elements should be placed on a single chip because of ever complexity of applications. In order to connect this number of processing elements, providing a interconnection infrastructure is essential. Network on Chip (NoC) is used as an efficient architecture for chip multiprocessors, which brings a reasonable performance and high scalability. A relevant challenge in NoC architectures is latency that increased in the presence of network contention. In this paper, we utilized virtual channels in buffer structure of routers. We calculated end to end latency and link utilization in NoCs by considering different numbers of virtual channels ranging from 2 to 5. The simulation results show that using virtual channels for specified routers in 2D mesh 4x4 decreases end to end latency up to 17% . It also improves link utilization and throughput significantly.

Keywords: *Network on Chip, virtual channel, path-based contention, latency.*

1. Introduction

With technology advances, the number of cores integrated into a single chip is increasing[1]. Traditional infrastructures such as non-scalable buses cannot handle the complexity of inter-chip communications. NoC (Network on Chip) architectures were appeared as a new paradigm of SoC design offering scalability, reusability and higher performance [1],[2]. NoC is a set of interconnected switches, with IP-cores connected to these switches.

Nowadays, one of the principal research issues that on-chip interconnection network designers Encountered is the network's performance and efficiency [6]. In other words, when the network becomes congested some problems such as contention which would be have a direct impact on NoC's delay appeared that should be solved. Virtual channels are a solution for contentions but the way and the number of them that should be use is a significant challenge[4],[5],[8].Different application specific NoCs have specified traffic pattern[13]. Occasionally, the data flow control of multiple paths may overlapped in these traffic patterns. When two or more packet contend for one path, congestion appears and as said, consequently the network influenced drastically by delay.

This discussion leads us to focus on delay as major challenges in data transmitting of specific-application NoCs[15]. In this article we proposed a solution by the help of virtual channels to reduce delay.

Generally, in NoC architectures, the global wires are replaced by a network of shared links and multiple routers transmitting data flows through the cores. As said, one of the main problems in these architectures is traffic congestion that should be avoided in order to optimize the efficiency and performance of NoCs.

In addition, the contention that created in some special simultaneous data flows through the cores and path also decreases the network performance, especially latency and throughput. We should consider that it is very hard to eliminate the contention completely in such networks but there are some solutions to control these contentions and degrade the bad impact of it on the whole data flows and performance in the network[16].

In this work by considering a special core graph as a sample, we use our proposed technique foe specified cores that contention would happen in them.In the technique proposed in this paper for solving contention in NoCs, virtual cannels are used for controlling path-based contentions.

The structure of this article is organized as follows: Section 2 reviews the fundamental structure of NoC, preliminaries are presented in Section 3. Following that,

in Section 4, path-contention issue is described by motivational example and an efficient allocation virtual channels are proposed for some routers in section 5. Experimental results are given and discussed in Section 6. Finally, concludes the paper.

2. Fundamental structure of NoC

NoC architecture consists of three components : routers, links and processing elements[14]. as shown in Figure 1 each router is composed of five input and five output ports. One pair of input/output ports is connecting to processing element while the other four pairs are connected to four adjacent routers in four directions (north, south, east and west). Every input port has a FIFO buffer and link controller, The purpose of buffers is to act as a holding area, enabling the router to manipulate data before transmitting it to another router and LC controls the empty area in the buffer for coming a new packet/flit from neighbor routers. The transferring of data between the input and output ports of the switch is carried out by the crossbar, whose active connections are decided by the arbiter which is a circuit used to make decisions which controls the crossbar.

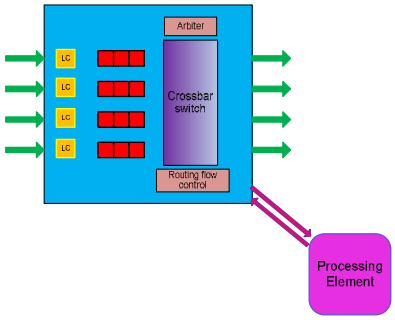


Fig.1. structure of tile (router and processing element)

In this paper, we consider a 2-D mesh as our topology with XY routing where the packet is first routed on the X direction and then on the Y direction before reaching its destination [7].

Wormhole switching is a well-known technique of multiprocessor network[4]. The structure of buffers in the wormhole switching is based on flit units that allows minimization of the size of buffers; In other words, power consumption reducing is caused by flit segmentation instead of packet segmentation.

Deadlock is a critical problem in wormhole switching that occurs when one router used by multiple data flows which each of them has different destination. By using a single buffer for each port in the router structure, one data flow behind of another should be waited until the area of buffer on related port to becoming empty.

Virtual channels originally introduced to solve the deadlock problem, although and therefore they can be used to improve network latency and throughput[8]. Assigning multiple virtual paths to the same physical channel is the nature of VC flow control. General structure of virtual channel is presented in figure2. As shown each input port has a physical buffer that divided into two virtual channels, it means that instead of one channel which can be placed by four flits we can use two separated channels that each of them can be managed two flits alone.

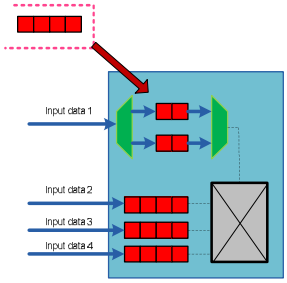


Fig.2. virtual channel

3. PRELIMINARIES

In this section, we introduce some related concepts first. The network contention that occurs into network on chip communications can be classified into three types: source-based contention, destination-based contention and path-based contention[3].

The source-based contention (figure.3(a)) is appeared when two variety traffic flows are distributed from the same source on the same link .The destination-based contention (figure.3(b)) occurs when two traffic flows have the same destination, in this case it is common to see a contention on the link which two traffic flows are leading to the same destination at the end of the considered link. the path-based contention (figure.3(c)) occurs when two data flows which come from the different sources, and go towards the different destinations contend for the same links somewhere in the network, (in other words some links that are common between two traffic flows are prone to contention.)

In the source-based contention, time scheduling method for transmission of data can be a good solution for avoidance of contention inside of specified source core. Similarly, in the destination-based contention, scheduling

packets to send to the same destination can has significant impact on reducing this kind of contention.

Unlike two previous methods path-based contention are not improved by timing technique, so another methods are required for optimizing this contention.

Network contention issue affects performance, latency and communication energy consumption.

For better explaining the technique proposed in this paper, we firstly describe the following definitions:

Definition: The core graph is a directed graph G(V,P). Each vertex $v_i \in V$ indicates a core in communication architecture. A directional path $p_{i,j} \in P$ illustrates a communication path for data flow from core-source v_i to the core-destination $v_{j.}$ The weight of $p_{i,j}$ that is shown by $W(p_{i,j})$ represented the data flow volume (bits) from core v_i to core $v_{j.}$.

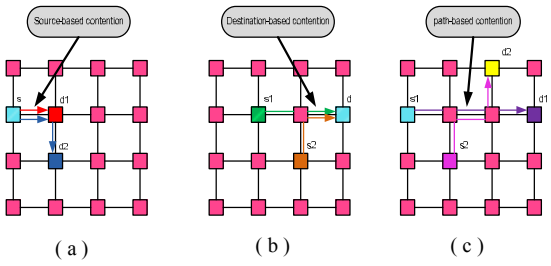


Fig.3. (a) source-based contention (b) destination-based contention (c) path-based contention

Every specific path from core v_i to core v_j consists of some links that demonstrated by $L_{m,n}$ which means a directed link from c_m to v_n (it is obvious that v_m , v_n are always considered as adjacent cores). As said in section… when two or more paths contend at the same time for the same link for transferring their data flow, path-contention occurs. $\alpha(L_{i,j})$ is a parameter used for a link from core v_i to core v_j that shows how many path contends simultaneously on accessing to it.

4. Motivational example

As a starting point we assume to have an application that be mapped onto a 2-D mesh topology.

the communication between the cores of the NoC are exposed by the core graph[9]. The connectivity and link bandwidth are presented by the NoC graph. As shown in figure.4 general application such as figure.4(a) can be mapped (as with any mapping algorithm) onto a 2-D mesh topology with dimension of 4×4, to show the impact of path-based contention consider the example of figure.4(b) which represent a network of switches (cores are ignored for simplicity) and several paths onto links are shown.

We apply XY routing and wormhole switching for data transmission with 4 flit per packet. As seen in figure.4(c) there are content in some paths, for instance five paths have a common link across transmitting their data.

Each specific path that is under XY routing in the traffic pattern shown in Figure.4(c) , composed of some links that explain as bellow:

$P_{0,15} = l_{0,1} + l_{1,2} + l_{2,3} + l_{3,7} + l_{7,11} + l_{11,15}$

$P_{1,11} = l_{1,2} + l_{2,3} + l_{3,7} + l_{7,11}$

$P_{7,15} = l_{7,11} + l_{11,15}$

$P_{6,15} = l_{6,7} + l_{7,11} + l_{11,15}$

$P_{5,11} = l_{5,6} + l_{6,7} + l_{7,11}$

Notice that Path-contention occurs on the paths that have overlapped on the same links. When $p_{i,j} \cap p_{p,q} \neq \emptyset$, then there would be at least one path contention. we calculate the content of α for each link that has path-contention in this traffic pattern.

$\alpha(l_{1,2}) = 2$, $\alpha(l_{2,3}) = 3$, $\alpha(l_{3,7}) = 2$, $\alpha(l_{7,11}) = 5$,

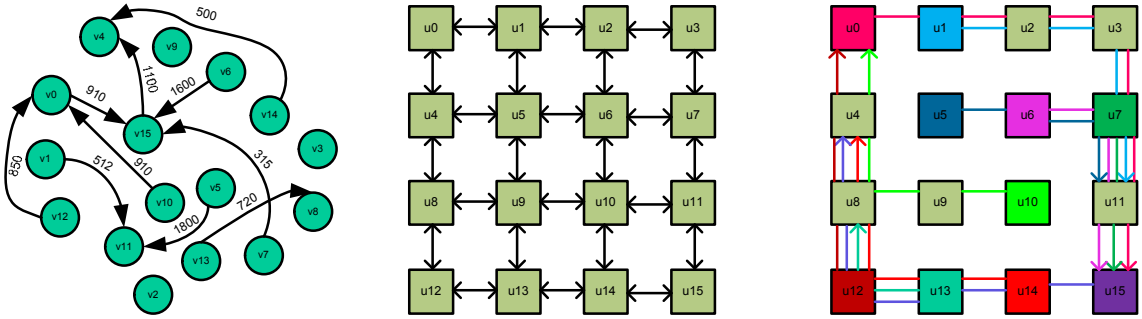
$\alpha(l_{11,15}) = 3$, $\alpha(l_{6,7}) = 2$

Node u5 to node u15 wants to send a packet based on XY routing, it goes forward two hops to east (first hop to u6 and second hop to u7) and then goes down as two hops.

Similarly, node u1 to sent a packet that allocated to u15 destination should goes east to delivery to u3, then goes down as three hops. Like two previous data flows, the packets belong to node u5, u0 and u7 are delivered to destinations across some common links.

As described in the mentioned scenario five flows of data can be inferred some paths, in more detailed some links, are common, hence these can be prone to content. In this example, the link between u7 and u11 has five different data flows, as shown in figure.4(c) although It may seen that some of these contention can be appeared as a destination-based contention, we address them to path-based contention in this paper.

The ratio of path-based to source-based contention and the ratio of path-based to destination-based contention increase linearly with the network size [3]. therefore in this paper we focus to reduce path-based contention since this has the most significant impact on the packet latency and can be mitigated through the mapping process. hence to overcome this problem we proposed using virtual channels in this paper.

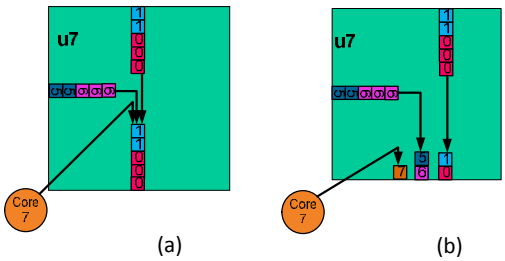


(a) (b) (c)

Fig.4. (a) core graph (b) NoC graph (c) mapping

5. Proposed Technique

In this section we focus on buffer structure as a fundamental element of router to reduce contentions. using a single buffer into router structure does not aid the contention problem remarkably. Take the case of with no virtual channels, flits 0 and 1 will prevent flits 5, 6 and 7 from advancing until the transmission of flits 0 and 1 have been completed (figure.5).



(a) (b)

Fig.5. (a) router7 without VC (b) router7 with three VCs

In this paper, we propose the use of virtual channels for some routers that are placed on the path contention. The number of virtual channels for any port of each router can be modified. Increasing the number of virtual channels has a direct impact on router performance, through their effect on the achievable hardware cycle time of the router[4]. For example, in u7 we can define three VCs for south port, one VC for flits coming from north port (u0, u1),one VC for flits coming west port (u5, u6) and one single buffer(one VC) for processing element port.

The paths u0 à u15, u1 à u11, u7 à u15, u5 à u11 and u6 à u15 have common links, implicitly there are contention pending these paths. however, the flits of u5 and u6 from one virtual channel and on the other hand the u0 and u1 flits go through its destination by another virtual channel and u7 processing element has a single virtual channel separately.

In this paper, we consider 5 flits as a buffer sizing for any port of each router. In the virtual channel allocation how many flits we should allocate to each virtual channel Based on this assumption there are 5 flits that should be divided between virtual channels proportionally. As an illustration, u1 has two virtual channels on output east port: one virtual channel is allocated to u0 and another belongs to u1 processing element. In this case considering to the volume of data can determine the number of flits in each virtual channel. The data volume of u0 is 910Mb and the data volume of u1 is 512Mb (figure.4(a)). Hence, u0 gives 3 flits because of more data volume and 2 remaining flits are allocated to u1.

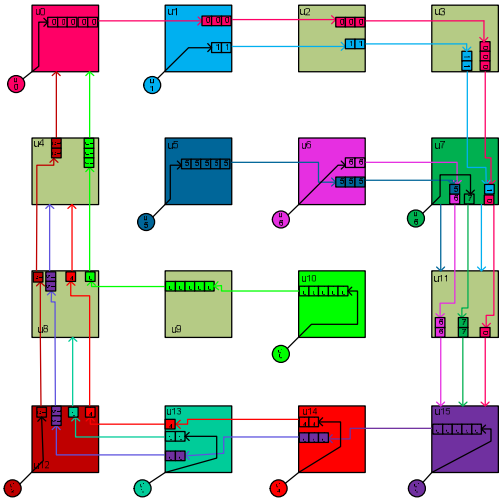


Fig.6. VCs for some routers on path-contention

6. Experimental Results

In this paper we have used HNoC which is developed in Omnet++ environment [18] in order to evaluate the functionality of Networks-on-Chip[10],[11],[12]. This simulator makes it possible to assess different traffic patterns with different packet generation rate. However, We define 4×4 mesh and 4B as flit sizing.

We first evaluate our network from the path-based contention point of view and routers that encounter with this kind of contention are selected. After that, virtual channels are allocated to specified ports of these routers that path-based contention occurred in them. For the various numbers of VCs ranges from 2 to 5, different results are achieved from simulation.

Simulations are conducted to evaluate the end to end latency, link utilization and throughput. However, the most important achievement is the fact that the end to end latency is reduced to 17% by increasing the number of virtual channels as seen in figure7.

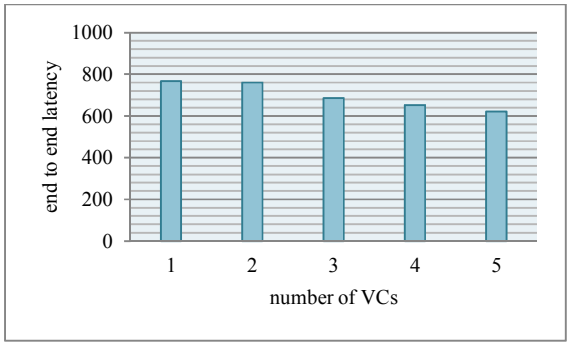


Fig.7. average end to end latency results for different number of VCs

By modifying the number of VCs the link utilization differs. Figure8 show that using 5 VCs has the highest link utilization in compared with 2, 3 and 4 VCs. More virtual channels lead to more virtual independent path between routers hence, the bandwidth of links are utilized efficiently.

Throughput can be defined in a variety of different ways depending on the specifics of the implementation[17]. We used the definition of throughput based on[17] where for massage passing system, TP can be calculated, as follows in Eq.(1):

$$TP = \frac{(total\ message\ completed) \times (message\ length)}{(number\ of\ IP\ blocks) \times (total\ time)} \qquad (1)$$

As shown in figure9 we calculated the number of received packets which are equal to (total message) × (message length). In the simulation process, the number

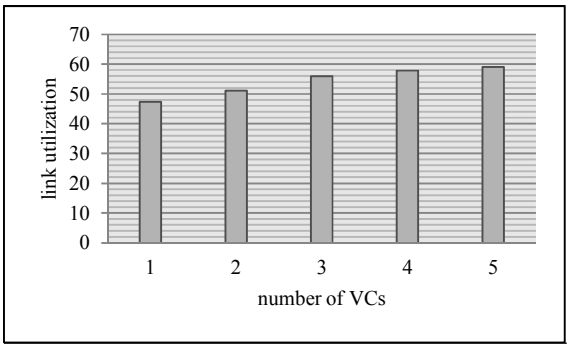


Fig.8. average link utilization results for different number of VCs

of IP blocks has been 16 and the total time is 100 µs. Also. according to proposed technique in this paper , the number of virtual channels varies from 2 to 5. The TP results are obtained from equation 1 can be seen in table 1. In this table, we observe that throughput(TP) is slightly higher when using 5 VCs compared to the case of without VC.

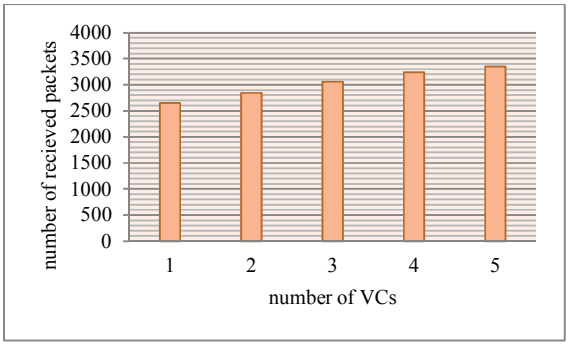


Fig.9. average number received packets results for different number of VCs

Table.1. throughput for different number of VCs

Number of VCS	Throughput
1	0.0032022
2	0.0035555
3	0.0038215
4	0.0040455
5	0.0041890

7. Conclusions

In this paper we have explored the influence of virtual channels in the presence of path contention. In application-specific NoC that mapped on any topology , path-contention is appeared as a problem. we show that selecting the appropriate number of virtual channels based on data volume for some specified routers can

significantly improve properties of network contention. Simulation has been carried out with a number of different virtual channels. the experimental results show that by using our proposed method an improvement is obtained in end to end latency, link utilization and throughput parameters.

References

[1] Benini, L., & De Micheli, G. (2002). Network on Chip: A New SoC Paradigm. IEEE Trans. Computer, 35(1), 70.

[2] Daly, W., & Towles, B. (2001). Route Packets, Not Wires: On-Chip Interconnection Networks. Proceeding of Design Automation Conference (DAC), 684.

[3] Chou, C. L., & Marculescu, R. (2008). Contention-Aware Application Mapping for Network on Chip Communication Architecture. IEEE International Conference, 164.

[4] Duato, J., Yalamanchili, S., & Lionel, N. (2002). Interconnection Networks. Morgan Kaufmann.

[5] Dally, W. J., & Towles, B. (2003). Principles and Practice of Interconnection Networks. The Morgan Kaufmann Series in Computer Architecture and Design.

[6] Bjerregaard, T., & Mahadevan, S. (2006). A Survey of Research and Practices of Network-on-Chip. Journal of ACM Computing surveys (CSUR), 38(1).

[7] Kakoee, M. R., Beatacco, V., & Benini, L. (2011). A Distributed and Topology-Agnostic Approach for On-line NoC Testing. Networks on Chip (NoCS), Fifth IEEE/ACM International Symposium, 113.

[8] Mullins, R., West, A., & moore, S. (2012). Low-Latency Virtual-Channel Routers for On-Chip Networks. Proceedings of the 31st annual international symposium on Computer architecture, 188.

[9] Murali, S., & Mecheli, G. De. (2004). Bandwidth-Constrained Mapping of Cores onto NoC Architectures. Design, Automation and Test in Europe Conference and Exhibition, 896.

[10] Walter, I., Cidon, I., Ginosar, R., & Kolodny, A. (2007). Access Regulation to Hot-Modules in Wormhole NoCs. Proceedings of the First International Symposium on Networks-on-Chip, 137.

[11] Ben-Itzhak, Y., Zahavi, E., Cidon, I., & Kolodny, A. (2013). HNOCS: Modular Open-Source Simulator for Heterogeneous NoCs. *International Conference on* Embeded Computer Systems(SAMOS), 51.

[12] Ben-Itzhak, Y., Cidon, I., & Kolodny, A. (2012). Optimizing Heterogeneous NoC Design. Proceeding of the International Workshop on System Level Interconnect Prediction, 32.

[13] Murali, S., & De Micheli, G. (2004). Bandwidth-Constraint Mapping of Cores onto NoC Architectures. Proceeding of Design, Automation and Test in Europe Conference and Exhibition, 2, 896.

[14] Shiuan Peh, L., & Jerger, N. E. (2009). On-Chip Networks (Synthesis Lectures on Computer Architecture). Morgan and Claypool publisher, Mark D. Hill Series Editor, Edition1.

[15] Seiculescu, C., Rahmati, D., Murali, S., Benini, L., De Micheli, G., & Sarbazi-Azad, H. (2013). Designing Best Effort Networks-on-Chip to Meet Hard Latency Constraints. The Journal of ACM Transaction on Embedded Computing Systems (TECS), 12(4), 109.

[16] Lin, S. Y., Huang, C. H., & Chao, C. H. (2008). Traffic-Balanced Routing Algorithm for Irregular Mesh-Based On-Chip Networks. The Journal of IEEE Transaction on Computers, 57(9).

[17] Pande, P. P., Grecu, C., Jones, M., & Ivanov, A. (2005). Performance Evaluation and Design Trade-Offs for Network-on-Chip Interconnect Architectures. The Journal of IEEE Transaction on Computers, 54(8), 1025.

[18] Varga, A. (2012). The OMNeT++ discrete event simulation system. Proceedings of the European Simulation Multi conference (ESM'2001), 319.

Mona Soleymani received her B.Sc. from Tehran central branch of Islamic Azad University, Tehran, Iran in 2010 in Hardware engineering and her M.S. from science and Research Branch of Islamic Azad University, Tehran, Iran in 2013 in Computer architecture engineering. She is currently working toward the Ph.D. in computer architecture engineering at the Science and Research Branch of IAU. Her research interests lie in Network on Chip and routing, switching and mapping issues in on-chip communication networks.

Fatemeh Safinia received her B.Sc. from Tehran central branch of Islamic Azad University,Tehran, Iran in 2010 in Hardware engineering and her M.S. from science and Research Branch of Islamic Azad University, Tehran, Iran in 2013 in Computer architecture engineering. She is currently working toward the Ph.D. in computer architecture engineering at the Science and Research Branch of IAU. Her research interests lie in Network on Chip and routing, switching and mapping issues in on-chip communication networks.

Somayyeh Jafarali Jassbi received her M.Sc. degree in computer architecture from Science and Research Branch of Islamic Azad University(SRBIAU), Tehran, Iran in 2007. She also received her Ph.D. degree in computer architecture from SRBIAU, Tehran, Iran in 2010. She is currently Assistant Professor in Faculty of Electrical and Computer Engineering of SRBIAU. Her research interests include Information Technology(IT), computer arithmetic, cryptography and network security.

Midia Reshadi received his M.Sc. degree in computer architecture from Science and Research Branch of Islamic Azad University (SRBIAU), Tehran, Iran in 2005. He also received his Ph.D. degree in computer architecture from SRBIAU, Tehran, Iran in 2010. He is currently Assistant Professor in Faculty of Electrical and Computer Engineering of SRBIAU. His research interests include Photonic NoCs, fault and yield issues in NoCs, routing and switching in on-chip communication networks. He is a member of IEEE.