

## Performance Analysis of Mesh-based NoC's on Routing Algorithms

Anala M. R., Amit N. Subrahmanya, Allbright D'Souza

Department of Computer Science and Engineering, R V College of Engineering, India

---

### Article Info

#### Article history:

Received Nov 30, 2017

Revised Jan 19, 2018

Accepted Feb 7, 2018

---

#### Keyword:

Network-on-chip

Performance

Routing algorithm

Switching technique

Traffic pattern

---

### ABSTRACT

The advent of System-on-Chip (SoCs), has brought about a need to increase the scale of multi-core chip networks. Bus Based communications have proved to be limited in terms of performance and ease of scalability, the solution to both bus – based and Point-to-Point (P2P) communication systems is to use a communication infrastructure called Network-on-Chip (NoC). Performance of NoC depends on various factors such as network topology, routing strategy and switching technique and traffic patterns. In this paper, we have taken the initiative to compile together a comparative analysis of different Network on Chip infrastructures based on the classification of routing algorithm, switching technique, and traffic patterns. The goal is to show how varied combinations of the three factors perform differently based on the size of the mesh network, using NOXIM, an open source SystemC Simulator of mesh-based NoC. The analysis has shown tenable evidence highlighting the novelty of XY routing algorithm.

*Copyright © 2018 Institute of Advanced Engineering and Science.*

*All rights reserved.*

---

### Corresponding Author:

Anala M. R.,

Departement of Computer Science and Engineering,

R V College of Engineering,

R V Vidyanikethan Post, Mysore Road, Bengaluru - 560 059, India.

Email: analamr2004@gmail.com

---

## 1. INTRODUCTION

Network-on-Chip is an essential communication subsystem for System-on-Chip (SoC) architectures as it has the capability to meet the requirements of scalability, ease of fabrication and advancing technologies. A Multiprocessor System-on-Chip (MPSoC) is the agglomeration of microprocessors, microcontrollers, memory blocks, and more such analog and digital interfaces, counters, power management circuits, etc. The system is built to make use of a large number of Intellectual Property (IP) cores for high processing power. Buses and point-to-point (P2P) links were used as communication infrastructure for SoC at its inception, but it failed to fulfill future communication requirements such as higher scalability, stable performance and acceptable power consumption [1], [10]. Hence, there was a need to veer away from computation centric models to communication centric models [3]. These problems were solved Network-on-Chip (NoC) models. Coupled with minimal complexity and high scalability, it provides high versatility along with quality performance.

The pith of the NoC approach is to use a macroscopic concept such as packet-switching and infrastructural IPs which allows access to individual functional-IP block. The most important features that distinguish NoC architectures are network topology, routing, and packet priority. However, parameters such as buffer size, power consumption, and area overhead should also be considered while dealing with network performance. The NoC architecture is comprised of three parts - Resources, Routers and Resource Network Interface [2].

- a. Resources: They are the processing elements that are selected based on functionality, size and speed requirements of a network. E.g. Microprocessors, Microcontrollers, ADCs, DACs, FPGAs, Audio Video controllers or Memory Block controllers.
- b. Resource to Network interface: They provide the interface for the communication between the resource and the routers. They are involved with the conversion of packets of data into a suitable form for transportation (flits or phits) and its conversion back to the original packet of data.
- c. Router: It is responsible for the transfer of data from the source IP to destination IP through routers in accordance with the routing algorithm.

The two main resources along with the IP that characterizes NoCs are buffers and channels. To increase the resources allocation for each packet, a physical channel is multiplexed into a number of virtual channels. It is via these channels that the packets are transferred from the source IP to the destination IP by deferring to the routing algorithm as prescribed. Each channel is composed of buffers that allow the free movement of packets in a more discrete fashion.

As given in the literature survey below, the research so far has been focused on grading the performance of routing algorithms based on latency and packet injection rate. However, there has been a lack of focus on size of the mesh, which creates a dramatic change in the performance of the routing algorithm. Hence, this experiment is focused on emphasizing the effect of size on performance through the simulations carried out on NOXIM.

### 1.1. Performance dependencies

As previously mentioned, an NoC is distinguished based on variables such as network topology, routing algorithm, switching-technique, and traffic pattern. NoC synthetic traffic patterns are essential tools for NoC performance assessment and devising new architectures. Performance of the NoC with respect to the size of the network is dramatically dependent on these factors, the main metrics used for the evaluation of performance are throughput and average packet delay, but our focus in this paper is on Global Average Delay (latency).

#### 1.1.1. Topology

Network topology is a spatial arrangement of nodes connected to form a network. A distinguished array of topologies such as Mesh, Torus, RiCoBit and BFT etc. has been designed to suit different systems and their niche applications. However, for all general purpose simulation and most real-world applications, Mesh topology is used due to its simplicity in construction and designing routing algorithms.

**Mesh Topology:** It is the easiest to fabricate of all the arrangements as all nodes are arranged in the order of an  $m \times n$  matrix. The matrix form factor allows the network to be flat and very thin.

#### 1.1.2. Routing

Routing algorithms determine the path selected by a packet to reach its destination. Routers have been de-signed such that at each intermediary router, a decision is made to transfer the packet in a particular direction i.e. north, south, east, west or local (to IP). Routing algorithms have two basic approaches 1) Deterministic and 2) Adaptive [2], [6]. The deterministic algorithm needs no information other than the source address and the destination address to determine a particular route for the packet. The adaptive algorithm has no predefined route and is determined dynamically at each router based on the network load and traffic conditions. The deterministic algorithms are proven to be deadlock and livelock free since each packet will eventually reach its destination. However, packet latency in such an algorithm will increase drastically if multiple packets need to travel across the same channel [6]. On the other hand, adaptive algorithms are prone to deadlock and livelock, be-cause they are dependent on the traffic density and make decisions based on its availability. To avoid deadlocks and livelocks caused by packets cycling around, certain turns need to be eliminated [6]. Due to computation dependency, it is less efficient to use adaptive routing at lower packet injection rate. The most common routing algorithms are as follows:

- a. XY routing: This is a deterministic routing algorithm in which the packet first traverses across the X axis until the source and destination are aligned in the same column, and then it traverses in the Y-axis direction till the destination is reached. An alternate design to this would be the YX routing, which has the packet move along the Y-axis first followed by those in the X-axis.
- b. West first: This is an adaptive algorithm in which packets must first travel in the westward direction and then move towards the destination without making any movements west [7].
- c. North last: Analogous to the previous, this is an adaptive algorithm makes adaptive decisions in the beginning and then finally allows the packet to travel northwards to its destination if needed [7].
- d. Negative First: In this adaptive algorithm, the packet must adaptively make all negative directional movement in the beginning and then, all the positive movements. Negative movements here are defined as west and south while positive movements are north and east [7].

- e. Odd-Even: This adaptive routing prohibits east to north turns and east to south turns at any node in an even column. It also disallows north to west turns and south to west turns at any node in an odd column [8].
- f. DyAD: Dynamically switching between Adaptive and Deterministic routing is the culmination of the advantages of both deterministic and adaptive algorithms. It could be a combination of any two algorithms, such that it continuously monitors its local network load and makes decisions based on this information [6].

#### 1.1.4. Synthetic traffic patterns

These are speculative models of traffic to demonstrate how the simulation would progress. They differ from realistic models that are based on certain benchmark programs. Traffic patterns predefine the destination node for a particular source node. In our simulations, we used the following [3]:

- a. Random pattern: Each source node dispatches packet to another random node at the specified packet injection rate.
- b. Transpose 1: It is a matrix-transpose pattern in which each PE present in a chip at location (I, J) dispatches a message to the PE at  $(n - 1 - I, n - 1 - J)$  for an  $(n \times n)$  size NOC.
- c. Transpose 2: The PE present at chip (I, J) communicates only with the node (J, I) in the network.

#### 1.2. Literature survey

NoCs have become the most sought-after architecture due to its overwhelming advantages over the primitive SoC model. The ability to provide higher scalability and re-usability gives it an edge over its competitors. Extensive research has been done in developing new topologies and routing algorithms. Less conventional topologies like the RiCoBit have been explained with fleshed out information and also performance evaluations in many papers [9]. The ratio of average throughput and average packet delay is computed to compare different simulations. It is shown that Odd-Even routing is superior to XY routing with a ratio of 2.5358 as compared to 2.1126 respectively [10]. The performance of XY routing is preferable at uniform traffic patterns. For non-uniform traffic, the load at the center of the network is higher than the average network load leading to a traffic hotspot [11]. Another disadvantage is that if a node is faulty, that packet will remain blocked on that path [12]. However, XY routing is simple and has very low hardware overhead as compared to adaptive routing [13]. Under distributed traffic patterns, XY routing and Odd-Even routing work better. However, in directed traffic for most practical circumstances, DyAD routing works better than most of the other routing algorithms [14]. Investigating the design of a NoC by evaluating its performance requires the availability of fast and accurate simulation tools. Noxim is a SystemC based open source, cycle accurate platform for the simulation of both conventional wire-based NoCs and emerging WiNoC architectures. Several documents are available for getting started on the simulator. The routing algorithms used in this paper have been developed with adequate documentation. An elaborate article describing the general system structure of the NoC is available to provide the necessary knowledge for further study [1].

## 2. RESEARCH METHOD

We evaluate the performance of mesh networks of sizes varying (from 4x4 to 10x10) based on the afore-mentioned dependencies such as traffic patterns, routing algorithms and switching schemes. Traffic patterns considered are random, transpose 1 and transpose 2 [3]. The routing algorithms considered are XY, west first, north last, negative first, odd even & DyAD [6]-[9]. Switching schemes used are neighbors on path (NOP) and buffer level strategy. All the tests were conducted on NOXIM, an open source SystemC Simulator of mesh-based NoC [1]. The results have been obtained with respect to global average delay at different packet injection rates (PIR) or throughput for the following Table 1 configurations:

Table 1. Simulation Settings

Setting	Value
Buffer Depth	4
Size of Flits(in bits)	32
Dyad Threshold	0.6
Simulation Time(in cycles)	10000
Packet Size(in flits)	8
Packet Injection Rate	0.008 to 0.020
Selection Strategy	Buffer Level, NOP

### 3. RESULTS AND ANALYSIS

The performance of NoCs is evaluated based on two different traffic patterns (random, transpose 2) and switching schemes (Neighbors on Path, buffer level) have been made across four different Injection rate, giving rise to a descriptive listing of Figure 1 and Figure 2 that illustrate how each routing algorithm performs. Intuitively, we can relate the size of the mesh with the delay as directly proportional, since larger time would be required to transfer the packets across larger networks. But, routing algorithms have been designed differently to handle light and heavy loads. More often than not, Traffic patterns may favor certain paths more than others due to common destination nodes this can be better shown by the following two traffic patterns.

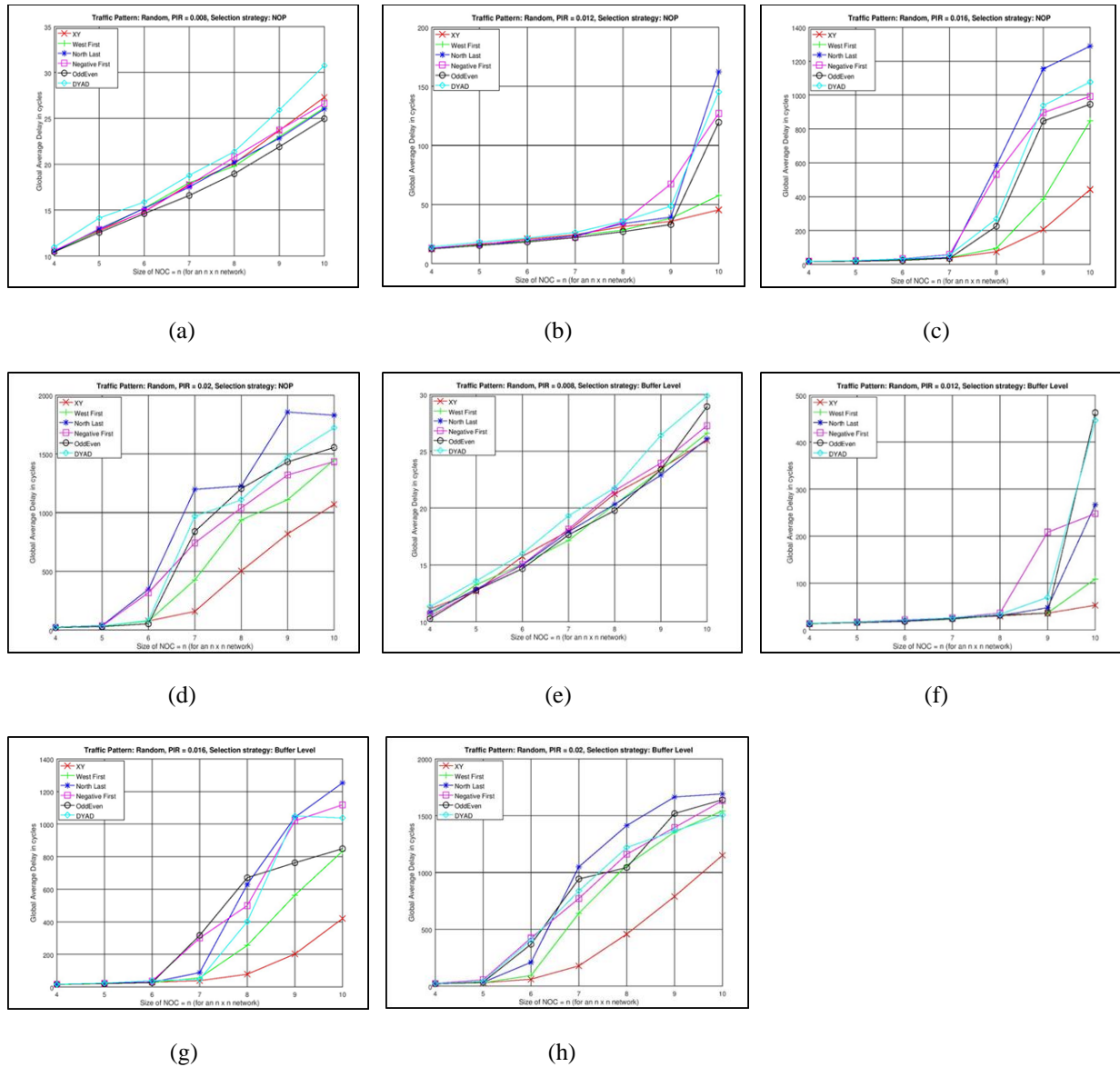


Figure 1. (a) PIR = 0.008, Traffic Pattern: Random on switching schemes: NOP, (b) PIR = 0.012, Traffic Pattern: Random on switching schemes: NOP, (c) PIR = 0.016, Traffic Pattern: Random on switching schemes: NOP, (d) PIR = 0.02, Traffic Pattern: Random on switching schemes: NOP, (e) PIR = 0.008, Traffic Pattern: Random on switching schemes: Buffer Level, (f) PIR = 0.012, Traffic Pattern: Random on switching schemes: Buffer Level, (g) PIR = 0.016, Traffic Pattern: Random on switching schemes: Buffer Level, (h) PIR = 0.02, Traffic Pattern: Random on switching schemes: Buffer Level

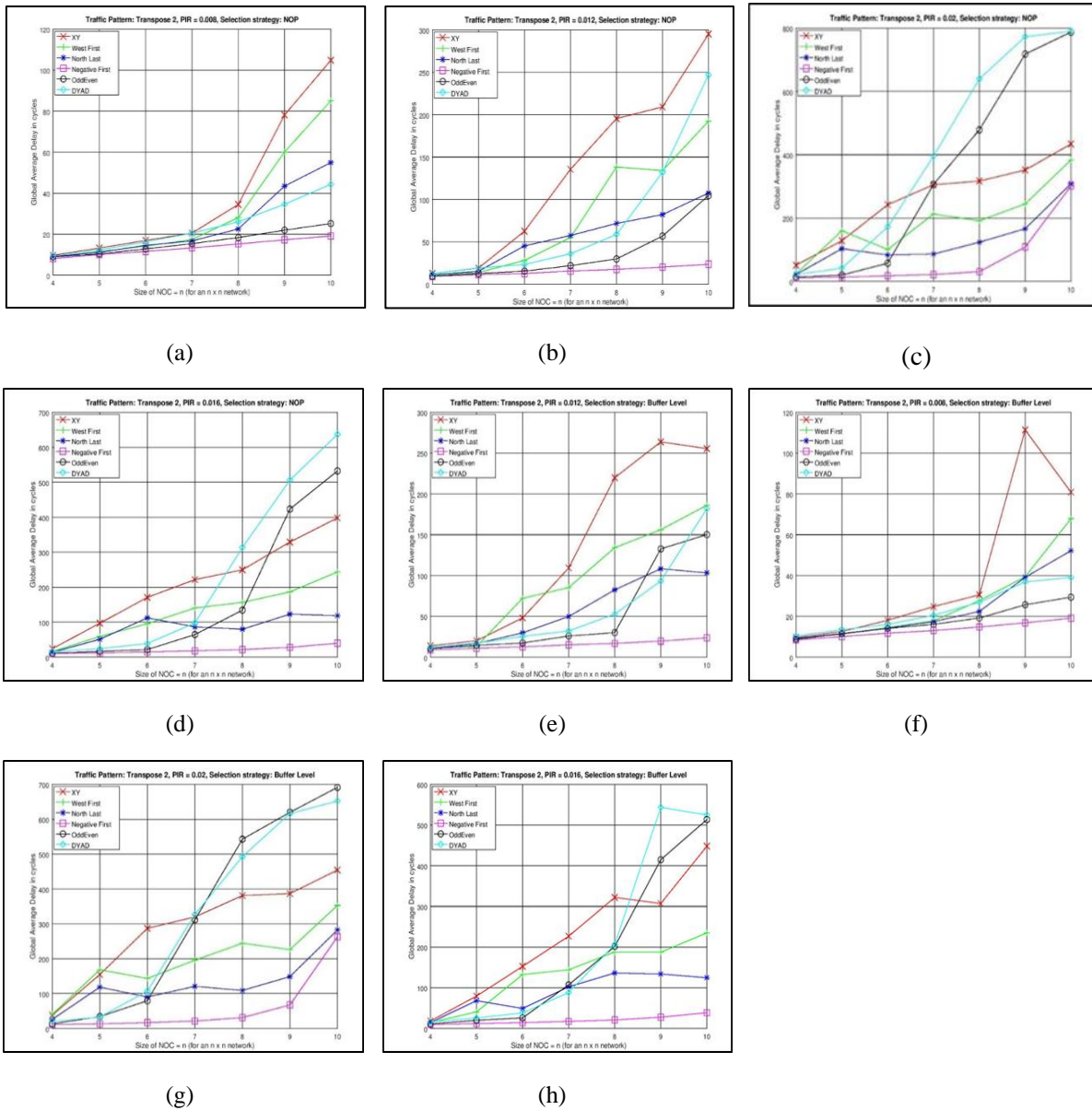


Figure 2. (a) PIR = 0.008, Traffic Pattern: Transpose 2 on switching schemes: NOP, (b) PIR = 0.012, Traffic Pattern: Transpose 2 on switching schemes: NOP, (c) PIR = 0.016, Traffic Pattern: Transpose 2 on switching schemes: NOP, (d) PIR = 0.02, Traffic Pattern: Transpose 2 on switching schemes: NOP, (e) PIR = 0.008, Traffic Pattern: Transpose 2 on switching schemes: Buffer Level, (f) PIR = 0.012, Traffic Pattern: Transpose 2 on switching schemes: Buffer Level, (g) PIR = 0.016, Traffic Pattern: Transpose 2 on switching schemes: Buffer Level, (h) PIR = 0.02, Traffic Pattern: Transpose 2 on switching schemes: Buffer Level

In Traffic pattern: Random, which transfers packets haphazardly to any destination node, in both buffer level and NOP selecting scheme, XY routing is seen to outperform the rest. As the overall distribution of requests is considered, no traffic hotspots exist. This makes the traffic distribution seem more uniform. Due to this, XY routing doesn't spend the time trying to determine and select free paths like the adaptive algorithms. This behavior of XY routing holds good even at high PIR and large mesh sizes as illustrated in Figure 1(d), Figure 1(e), Figure 1(i) and Figure 1(j). Making it more reliable than even DyAD which shows consistently higher delay than XY as shown in Figure 1(a) and Figure 1(b). However, XY routing demonstrates better delay than the other adaptive algorithms as shown in Figure 1(c) and Figure 1d). This may be accounted due to the fact that it has to determine which approach to proceed with and its complexity of hardware. DyAD also behaves as expected it has a higher delay at low PIR in both the selection strategies. However, its latency reduces with an increase in PIR as it becomes more stable. It has a very high delay in

comparison to other algorithms as seen in Figure 1(a) to Figure 1(e). This pattern is visible only in NOP. However, it shows improvement in latency with an increase in PIR under Buffer Level. The remaining adaptive algorithms have delays ranging from medium to high values.

In Traffic pattern: Transpose 2, XY routing has higher delay than its adaptive counterparts at low PIR such as 0.008 and 0.012 as shown in Figure 2(a), Figure 2(b), Figure 2(f), Figure 2(g), but with an increase in PIR at sizes larger than 8x8, it generally performs better than the adaptive counterparts DyAD and odd-even as indicated in Figure 2(c). Compared to the remaining adaptive algorithms, Negative First, as shown in Figure 2(g), Figure 2(h) shows high performance at medium to higher PIR. This is because it is better suited for sending packets to opposite ends as it adaptively allows initial negative movements and then adaptively positive movements, making the delays shorter, though its latency increases at smaller mesh due to extra time spent in making more adaptive decisions. North last shows low to medium delay across all figures and this may be because north last is adaptive in the beginning and deterministic in the end with its approach. Since along the principal diagonal is where most traffic will be concentrated, the adaptive first nature of north last makes it faster than the rest at low to medium PIR and low to medium size. West first is shows similar, if not slightly worse performance in comparison to North Last. DyAD is more stable at low PIR value 0.008 as shown in Figure 2(a) and Figure 2(f). Its performance worsens with an increase in both the PIR and the mesh size.

#### 4. CONCLUSION

In this paper, we have aimed at deriving and portraying the effect of various factors on performance with respect to the size of an NoC. It can be corroborated using the graphs that the deterministic routing algorithm - XY is not stable due to its fluctuating behavior for different combinations of traffic and mesh sizes (i.e. at low PIR and small mesh size). Adaptive algorithms like West First, North Last, Negative First and Odd-Even show that these algorithms are quite reliable in dealing with certain traffic patterns but suffer from larger delay across larger, less congested mesh structures due to increased packet latency caused by its more complex hardware. For large NoCs at higher PIRs, XY is undoubtedly a more attractive algorithm for most traffic patterns due to its lack of dependence on network load. Since the path of transaction never changes, it is independent of the path taken by alien packets as long as its own path is free. As larger meshes lack in the number of common paths, XY is definitely a better choice for a routing algorithm. Moreover, XY routing has a simple implementation thus having a small overhead during packet transit. As seen by the observations thus far, the results observed go hand in hand with the introduction, showcasing the overall effectiveness of XY routing algorithm over the other algorithms.

#### REFERENCES

- [1] Vincenzo Catania, *et al.*, "Cycle-Accurate Network on Chip Simulation with Noxim", *ACM Trans. Model. Comput. Simul.*, 27, 1, Article 4, pp: 1-21, August 2014.
- [2] Sudhanshu Choudhary and Shafi Qureshi, "Performance Evaluation of Mesh-based NoCs: Implementation of a New Architecture and Routing Algorithm", *IJAC*, August 2012, pp. 5-10.
- [3] Jayant Kumar Singh, "Performance Evaluation of different Routing Algorithms in Network on Chip", Master's Thesis June 2014.
- [4] Z. Sharifi, *et al.*, "Comparison of NoC Routing Algorithms using Formal Methods", July 2013, pp. 1-15.
- [5] Jingcao Hu Radu Marculescu, "DyAD: Smart Routing for Networks-on-Chip", April 25, 2004, pp:1-10.
- [6] Maksat Atagoziyev, "Routing Algorithms for on Chip Networks", December 2007, p. 79.
- [7] Christopher J. Glass and Lionel M. NI., "The Turn Model for Adaptive Routing", September 1994, pp:1-10.
- [8] Wang Zhang, *et al.*, "Comparison Research between XY and Odd-Even Routing Algorithm of a 2-Dimension 3X3 Mesh Topology Network-on-Chip", *WRI Global Congress on Intelligent Systems, GSIS'09, May 2009*, pp. 329-333.
- [9] Pan Hao, *et al.*, "Comparison of 2D MESH Routing Algorithm in NOC", *2011 IEEE 9th International Conference*, pp. 791-795.
- [10] M. Nickray, *et al.*, "Adaptive Routing Using Context-Aware Agents for Networks on Chips", *Design and Test Workshop (IDT), 2009 4th International*, pp. 1-6, 2009.
- [11] R. Holsmark, *et al.*, "Deadlock Free Routing Algorithms for Mesh Topology NoC Systems with Regions", *DSD 2006. 9th EUROMICRO Conference on*, pp. 696-703.