



**HAL**  
open science

## A Methodology for Design of Unbuffered Router Microarchitecture for S-Mesh NoC

Hao Liu, Feifei Cao, Dongsheng Liu, Xuecheng Zou, Zhigang Zhang

► **To cite this version:**

Hao Liu, Feifei Cao, Dongsheng Liu, Xuecheng Zou, Zhigang Zhang. A Methodology for Design of Unbuffered Router Microarchitecture for S-Mesh NoC. IFIP International Conference on Network and Parallel Computing (NPC), Sep 2010, Zhengzhou, China. pp.442-451, 10.1007/978-3-642-15672-4\_37. hal-01054965

**HAL Id: hal-01054965**

**<https://inria.hal.science/hal-01054965v1>**

Submitted on 11 Aug 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# A Methodology for Design of Unbuffered Router Microarchitecture for S-mesh NoC

Hao Liu<sup>1,3</sup>, Feifei Cao<sup>2</sup>, Dongsheng Liu<sup>3†</sup>, Xuecheng Zou<sup>3</sup>, Zhigang Zhang<sup>1</sup>

<sup>(1)</sup> Henan Electric Power Research Institute, Zhengzhou 450052, China; <sup>(2)</sup> Henan Electric Power Industrial School, Zhengzhou 450051, China; <sup>(3)</sup> Department of Electronic Science & Technology, Huazhong University of Science & Technology, Wuhan 430074, China)  
[freemansoc@126.com](mailto:freemansoc@126.com)

**Abstract.** Currently, most of Network on-Chip (NoC) architectures have some limitation in routing decisions. And it makes router nodes overloaded, and sequentially forms deadlock, livelock and congestion. A simple unbuffered router microarchitecture for S-mesh NoC architecture is proposed in this paper. Unbuffered router transforms message without making routing decision. Simulation results showed that S-mesh could get optimal performance in message latency compared with 2D-mesh, Butterfly and Octagon NoC architectures. The Design Compiler synthesis results showed that unbuffered router has obvious advantages on area, and it gets higher operation speed.

**Key words:** NoC, separated-mesh (S-mesh), unbuffered, low-latency, low-cost.

## 1 Introduction

### 1.1 Background

With the arrival of multicore era, traditional bus-based interconnect architectures have become bottleneck for the multicores communication. The Network on-Chip [1], [2] design paradigm is seen as the ultimate solution of enabling the integration of exceedingly high number of cores for the future on-chip communication network architectures. In NoC based system, messages are exchanged between cores using a network and packet switching paradigm. The messages are relayed from one core to another along the path constructed by routers and links. The design of NoC communication network architectures would be facing a major design challenge that is to construct an area-efficiency, low-latency, scalable on-chip communication network.

NoC is an emerging paradigm for communications with large VLSI systems implemented on single silicon chip. It brings forward a networking method to on-chip communication and brings about notable improvements over conventional bus systems, such as ARM AMBA, Wishbone, STBus, Core Connect, and so on. There are several architectures for NoCs such as Fat-tree [3], Mesh [4], Octagon [5] and Spidergon [6]. Fat-tree and 2D-mesh as two of the most popular topologies. Some NoC implementation and research, such as Nostrum [7], Æthereal [8], Raw network [9], Xpipes [10], Eclipse [4] have been implemented on top of Fat-tree, 2D-mesh in some

extent. From a perspective of these authors, of course, these architectures are topology-independent.

## 1.2 Overview of NoC architecture

A typical NoC chip is a matrix of resource slots containing integrated embedded processors or systems connected to each other via a multi-dimensional mesh/tree network. Therefore, a typical NoC system includes resource nodes, router nodes, links and network interface unit, and routing algorithms for meet the requirements of the different architectures.

The router nodes include the routing controller and an arbiter for resolving local route conflicts. The routing algorithm currently under consideration can be labeled pseudo-dynamic since it's allowed only for restricted dynamic routing in case of router conflicts. In addition, router architecture should be adjusted with different routing algorithms. These influencing factors potentially increase uncertainty condition for system performances, especially network latency, congestion, cost and other limitations. Because the routers do not exactly know subsequent routers working conditions in real time. So local performance optimization always makes the whole system performance worse. Using these architectures for extremely large systems is very difficult [3]. We believe that NoC router architecture should be simple, low-latency, low-cost, and the number of data buffers should be minimal in the future. In accordance with the view of NoC as a research field of SoC. We focus on constructing feasible, low-latency and low-cost communication-centric design.

## 1.3 Outline of this paper

The methodology of multicores system based on NoC will be changed from computation-centered to communication-centered. Moreover, what its key goal of on-chip network is to construct a high-performance on-chip communication network with low-latency, and scalability for multicore chip system.

In this paper, an unbuffered router architecture for Separation of Control and Data-transmission NoC architecture is proposed. This NoC architecture decouples the routing decision from router. The routers employ a pre-connection mechanism for the input channels and output channels that help to reduce the complexity of the crossbar matrix design. It also meets the aforementioned features of simple, fast and less buffer. The studies of this paper mainly embody in several aspects, such as network architecture, unbuffered router microarchitecture.

This paper is organized as follows: In section II, S-mesh NoC architecture is discussed, and router microarchitecture is illuminated in Section III, the simulation results are presented in Section IV and conclusions are provided in Section V.

## 2. S-mesh NoC architecture

S-mesh NoC [11] network borrows features and design methods from those used in parallel computing clusters and communication networks. This includes design conception that separate service and bearer control in communication network. It also in-

cludes the implementation way of centralize-distribution in communication system and IP carrying network, as well as the Message Transfer Part [12], which is a part of the Signaling System #7 [13] used for communication network. Nevertheless, these methods cannot be adopted directly.

The S-mesh is based on 2D-mesh topology. In the S-mesh system, the kernel communication network adopts circuit-switching mode. And the edge devices, such as resource nodes, adopt the packet switching mode. The S-mesh network architecture consists of three types of sub-networks: mesh-based data transmission network (DN), butterfly-based control network (CN) and local bypass network (BN). The S-mesh NoC consists (Fig.1) of resources, routers and connection process unit. DN connects the resources to its nearest routers. Each router should be connected to the CN as well as four other neighboring routers through BN and DN. A BN or DN link consists of two one-directional point-to-point buses. The functionalities of router nodes only undertake link layer functionalities and physical layer functionalities. Routers do not need to store any packets before forwarding though dedicated path has been established advanced. CN is responsible for the system resource management, routing decisions, and flow control. All resources are connected to interconnection fabric with a Network Interface Unit (NIU). The NIUs handle all communication protocols, which are used to make the network as a transparent communication network.

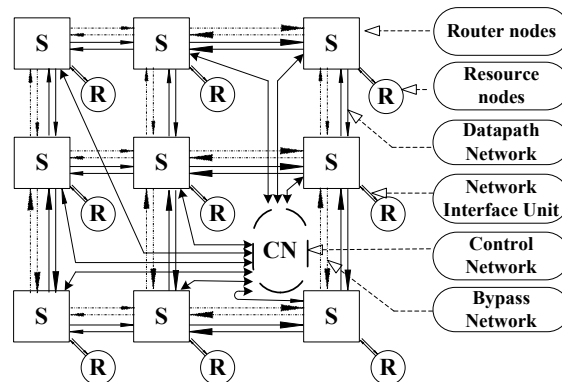


Fig.1. S-mesh NoC architecture.

## 2.1 Control Network and Datapath Network

Control Network architecture is similar to Fat-tree architecture. Routers are located at the leaf node of the Fat-tree, and CN is located at the root node of the Fat-tree. CN unit is designed not only for efficient commands between resources, but also for efficient movement of operands between special resources just like processors or computing units. CN uses iSLIP algorithm [14] to schedule each active request and acknowledge in turn or forward command. All on-chip dynamic data movement uses packet-based and connection-oriented communication over DN. This includes memory accesses, user-level DMA transfers, and I/O. In S-mesh, the packets header does not contain routing information, only contains labels and payload. DN network is

deadlock free and congestion-immunity due to connection-oriented communication mechanism.

## 2.2 Bypass Network

Using BN to transfer messages between adjacent cores would be better way than using DN. It responses only to its four neighbors, not acts as a router. It can efficiently reduce the traffic load of DN and signaling load of CN. Moreover, Router has two logically disjointed networks, which could be implemented as two separate networks, or as two logical networks using the two groups of physical wires. When we map the internal communications between process cores upon two-dimensional mesh networks, about 72.9% communication data occurs between the adjacent cores (routing distance is 1) [15]. The obvious local data would be transmitted through the BN. However, the global data would be transmitted through the DN. This mapping rule can ensure that local and global communication in different region could be transmitted at the same time, thereby to reduce congestion and promote transmission efficiency of the whole system.

The applicable condition is described in the pseudo code as below.

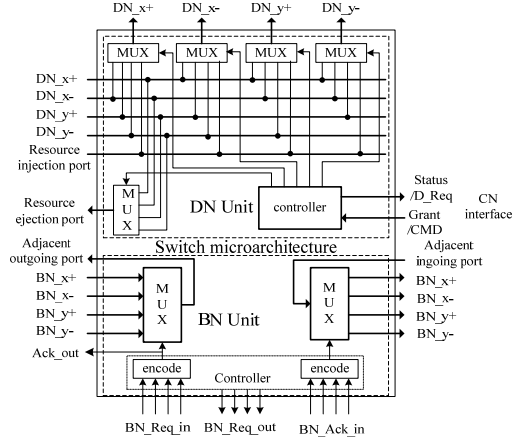
```
if (abs(XDestination - Xsource) =1 or abs(YDestination - Ysource) =1)
  and (abs(XDestination - Xsource) ≠ abs(YDestination - Ysource))
  {Data transmission via BN port without ControlNetwork;}
else
  {Data transmission via DN port with ControlNetwork;}
```

## 3 Router microarchitecture

The task of routers is to carry messages injected into the network to their final destination, following a defined determined routing path in advance. The router exchanges message flits from one of its input link to one or more of its output link under controlled by CN. Meanwhile, router can directly transfer message flits between adjacent resource nodes using BN port. The router microarchitecture consists (Fig.2) of DN crossbar, BN crossbar and controller.

In S-mesh architecture, messages are divided into packets in NIU, and packets are further divided into flits. Every packet is only composed of a header flit, a tail flit and some data flits. The every flit width is equal to N+2. The “2” is used as a packet label to indicate which packet belongs to idle, header, tail or payload. The link bandwidth also can be configured in 18-bit, 36-bit, 72-bit and others according to service demands. This attribute can increase data transmission efficiency and utilization of links. Meanwhile, it can overcome the effect of best-effort services using packet switching.

In 2D-mesh and Fat-tree NoC architectures, current routers do not predict any second hop router working condition. Therefore, routers need enough memory to buffer packets while the next routers might be stalled or under the condition of congesting possibility. Furthermore, local congestion is possible to cause global congestion, thus affect the whole system performance.



**Fig. 2.** S-mesh router logical implementation model.

Regarding the buffer in routers, it is obviously different in router microarchitecture comparison S-mesh and other NoC architectures. Different switching techniques are implemented in current NoC architectures except S-mesh, which has different performance metrics along with different requirements on hardware resources. Routers in S-mesh do not need to packed/unpacked and make decisions. This property makes the design of the switches simple, and the buffer is reduced or eliminated. The transmission latency of each router can therefore be reduced to a cycle. And, transmission and control of packet in S-mesh are separated. This indicates clearly that the router microarchitecture would not be changed due to routing algorithms running in CN.

In computer networks, different techniques are used to perform message switching between different nodes. Popular switching techniques include Store-And-Forward (SAF), Virtual-Cut-Through (VCT) and WormHole (WH). When these switching techniques are implemented in NoC chips, they have different performance metrics along with different requirements on memory resources. The buffer requirement in various different routing models is shown in Table 1.

**Table 1.** Buffer requirements and latency for different routing techniques.

Types	Message length	per router buffer capacity		Resource occupied	Latency
		normal	Stalling		
SAF	L	$L \times M$	$L \times M$	At two nodes	unpredictable
VCT	L	$\leq F \times M$	$L \times M$	At the local node	unpredictable
WH	L	$F \times M$	$F \times M$	At all nodes spanned by the message	unpredictable
S-mesh	L	0	0	Good running	1clk

Table 1 indicates that router in S-mesh no longer need memory to buffer message flits. In SAF model [16], an entire packet should be received and stored prior to transmission to the next router. When the message size is big enough, it not only introduces extra delay at every router stage, but also requires a substantial amount of buffer to store multiple entire packets at the same time. VCT [17] requires the buffer for an

entire packet. It is forwarded as soon as the next router guarantees that the entire packet will be accepted. However, when the next stage router is not available, the entire packet still needs to be stored in the buffers of the current router. A WH routing scheme can reduce routers memory requirements with low latency communication. If a certain flit faces a busy channel, subsequent flits have to wait at their current locations, and therefore they are spreaded over multiple routers. While packets block each other in a circular fashion such that no packets can advance, thus a deadlock is generated.

## **4 Simulation and analysis**

We evaluated the S-mesh NoC by means of a Gpnocsim [18] simulator for NoC, an architectural level cycle-accurate simulator using Java. We measured message latency from the time a packet created in origination to the time the last flit arriving at destination message center. Nevertheless, in S-mesh architecture, the start time is the time that message request is sent to CN. Every simulation initiates a warm-up phase of 2 percentage of all number of running cycles. All types of the topologies use the WH switching technique except S-mesh.

### **4.1 Message length**

In this scenario, we studied the relationships among latency and various message lengths. In this case, size of network is fixed at 16 nodes; width of flit is 64-bit.

Average message latency was increased with message length in all NoC architectures. However, the average latency of S-mesh rises slowly by comparison [13]. When more and more message flits have been injected into network, contentions for resources in routers have become more serious. As a result, time spent in data buffer and queue process would be increased quickly in 2D-mesh and others. The utilization of each router should rise sharply. Especially, fewer routers are in overload operation when the packet length is up to 96 bytes. Then local congestion would be formed as time goes by. On the other hand, when routers' output links have stalled, the routers need more memory to buffer the injecting data. Additionally, local router cannot predict following link and router running states of next hop, so congestion would be easily formed. And, the next hop router refuse the previous router ejection traffic because of resource exhausting. Further more a local congestion can quickly spread to a region or the entire network. Some NoC architectures provide a local best solution easily, but do not guarantee the global optimization performance. Because the datapath is pre-connected in S-mesh architecture, it indicates that the S-mesh would provide lower latency than other NoCs under bigger message application services.

### **4.2 Influence of buffer sizes**

Buffer is the major part of any network router. In the most NoC architectures, buffers occupy the main part of the router area. As such, it is a major concern to minimize the amount of buffer under given performance requirements. Moreover the influencing

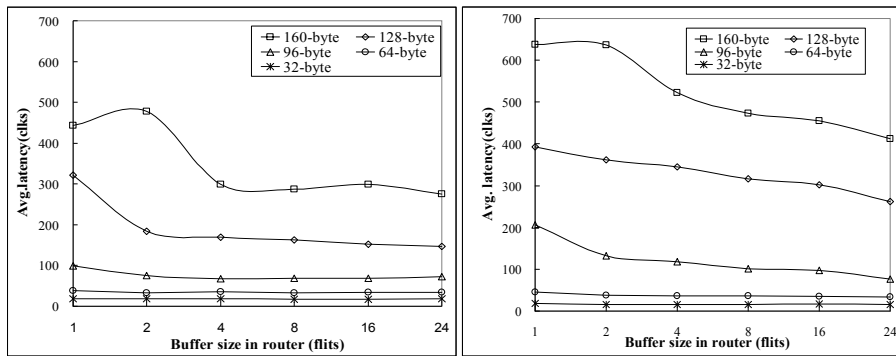
degree of buffer sizes on average network latency has also been studied. In this scenario, size of network is fixed at 16 nodes.

**Table 2.** The influence of input buffer size on network latency

Flit/Buf	Fat-tree	2D-mesh	Torus mesh	Butterfly	Octagon	S-mesh
1 -> 2	12.3%	16.0%	15.5%	25.2%	15.9%	2.1%
2 -> 4	9.9%	12.4%	9.6%	6.6%	6.9%	0.5%
4 -> 8	6.0%	7.6%	10.1%	5.9%	4.8%	0.6%
8 ->16	3.74%	-0.52%	8.18%	-4.98%	14.58%	-0.16%
16->24	3.83%	4.16%	-0.15%	-1.12%	4.32%	-1.38%

Table 2 summarizes the propagation time reduction with various buffer sizes. Average network latency on five NoC architectures can be reduced as much as from 12.3% to 25.2% while buffer size changes from 1 to 2 flit/buffer. However, with the deepening of buffer size, the adverse impact on message latency is obvious. Such as 2D-mesh, Butterfly, and S-mesh architectures, the larger buffer sizes in routers wouldn't help to promote the network performance while buffer sizes are changed from 8 flits to 16 flits.

For further research, we gave impact degree of buffer sizes with different message length in three types of NoC architectures. For the large capacity of buffer sizes, it can efficiently reduce message latency in circumstance of large message packet as shown in Fig.3. Mainly because large buffer sizes reduce the network contention. But as shown in Fig.4, S-mesh architecture can get shorter network latency no matter what message is without buffer units. It is shown that increasing the buffer size is not a solution to avoid congestion. At best, it delays the onset of congestion since the throughput is not increased. That is, buffers are useful to absorb burst traffic, thus leveling the bursts. Moreover, the performance improves marginally in relation to the power and area overhead. Unbuffered router architecture can efficiently reduce buffer requirements without reducing network latency. It is great improvement because the cost of on-chip memory is much higher.



(a) 2D-mesh (b) Fat-tree  
**Fig. 3.** Relative curve of message length and buffer sizes



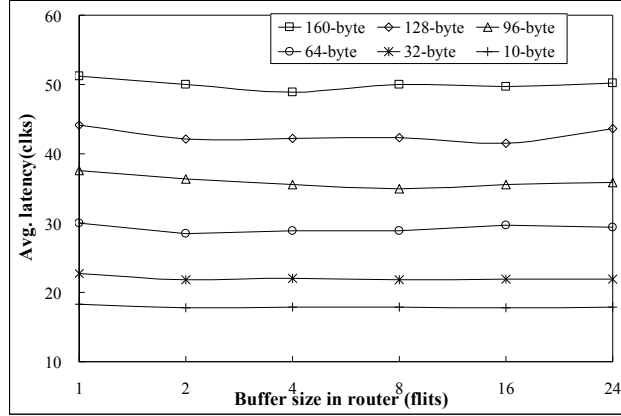


Fig.4. Relative curve of message length and buffer sizes in S-mesh architecture

### 4.3 Area and power consumption results

Area figures were achieved with Design Compiler logic synthesis tool targeting a Chartered 0.13 $\mu$ m CMOS High Performance (HP) technology. Power consumption was estimated by using the same tool that performs cycle-based simulations on the synthesized netlist.

The comparison of the alternative NoC architectures for area and power consumption are summarized in Table 3. It should be noted that the area of router of S-mesh is smaller than the area of other architectures except routers in 3D mesh. Because of using Chartered HP technology, the power dissipation of the router is slightly higher. Though the same traffic is transmitted, it's along with the optimal distances in S-mesh without more latency and no power is wasted on data buffer and congestion. These factors are all perfectly processed by Control Network. The operating frequency of Router in S-mesh is up to 1250 MHz, and achieves 200Gbps maximum bandwidth.

Table 3. Area and power consumption results

	Router	Buffer	Equivalent area (mm <sup>2</sup> )	Power (mw)
1	3D mesh[19]	80 flits	0.0346	9.41@500M
2	<b>S-mesh:2009</b>	--	<b>0.0411</b>	<b>7.39@200M</b>
3	Reconfigurable [20]	Circuit-	0.051	n.a.
4	MaRS[21]	32-flits	0.052	4.47@432M
5	ReNoC [22]	40 flits	0.061	n.a.
6	Aethereal[23]	24-word	0.175	n.a.
7	Xpipes [24]	64b flits	0.19	n.a.
8	GT-BE[25]	8 flits	0.26	n.a.
9	QNoC[26]	input buffer	0.314	n.a.
10	GALS [27]	n.a.	0.884	n.a.

## 5 Conclusion

S-mesh NoC architecture borrows excellent architectural features of packet switching network, and unbuffered router microarchitecture borrows excellent architectural characteristic of circuit switching network. Therefore, S-mesh can gain higher performance than 2D-mesh and other NoC architectures under the condition of long message packet. The Equivalent area of router architecture is only  $0.0411\text{mm}^2$ . And local network performances optimizing would interfere in overall network performances. The result shows that S-mesh architecture and unbuffered router architecture are feasible and effective.

**Acknowledgments.** We are grateful to thank the anonymous reviewers for their useful comments and suggestions. This research is supported by the High Technology Research and Development Program of China (No. 2009AA01Z105), the Postdoctoral Science Foundation of China under Grant (No.20080440942 and 200902432), and the Ministry of Education–Intel Special Foundation for Information Technology (No. MOE-INTEL-08-05).

## References

1. Dally, W.J., Towles, B.: Route Packets, Not Wires: On-Chip Interconnection Networks. Design Automation Conf., USA (2001)683-689
2. Jantsch, A., Tenhunen, H.: Networks on Chip. Kluwer Academic Publishers, Hingham, USA(2003)3-39
3. Pande, P. P., Grecu, C., Ivanov, A., Saleh, R. : Design of a Switch for Network on Chip applications. ISCAS, Bangkok, Thailand (2003)5:v217-v220
4. Kumar, S., Jantsch, A., Soininen, J.-P., Forsell, M., Millberg, M., Tiensyrja, K.: A network on chip architecture and design methodology. VLSI, Proceedings of IEEE computer society annual symposium on, Pittsburgh, USA (2002)105-112
5. Karim, F., Nguyen, A., Dey, S.: An interconnect architecture for networking System on Chips, Micro, IEEE (2002)22(5): 36-45
6. Bononi, L., Concer, N.: Simulation and Analysis of Network on Chip Architectures: Ring, Spidergon and 2D Mesh. Design, Automation and Test in Europe, Munich, Germany (2006)2:6-10
7. Millberg, M., Nilsson, E., Thid, R., Kumar, S., Jantsch.: The Nostrum backbone - a communication protocol stack for networks on chip, VLSI Design, Proceedings. 17th International Conference on, Mumbai, India (2004)693-696
8. Rijpkema, E., Goossens, K.: A router architecture for networks on silicon. Progress 2001, 2nd Workshop on Embedded Systems (2001)
9. Taylor, M.B., Kim, J., Miller, J.: The Raw microprocessor: a computational fabric for software circuits and general-purpose programs. Micro, IEEE (2002)25(2):25-35
10. Dall'Osso, M., Biccari, G., Giovannini, L., Benini, L.: Xpipes: a latency insensitive parameterized network-on-chip architecture for multi-processor SoCs. Proc. ICCD, SanJose, USA (2003)536-539
11. Liuhaio, Zou Xuecheng, Ji Lixin, Cai Meng, Zhang Kefeng.:S-mesh: A mesh-based on-chip network with separation of control and transmission. The journal of China universities of posts and telecommunications (2009)16(5):86-92,102

12. ITU-T.: Network element management information model for the Message Transfer Part (MTP) . ITU-T, Rec. Q.751.1. International Telecommunication Union - Telecommunication Standardization Sector, Geneva (1995)
13. ITU-T 2001b.: Signalling connection control part procedures. ITU-T, Rec. Q.714. International Telecommunication Union - Telecommunication Standardization Sector, Geneva (2001)
14. McKeown N.: Fast Switched Backplane for a Gigabit Switched Router[online]. Available from: [http://tiny-tera.stanford.edu/~nickm/papers/cisco\\_fasts\\_wp.pdf](http://tiny-tera.stanford.edu/~nickm/papers/cisco_fasts_wp.pdf) [Accessed 03/23/07]. Stanford University, Stanford, USA (2008)
15. Yuan, T., Fan, X.Y., Jing, L.: Application specific network on-chip architecture. *Computer Engineering and Applications*, (2007) 43(6):88-91(in Chinese)
16. Terry, T.Y.: On-chip multiprocessor communication network design and analysis. PhD thesis, Stanford University, USA (2003)
17. Benini, L., Bertozzi, D.: Network-on-chip architectures and design methods. *Computers and Digital Techniques*, *IEEE Proc*, (2005)152(6):261-272
18. Hossain, H., Ahmed, M., Al-Nayeem, A., Islam, T.Z., Akbar, M.: GPNOC SIM – A General Purpose Simulator for Network-on-Chip, *ICICT*, Dhaka , Bangladesh (2007)254-257
19. Kim J, Nicopoulos C, Park D, Reetuparna Das R, Xie Y, Vijaykrishnan N, Mazin S. Chita R.: A Novel Dimensionally-Decomposed Router for On-Chip Communication in 3D Architectures. In: 34th International Symposium on Computer Architecture (ISCA2007). San Diego, California, USA (2007)138-149
20. Wolkotte, P T, Gerard J.M. Smit J.M.G, Rauwerda G K, Smit L T.: An energy-efficient reconfigurable circuit-switched NOC. In: *Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium(IPDPS 2005)*. Denver, Colorado, USA (2005)155a-163a
21. Bahn J H, Lee S E, Bagherzadeh N.: Design of a router for network-on-chip. *International Journal of High Performances Systems Architecture* (2007) 1(2): 98-105
22. Stensgaard M B, Sparso J.: ReNoC: A Network-on-Chip Architecture with Reconfigurable Topology. 2nd ACM/IEEE International Symposium on Networks-on-Chip. Newcastle University, UK (2008) 55~64
23. Dielissen J, Rădulescu A, Goossens K, Rijpkema E.: Concepts and implementation of the Philips network-on-chip. In: *IP Based SoC Design 2003*.Grenoble, France (2003)
24. Benini L, Bertozzi D.: Network-on-chip architectures and design methods. *IEE Proceedings of Computers and Digital Techniques* (2005) 152(2):261-272
25. Rijpkema E, Goossens K G W, Rădulescu A, Dielissen J, Meerbergen J, Wielage P, and Waterlander E.: Trade Offs in the Design of a Router with Both Guaranteed and Best-Effort Services for Networks on Chip. *IEE Proceedings of Computers and Digital Techniques* (2003) 150(5): 294-302
26. Bolotin E, Cidon I, Ginosar R, Kolodny A.: QNoC - QoS architecture and design process for Network on Chip. *Journal of system architecture* (2004) 50: 105-128
27. Zipf P, Hinkelmann H, Ashraf A, Glesner M.: A Switch Architecture and Signal Synchronization for GALS System-on-Chips. In: *17th Symposium on Integrated Circuits and Systems Design (SBCCI 2004)*. Pernambuco, Brazil (2004) 210-215