Bulletin of Electrical Engineering and Informatics
Vol. 8, No. 2, June 2019, pp. 438~442
ISSN: 2302-9285, DOI: 10.11591/eei.v8i2.1422

A efficacy of different buffer size on latency of network on chip (NoC)

Farah Wahida Binti Zulkefli, P. Ehkan, M. N. M. Warip, Ng. Yen. Phing
School of Computer and Communication Engineering, Universiti Malaysia Perlis,
02060 Arau, Perlis, Malaysia

Article Info

Article history:

Received Dec 28, 2018
Revised Feb 15, 2019
Accepted Marh 2, 2019

Keywords:

Buffer
Latency
Network-on-Chip
Router

ABSTRACT

Moore's prediction has been used to set targets for research and development in semiconductor industry for years now. A burgeoning number of processing cores on a chip demand competent and scalable communication architecture such as network-on-chip (NoC). NoC technology applies networking theory and methods to on-chip communication and brings noteworthy improvements over conventional bus and crossbar interconnections. Calculated performances such as latency, throughput, and bandwidth are characterized at design time to assured the performance of NoC. However, if communication pattern or parameters set like buffer size need to be altered, there might result in large area and power consumption or increased latency. Routers with large input buffers improve the efficiency of NoC communication while routers with small buffers reduce power consumption but result in high latency. This paper intention is to validate that size of buffer exert influence to NoC performance in several different network topologies. It is concluded that the way in which routers are interrelated or arranged affect NoC's performance (latency) where different buffer sizes were adapted. That is why buffering requirements for different routers may vary based on their location in the network and the tasks assigned to them.

Corresponding Author:

Farah Wahida Binti Zulkefli,
School of Computer and Communication Engineering,
Universiti Malaysia Perlis,
02060 Arau, Perlis, Malaysia.
Email: fwahida06@gmail.com

1. INTRODUCTION

The driving force behind Integrated Circuit (IC) technology has been Moore's law for almost five decades. Moore predicted that the number of transistors per square inch on integrated circuit will be doubled every year since the integrated circuit first invention in 1970s. Based on Moore's prediction, the future integrated systems will contain billions of transistors with hundreds of IP core to undergo complex multimedia delivery and networks services. In 1990s, System-on-Chip (SoC) has been introduced where many components such as microprocessor, custom IP and analog integrated in a single chip. As SoC complexity elevates, it is difficult to encapsulate the system's functionality with fully deterministic operation.

It is expected that interconnection technology has become a limiting factor in future SoC designs. A possible approach for coping with this problem is to use an on-chip interconnection network instead of ad-hoc global wiring. In order to extend the relevancy of Moore's law, network-on-chip (NoC) architectures has been proposed replacing shared bus in SoC. NoC technology is often called "a front-end solution to a back-end problem" [1]. These days, many NoC prototypes have been designed and analyzed by the educational community. Most of them are focusing on different aspects of the communication infrastructure such as the quality-of-service achievement, synchronization method of the routers, decreasing power

consumption, and application mapping process [2]. In this paper, the focus is on the relationship between different buffer sizes in virtual channel router that affect NoC performance in several NoC topology.

2. WHAT IS NOC?

NoC technology is already being endorsed in the majority of large SoCs for intelligible integration in the system at the IP-assembly functional verification level. NoC architecture is comprised of three main building blocks. As seen in Figure 1, the first and utmost essential block is the links that physically connect the nodes and eventually handle the communication. The other block is the router. Router is responsible for communication protocol in NoC architecture (the decentralized logic behind the communication protocol). Another building block is the network adapter (NA) or network interface (NI). The logic connection between the IP cores and the network are prepared by NI, considering each IP is allowed to have an exclusive interface protocol with respect to the network.

NoC can be perceived as an evolvement of the segmented busses where the router acts as a "much smarter buffer" [3]. Router in NoC incipiently receives packets from the shared links and then forward the packets according to the address acquainted in each packet to the core attached to it or to another shared link. The protocol alone has its place of a set of policies construed during the design to deal with common situations during the transmission of a packet, such as two or more packets arriving at the same time or disputing the same channel, avoiding deadlock and livelock situations, reducing the communication latency, and increasing the throughput.

Router plays a significant role in NoC. Figure 2 shows a typical NoC router architecture. NoC router commonly consists of a controller, routing units, crossbar switch and ports of input and output. The controller includes a switch allocator and virtual channel allocator. Virtual channel allocator usually used when there are virtual channels in input port. Each virtual channel in input port has their own buffer and output ports connected directly to the outgoing links [4]. The terms router and switch are frequently used as one and the same, but the term switch can also address the internal switch matrix that actually connects the router inputs to its outputs. Besides, NoC router also contains a logic block that implements the flow control policies (routing, arbiter, etc.) which defines the overall strategy for moving data through the NoC.

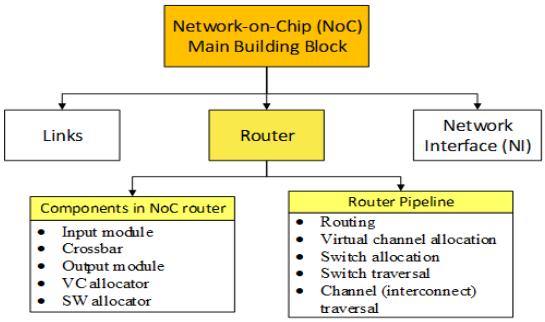

Figure 1. NoC main building block

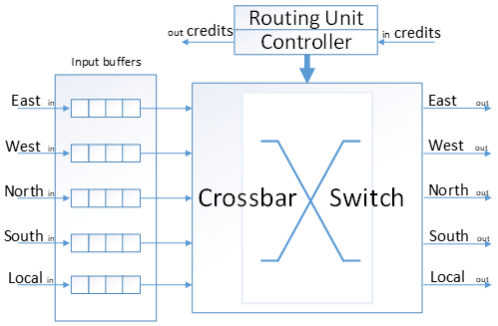

Figure 2. Basic router in NoC

3. VIRTUAL CHANNEL AND BUFFERING IN NOC

3.1. Virtual channel router

Virtual channel (VC) is a means to transport data packets over a network as if there are dedicated physical transport between the source and destination. The main goal for virtual channel is to reduce congestion when two or more flows compete for the same path in the network [5]. A virtual channel splits a single channel into multiple channels to provide two different roads for routing packets process [6].

The block diagram of a typical virtual channel router is shown in Figure 3. This router can be described based on two functionalities: the datapath and control plane [7]. The datapath is made of the input and output units, and a switch that connect the input unit to the output unit. The main function of these modules is to perform the allocation of packets. For each packet, router will assign output port while virtual-channel allocator will locate output virtual channel. The control plane, on the other hand, performs route computation, virtual-channel allocation and switch allocation. A control plane is very important in coordinating the packet's movement through datapath resources.

The input and output units of the router consist of input control state with buffers. For each input unit, five state fields have been used to trace each virtual channel status. There are global state (G), route (R),

output VC (O), pointers (P) and credit count (C). Similarly, output virtual channel state fields are represented by three vectors: global state (G), input VC (I) and credit count (C). Route computation is the first step to deliver a packet through the router. Each flit of the packet will be forwarded over a virtual channel once a route is fixed and a virtual channel is allocated. The flits are forwarded to the relevant output unit using switch allocator by allocating a time slot on the switch and output channel. Later on, those flits will be forwarded to the next router stated in the packet's predestined path. Route computation and virtual-channel allocation are performed once per packet [7]. Contrary, switch allocation is performed by per-flit basis. For this reason, R, O and I field states are updated once per packet while P and C are updated at flit's frequency.

3.2. Example of VC operation

Virtual channels depend upon the inclusion of buffers, separately for each VC at the receiver's side. At the same time, it is essential calling for enhancements to the flow control signaling to harbor the multiple and independent flows travelling in each VC.

Figure 4 presents an example of a three VC processing data transfer. Flits from only one VC can be sent for one clock cycle even though several VCs are active at the sender; only one valid (i) signal is avouch per cycle. In order to pick the flit that will pass through the link, some form or arbitration will be conducted to choose one VC from those that contain valid flits. At the meantime, the receiver may be ready to obtain the flits which possibly belong to any VC. There is no limitation on how frequent ready (j) signals can be asserted per cycle. Both the buffering resources and the flow-control handshake wires have been manifolded with the number of VCs in VC flow control.

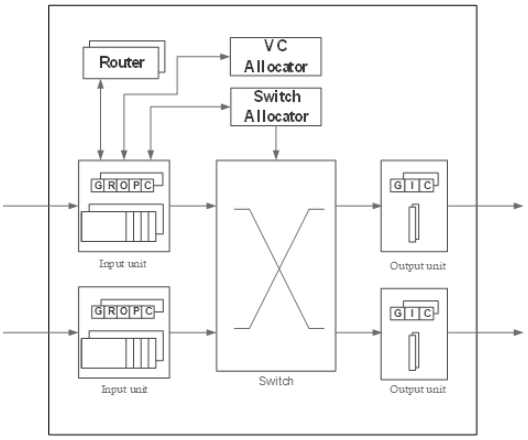

Figure 3. Virtual channel router general structure [8]

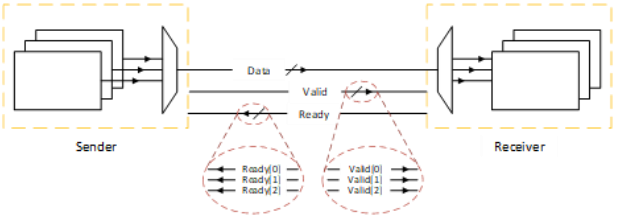

Figure 4. Example of three VC operations

3.3. Buffers in NoC router

In communication infrastructure, the router's buffer space minimization and simplified buffer control mechanisms are two important features of the NoC design, as they directly affect the overall area-power overheads and network latency [8]. Traditional virtual channel-based routers use buffers for deadlock freedom and performance optimization. While total storage is optimized for performance, actual buffer occupancies can be very low.

Buffer size and allocation policy play an important role in the performance and efficiency of a NoC router [9-11]. Furthermore, studies have shown that buffers can consume as much as up to 79% of NoC router power [8]. Thus, efficient management is required to ensure high performance and low power. Traffic in NoCs is not uniformly distributed [12]. Some nodes play a bigger role in generating and consuming traffic, while others have a smaller part. That is why buffering requirements for different routers may vary based on their location in the network and the tasks assigned to them. Besides, such requirements are subject to change during different phases of a single application.

4. EXPERIMENTAL RESULTS

In this experiment, VC routers with different size of buffers were evaluated to understand the effect of the buffer size in different type of NoC topologies. For the evaluation in this paper, Booksim 2.0 simulator

is being used. The performances are calculated in terms of latency. Latency is the time required for a packet to pass through the network from source node to destination node [7]. The calculations for latency have mainly focused on the zero-load latency of the network. Latency which is due to contention with other packets over shared resources often times being ignored. Once contention latency is included through modeling or simulation, latency becomes a function of offered traffic [13, 15].

$$Packet\ Latency = Actual\ Transmit\ time + Routing\ Delay + Contention\ Delay \quad (1)$$

The fixed variable in this experiment is the number of VC used by the router, which is four VC. VC behaves similar to having multiple wormhole channels present in parallel. However, adding extra VC to each link does not add bandwidth to the physical channel [14]. It just enables better sharing of the physical channel by different flows. For the buffer sizes, several sizes of buffer used are 4, 8, 16 and 32. Topologies arranged in this experiment are topologies that are widely used in the study of NoC which include torus topology, mesh topology, flattened-butterfly topology and fat-tree topology. These topologies are built-in topologies generated by Booksim 2.0 simulator. Table 1 shows differences from different topologies in term of latency.

As shown in Figure 5 and Figure 7, the latency in topology mesh and fat-tree slightly increases as the buffer size escalate. Meanwhile, latency in torus and flattened-butterfly topology from Figure 6 and Figure 8 appeared to be considerably declining. The result specified that different buffer size and varied topology may affect the performance of NoC even by a bit. Using 4-VC router, mesh topology proves to have the highest latency compared to the other topologies. Followed by torus, flattened-butterfly and fat-tree consecutively.

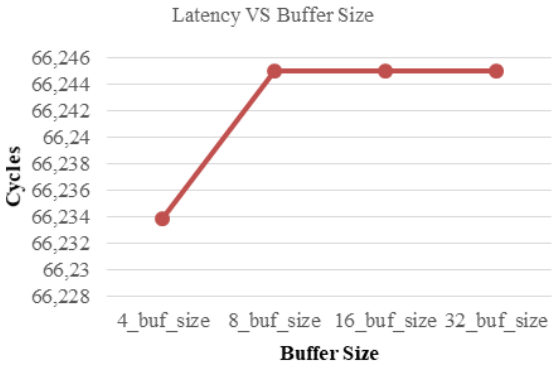

Figure 5. Mesh topology

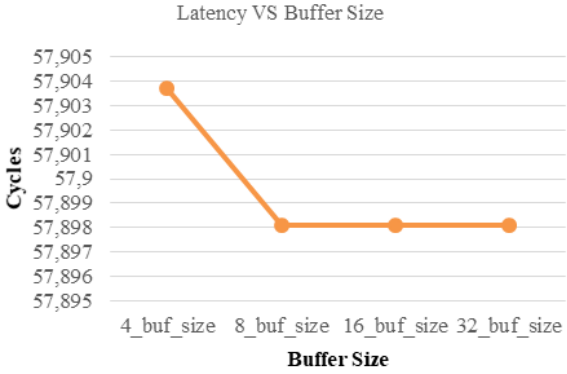

Figure 6. Torus topology

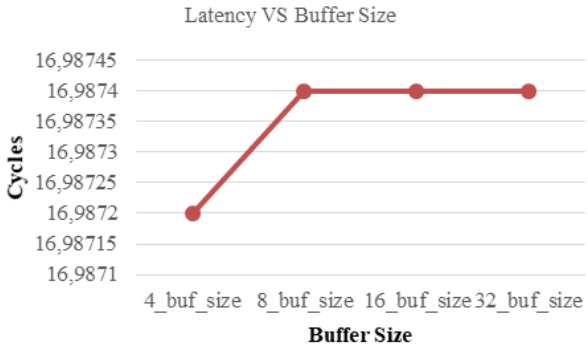

Figure 7. Fat-tree topology

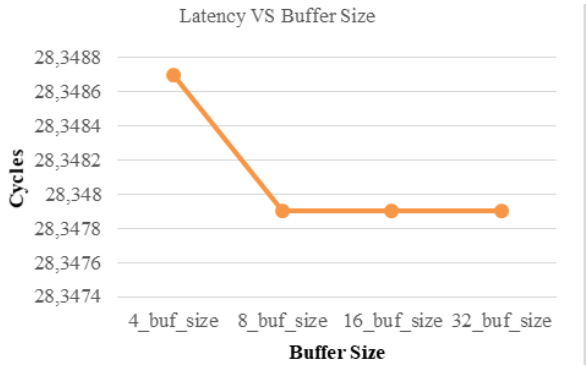

Figure 8. Flattened-butterfly topology

Table 1. Differences from different topologies in term of latency

Buffer Size	Torus	Mesh	Flattened-butterfly	Fat-tree
4	57.9037	66.2339	28.3487	16.9872
8,16,32	57.8981	66.245	28.3479	16.9874
Differences	0.0056	-0.0111	0.0008	-0.0002

5. CONCLUSION

As shown in Table 1, buffer size for NoC's router differ only starting from buffer size of eight and above (8, 16, and 32) where for some topologies, latency decrease while increase in another topology. Based on this experiment, torus and flattened-butterfly topologies have an increase of latency with value of 0.0056 and 0.0008 (cycles) consecutively while latency in mesh and fat-tree topologies decrease 0.0111 and 0.0002 (cycles) successively. We can conclude that the way in which routers are interrelated or arranged affect NoC's performance (latency) where different buffer sizes were adapted. That is why buffering requirements for different routers may vary based on their location in the network and the tasks assigned to them.

Networks-on-chip (NoC) are emerging as a viable interconnects architecture for multiprocessor SoC platforms. Even though a lot of NoCs improvement has been proposed, only few have been implemented on silicon. This shows that there are many more challenge to be dealt with from physical level up to the network layer and its system architecture. For future work, it's possible to run this experiment with other topologies such as folded-torus, octagon or any other hybrid topologies.

REFERENCES

[1] K. Tatas, K. Siozios, D. Soudris, and A. Jantsch, "Designing 2D and 3D Network-on-Chip Architectures." New York, NY: Springer New York, 2014.
[2] E. Carara, N. Calazans, and F. Moraes, "*Router Architecture for High-Performance NoCs*," ACM Proceedings, pp. 111–116, 2007.
[3] É. Cota, A. de Morais Amory, and M. Soares Lubaszewski, "Reliability, Availability and Serviceability of Networks-on-Chip," Boston, MA: Springer US, 2012.
[4] A. M. Method, "A Literature Review of on-Chip Network Design using an Agent-based Management Method," *International Journal of Engineering Research & Technology (IJERT).* vol. 3, no. 12, pp. 66–69, 2014.
[5] O. Ghorse, N. Meena, and S. Singh, "Design of Efficient Virtual Channel Router for Network-On-Chip," *International Journal of Engineering Research & Technology (IJERT).* vol. 2, no. 12, pp. 1800–1804, 2013.
[6] A. Agarwal, B. Raton, C. Iskander, H. Multisystems, and R. Shankar, "Survey of Network on Chip (NoC) Architectures & Contributions," *Networks*, vol. 3, no. 1, p. 15, 2009.
[7] R. Bott, "Principles and Practices of Interconnection Networks", no. 1. 2014.
[8] W. C. Tsai, Y. C. Lan, Y. H. Hu, and S. J. Chen, "Networks on Chips: Structure and design methodologies," *J. Electr. Comput. Eng.*, vol. 2012, pp. 1-15, 2012.
[9] R. Marculescu, J. Hu and U. Y. Ogras, "Key research problems in NoC design: a holistic perspective," *2005 Third IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS'05)*, Jersey City, NJ, USA, 2005, pp. 69-74.
[10] S. Qi, M. Zhang, J. Li, T. Zhao, C. Zhang and S. Li, "A high performance router with dynamic buffer allocation for on-chip interconnect networks," *2010 IEEE International Conference on Computer Design*, Amsterdam, 2010, pp. 462-467.
[11] S. K. Mandal, R. Denton, Saraju P. Mohanty, and R. N. Mahapatra, "Low Power Nanoscale Buffer Management for Network on Chip Routers," *Proceedings of the 20th ACM Great Lakes Symposium on VLSI 2009, Providence, Rhode Island, USA,* May 16-18 2010 pp. 245–250.
[12] R. Bashizade and H. Sarbazi-Azad, "Traffic-aware buffer reconfiguration in on-chip networks," *2015 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, Daejeon, 2015, pp. 201-206.
[13] S. Tyagi, P. Maheshwari, A. Agarwal and V. Avasthi, "Exploring 3D Network-on-Chip Architectures and Challenges," *2017 International Conference on Computer and Applications (ICCA)*, Doha, 2017, pp. 97-101.
[14] N. Y. Phing, M. N. Mohd Warip, P. Ehkan, F. W. Zulkefli, and R. B. Ahmad, "Performance Analysis of the Impact of Design Parameters to Network-on-Chip (NoC) Architecture," *Proceedings of the 2nd. International Conference of Reliable Information and Communication Technology (IRICT 2017), Eds. Cham: Springer International Publishing*, pp. 237–246, 2017.
[15] N. Jain, and M. Patel, "A Review of the Design Challenges for the 3-D on chips Network Paradigms", *International Journal of Computer Application*, pp. 11-15, 2017