An Efficient Virtual Channel Router for NoC's

Minakshi M. Wanjari
Assistant Professor,
Electronics and Telecomm. Engineering Dept.,
RCOEM, Nagpur, India.
meena.wanjari@gmail.com

Dr. Pankaj Agrawal
Associate Professor,
Electronics and Communication Engg. Dept.,
RCOEM, Nagpur, India.
pankaj_rknec@rediffmail.com

Dr. R. V. Kshirsagar
Professor,
Electronics Engineering Dept.,
PCE, Nagpur, India.
ravi_kshirsagar@yahoo.com

Abstract—**Network-on-Chip (NoC) introduces the design methodology of interconnection network into System-on-Chip. It overcomes the main disadvantages of traditional bus-based SoC, for example, large delay, small link bandwidth and poor scalability, etc. It is widely believed that Network-on-Chip will replace bus-based architecture to become the mainstream of SoC design methodology. Router is the main component of NoC and determines performance to a large extend. Virtual-channel router is one of the promising router architectures for Network-on-Chip, for its low delay and high network throughput. This paper will describe the parameters affecting the NoC performance and the remedial solution to overcome these. The focus of this paper is to optimize the design of arbitration units and to increase the buffer utilization so as to increase the performance of virtual-channel router.**

Keywords-Network-on-Chip (NoC); Virtual channel (VC); Virtual channel arbitration (VA); Switch allocation (SA)

I. INTRODUCTION

As the feature size is continuously decreasing and integration density is increasing, interconnections have become a dominating factor in determining the overall quality of a chip. Due to the limited scalability of system bus, it cannot meet the requirement of current System-on-Chip (SoC) implementations where only a limited number of functional units can be supported. Long global wires also cause many design problems, such as routing congestion, noise coupling, and difficult timing closure. Network-on-Chip (NoC) architectures have been proposed to be an alternative to solve the above problems by using a packet-based communication network [1, 2, 3].

A generic NoC implementation consists of a number of Processing Elements (PE) arranged in a mesh-like grid, as shown in Figure 1. The PEs may be of the same type, e.g. CPU, or of different type, e.g. audio cores, video cores, wireless transceivers, memory banks etc. Each PE is connected to a local router through a Network Interface Controller (NIC); each router is, in turn, connected to adjacent routers forming a packet-based on-chip network. The NIC module packetizes/de-packetizes the data into/from the underlying interconnection network. The PE together with its NIC forms a network node. Nodes communicate with each other by injecting data packets into the network. The packets traverse the network toward their destination, based on various routing algorithms and control flow mechanisms.

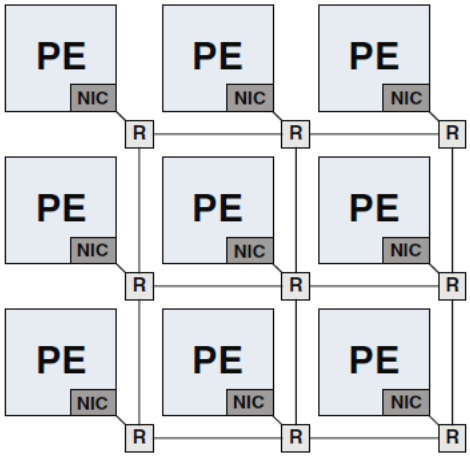

Figure 1. Generic NoC architecture.

The heart of an on-chip network is the router, which undertakes the crucial task of steering and coordinating the data flow. Performance of Network-on-chip is determined by the router architecture to a large extend and virtual-channel router is said to be a promising choice for NoC [4].

The rest of this paper is organized as follows. Section II will present an overview of virtual-channel routers for Network-on-Chip; section III will introduce generic virtual-channel router architecture, section IV will describe the parameters affecting the NoC performance and the remedial solution to overcome these by doing architectural modifications and in the last is conclusion.

II. RELATED WORK

For flow control, switching techniques are mechanisms by which information is forwarded through the NoC network. Switching techniques have a significant influence on the design of router micro-architecture, and are broadly classified into circuit switching and packet switching, based on the

ISBN: 978-81-923777-9-7

network characteristics. Packet switching techniques are the most commonly used in current NoC designs [5].

Packet switching is further classified as Store and Forward (SAF), Wormhole (WH) and Virtual Cut Through (VCT) switching. SAF switching requires large buffer size and increased latency in the router. In the VCT switching mechanism, the buffer requirements are reduced compared to the SAF switching. WH switching techniques are prone to deadlock when cyclic buffer dependencies develop from the topology and routing algorithm of the network. However, all switching mechanisms are prone to the Head-on-Line (HoL) blocking problem, which results from input buffering contention in destination routers.

To overcome the above problems in router switching techniques, researchers have proposed various buffering allocation techniques (static and dynamic), micro architectural buffer structures, and efficient buffer usage (arbitration) algorithms. The most significant improvement to WH switching is the introduction of virtual channels (VCs) [6]. J. Dally introduced the idea of the virtual channel to develop deadlock-free routing algorithms for networks that use WH routing [7]. Earlier buffer allocation techniques proposed by various researches include: speculative allocation [8]; traffic aware VC allocation [9]; advance reservation control of resources [10]; buffer size allocation, based on channel utilization [11]; and implementing VCs, using asynchronous circuit design [12].

Dally and Towels illustrate the basic virtual-channel router architecture in interconnection networks [13]. They point out that virtual-channel router works in a pipelined fashion. In order to decrease router delay speculative arbitration (SPA) and look-ahead routing (LAR) mechanisms are proposed for basic virtual-channel router. Using these mechanisms, a single flit can travel through VC router within only one cycle. This low latency virtual channel router is introduced to NoC in [14]. In [15], authors proposed a low latency router which utilizes adaptive routing.

Virtual channel router can also be used to solve network deadlock when adopting adaptive routing [16], and simultaneously provide guaranteed service and best-effort service [17].

Sometimes all these performances are achieved at the cost of increasing router complexity and buffer requirement. Because Network-on-Chip is strict resource constrained, a good virtual-channel router should make a better tradeoff between performance and implementation cost.

III. VIRTUAL CHANNEL ROUTER

A. *Generic router architecture*

In general, the router has P input and P output channels (or ports). In most implementations, P = 5; four inputs from the four cardinal directions (North, East, South and West) and one from the local Processing Element (PE), which is attached to the NoC router. To minimize router complexity and traffic congestion, NoC routers are usually assumed to connect to a single PE. The input/output channels may consist of unidirectional links (as shown in Figure 2), bidirectional, or even serial links. Buffering within a network router is necessary due to congestion, output link contention, and intra-router processing delays (e.g. routing computation), which impede data flow. In the case of virtual channel-based NoC routers, each input port consists of a number of FIFO buffers, with each FIFO corresponding to a virtual channel (see Figure 2). Hence, each input port has v virtual channels, each of which has a dedicated k-flit FIFO buffer (a flit is the smallest unit of flow control; one network packet is composed of a number of flits) [10].

The router control logic forms the heart of the NoC router, and comprises four components: (a) the Routing Computation (RC) unit, (b) the Virtual Channel Arbitration (VA) logic, (c) the Switch Allocation (SA) logic, and (d) the Crossbar (XBAR).

1) Routing Computation (RC) Unit: The RC unit is responsible for directing the header flit of an incoming packet to the appropriate output Physical Channel (PC) and/or dictating valid output Virtual Channels (VC) within the selected PC. Output VCs are, essentially, the input VCs of the adjacent routers. The routing is done based on the packet's destination address, which is present in the header flit, and can be deterministic (e.g. dimension-order routing) or adaptive (e.g. load-balancing). RC is a "per-packet" operation; it is performed once for each packet within a router (i.e., only on the header flit of each packet).

2) Virtual channel Allocation: The Virtual Channel Arbitration (VA) module arbitrates amongst all packets requesting access to the same VCs and decides on winners. Since the routing function may not specify a particular output VC in the requested output physical port two arbitration stages are generally required, as shown in Figure 3.

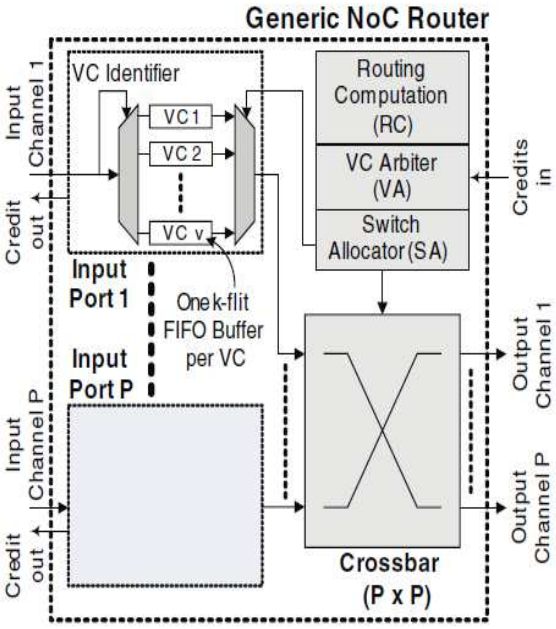

Figure 2. Generic NoC router architecture.

ISBN: 978-81-923777-9-7

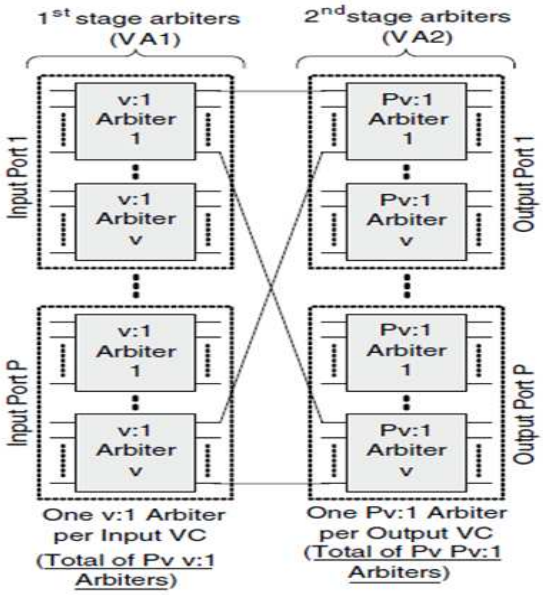

Figure 3. NoC router arbitration logic using VC arbitration (VA).

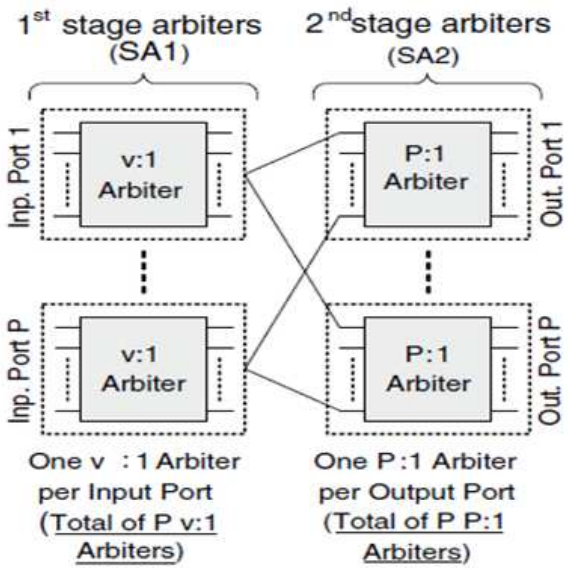

Figure 4. NoC router arbitration logic using Switch allocation (SA).

The first stage (VA1) reduces the number of requests from each input VC to one; this ensures the request of a single VC at a particular output port by each input VC. A total of Pv v:1 arbiters are, therefore, required in the first arbitration stage (see Figure 3.) one arbiter for each input VC. Subsequently, the winning request from each input VC proceeds to the second arbitration stage (VA2). A total of Pv Pv:1 arbiters are required in this stage: one arbiter for each output VC (each arbiter is of Pv:1 size to accommodate the worst-case scenario in which all input VCs request the same output VC). Just like RC, VA is also a "per-packet" operation; it is only performed on header flits [10].

3) Switch Allocation: The SA unit arbitrates amongst all VCs requesting access to the crossbar and grants permission to the winning flits. Switch allocation is also performed in two stages. The first stage called local stage (SA1) accounts for the sharing of a single port by a number of VCs; all VCs within the same input port compete against each other locally for access to the output physical channels. The second stage called global stage (SA2) arbitrates between the winning requests from each input port for each output port. The SA2 stage sets the crossbar control signals accordingly. The SA unit is very similar in structure to the VA unit. It consists of logically identical arbiters arranged in cascaded fashion. The difference lies only in the size and number of arbiters used: P v:1 arbiters are required for SA1 (i.e. one arbiter per input port), and P P:1 arbiters are required for SA2 (i.e. one arbiter per output port), as shown in Figure. 4. As opposed to the VA's "per-packet" operation, SA is a "per-flit" operation, i.e. it is performed on all flits traversing the router, not just header flits. The SA winners are then able to traverse the crossbar and are placed on the respective output links [10].

4) Crossbar: Flits that have been granted passage on the crossbar are passed to the appropriate output channel. In the router architecture illustrated in Figure 2., each input port is forced to share a single crossbar port even when multiple flits could be sent from different virtual-channel buffers. This restriction allows the crossbar size to be kept small and independent of the number of virtual-channels [8].

B. VC Router Pipeline Structure

According to the router architecture, virtual-channel router works in a pipelined fashion. All the packets have to experience four pipeline stages. They are routing computation, virtual-channel allocation, switch allocation and switch traversal. In virtual channel flow control, packets are divided into smaller flow control units called flits. There are three kinds of flits: head flit, body flit and tail flit. Because only head flit carries routing information of the packet, routing computation and virtual-channel allocation are merely carried out to head flit. Once RC and VA are done, there is a data path reserved for the packet in the virtual-channel router. Body flits and tail flit are transmitted in the reserved path following the head flit. Switch allocation and switch traversal are two stages all the flits should undergo in the same manner. Each flit has to send a request to switch allocator. Only if the request is granted, the flit could transfer through the crossbar in the following cycle. After the tail flit is ejected from virtual-channel router, reserved path is then released. Figure 5. illustrates the pipeline structure of virtual channel router [19].

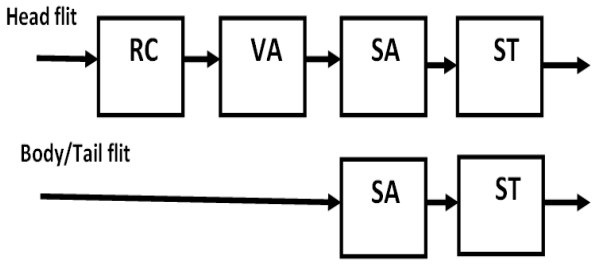

Figure 5. VC router pipeline structure.

ISBN: 978-81-923777-9-7

IV. DYNAMIC AND EFFICIENT VC ROUTER

In NoC, a router sends packets from a source to a destination router through several intermediate nodes. If the head of packet is blocked during data transmission, the router cannot transfer the packet any more. In order to remove the blocking problem, the researcher proposed wormhole routing method. The wormhole router splits the packet into several flits which can be transferred in a single transmission.

Buffers consume much leakage power since buffers, which are implemented with registers, occupy large areas compared to other combinational logics [20]. The buffers consume about 64 percent of the total router leakage power. Dynamic power consumption is proportional to switching activity, supply voltage, and capacitance load. Whenever the flit arrives at or departs from router, it consumes much dynamic power depending on switch activity. Therefore, buffer design plays an important role in implementing an energy efficient on-chip network [10].

The occurrence of malignant double-grants and the impact of no-grants and priority violation in the VA and SA units are of critical importance, since they lead to flit/packet losses, performance degradation. So, our aim is to modify the generic router by using three architectural modifications which will make it more efficient and dynamic than a conventional router.

The three architectural modifications are

A. A Dynamically allocated buffer structure

Dynamic buffer allocation scheme will increase the buffer utilization using dynamic virtual channels and buffer slots in real-time depending on traffic conditions and will decrease the overall power consumption in order to improve the network performance. Dynamic allocation of buffer slots will reduce the HoL blocking. This will achieve much higher throughput, will save in chip area and will reduce the power consumption.

B. A new virtual channel allocation scheme, called VA Reduction

VA reduction will eliminate the first arbitration stage altogether to ensure synchronized arrival times at the inputs of the second-stage (global) arbiters. Furthermore, it will reduce the number of second-stage arbiters from Pv to P, without affecting performance. VA reduction will provide full immunity from packet losses afflicting generic VA units and will offer a significantly better performance while being much more area and power-efficient than a conventional VA module.

C. A new interpretation of the switch allocation process, called Parallel SA

By parallelizing the two arbitration stages, the parallel SA mechanism will protect the switch allocation process from malignant double-grants and flit losses. The performance will improve because of (i) a more efficient arbitration scheme, and (ii) the halving of the SA critical path delay, which will allow for a much higher operating clock frequency.

V. CONCLUSION

Virtual-channel router is one of the promising router architectures for Network-on-Chip for its high throughput and low delay. However, it increases the buffer requirement to NoC, which is resource-constrained. The virtual channel buffer allocation scheme, virtual channel and switch allocation and arbitration logic are important parameters, which affects NoC performance. In this paper, our focus is to optimize the design of both the VA(virtual channel allocation) and SA(switch allocation) units and to increase the buffer utilization so as to minimize the latency, power consumption and area overhead and will in turn increase the performance of virtual-channel router.

REFERENCES

[1] W. J. Dally, “Virtual-channel flow control,” IEEE Trans. Parallel Distrib. Syst., vol. 3, no. 2, Mar. 1992, pp. 194–205.

[2] L. Benini and G. De Micheli, “Networks on chips: A new SoC paradigm,” Computer, vol. 35, no. 1, 2002, pp. 70–78.

[3] W. J. Dally and B. Towles, “Route packets, not wires: On-chip interconnection networks,” in DAC ’01: Proceedings of the 38th Conference on Design Automation, Jun. 2001, pp. 684–689.

[4] P. Guerrier and A. Greiner, “A generic architecture for on-chip packet-switched interconnections,” in DATE ’00: Proceedings of the Conference on Design, Automation and Test in Europe, Mar. 2000, pp. 250–256.

[5] A. Mello, L. Tedesco, N. Calazans and F. Moraes, “Virtual channels in networks on chip: implementation and evaluation on hermes NoC,” in Proceedings of the 18th annual symposium on Integrated circuits and system design, ACM: Florianolpolis, Brazil. 2005.

[6] J. Suseela and V. Muthukumar, “Loopback Virtual Channel Router Architecture for Network on Chip,” in Proceedings of the Ninth International Conference on Information Technology- New Generations. Apr. 2012, pp. 534 – 539.

[7] W. J. Dally, “Virtual-channel flow control,” IEEE Trans. Parallel Distrib. Syst., vol. 3, no. 2, Mar. 1992, pp. 194–205.

[8] L. Peh and W. J. Dally, “A delay model and speculative architecture for pipelined routers,” in Proc. Int. Symp. High-Performance Comput. Architecture, Jan. 2001, pp. 255–266.

[9] R. Mullins, A. West, and S. Moore, “Low-latency virtual-channel routers for on-chip networks,” in Proc. Int. Symp. Comput. Architecture, Jun. 2004, pp. 188–197.

[10] C. A. Nicopoulos et al., “ViChaR: A dynamic virtual channel regulator for network-on-chip routers,” in Proc. Int. Symp. Micro architecture, Dec. 2006, pp. 333–346.

[11] L. Peh and W. J. Dally, “Flit-reservation flow control,” in Proc. Int. Symp. High-Performance Comput. Architecture, Jan. 2000, pp. 73–84.

[12] J. Hu, U. Y. Ogras, and R. Marculescu, “System-level buffer allocation for application-specific networks-on-chip router design,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 25, no. 12, Dec. 2006, pp. 2919–2933.

[13] J. Kim, “Low-cost router micro architecture for on chip networks,” in 42nd Annual IEEE/ACM Int. Symp. Micro architecture (MICRO-42), New York, NY, USA, December 2009, pp. 255–266.

[14] W J. Dally, B. Towels, Principles and Practices of Interconnection Networks, Morgan Kaufmann Publishers Inc, 2003, pp. 305-324.

[15] J. Kim, D. Park, T. Theocharides, N. Vijaykrishnan and C. R. Das, “A low latency router supporting adaptivity for on-chip interconnects,” in Proceedings of the 42nd annual Design Automation Conference, .ACM: New York, USA, 2005.

[16] P Gratz, B. Grot, and S. W. Keckler, “Regional congestion awareness for load balance in networks-on-chip,” in IEEE 14th International Symposium on High Performance Computer Architecture, HPCA 2008, pp. 203-214.

ISBN: 978-81-923777-9-7

[17] T. Bjerregaard, J. Sparso, “A router architecture for connection oriented service guarantees in the MANGO clock less network-on chip,” in Proceeding of the conference on Design, Automation and Test in Europe, IEEE Computer Society, 2005, pp. 1226-1231.

[18] C. A. Nicopoulos, et al., “ViChaR: A dynamic virtual channel regulator for network-on-chip routers,” in MICRO ’39: Proceedings of the 39th Annual IEEE/ACM International Symposium on Micro architecture, Dec. 2006, pp. 333–346.

[19] F. Liu, H. Gu, and Y. Yang, “Performance Study of Virtual Channel Router for Network-on-Chip,” 2010 International Conference On Computer Design And Applications (ICCDA 2010), June 2010,pp. v5 255 – v5 259.

[20] C. Xuning and L. S. Peh, “Leakage power modeling and optimization in interconnection networks,” in Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED), 2003, pp. 90-95.

ISBN: 978-81-923777-9-7