

Survey on Arbitration Techniques Used in On-Chip Router Architecture

Pranita S. Muley, PG Scholar, S. D. College of Engineering, Wardha, M.S. India.

Sanghapal D. Kamble (Guide), Assistant Professor, S. D. College of Engineering, Wardha, M.S. India

Abstract: On chip communication is the new research in the technology. Network-on-Chip (NoC) is a new design method of communication network into System-on-Chip(SoC). Problems of traditional bus-based SoC can be solved and it will give the better communication requirements for SoC design. Efficient communication between devices of NoC are required, router are used for that. This router is called as the communication key in NoC. Router architecture is used to determine the performance of NoC and virtual-channel router is said to be a promising choice for this is the virtual-channel router. Arbitration techniques are needed to mediate access between multiple agents i.e.arbiter through a shared resource. In this paper, we studied NoC router architecture by using different arbitration techniques.

Keywords: System-on-Chip(SoC); Network-on-Chip(NoC); Router; Virtual Channel(VC); Arbiter

1.0 INTRODUCTION

Recently the more and more processing elements are embedding into a single on-chip to increase the processing speed. Traditional bus-based communication architecture has limits for the scalability, reusability and reliability of System-on-Chip (SoC)[5].

However buses suffer from poor scalability because as the number of processing elements increases, performance degrades dramatically. Hence they are not considered where processing elements are more[4]. Fixed scalability of system bus cannot meet the necessity of current System-on-Chip (SoC) implementation with limited number of functional units can be supported. Long global wires also has many problems in design as like noise coupling, routing congestion and hard timing closure.

Network-on-Chip (NoC) architectures have been obtained as an alternative to solve these problems by using a packet-based communication on systems[1]. Like in any other network, router is main element in the designing of a NoC system[3]. In a packet switched network, the function of the router is to forward an incoming packet to the destination source if it is directly connected to it, or to forward the

packet to another router connected to it[4]. Performance of NoC is depended and calculated mainly based on the router architecture and virtual-channel router is a promising solution for Network-on-Chip[1].

2.0 RELATED DESCRIPTION

The main feature of NoC is the use of networking technology to exchange the data within the chip. All links in NoC can be simultaneously used to transmit the data that provides a high level of parallelism and it replace the typical communication architectures as shared buses or point-to-point dedicated wires[3]. In Network-on-Chip, various switching techniques used for forwarding an information through the network which have significant effect on the design of router architecture. Switching techniques are broadly categorized into circuit switching and packet switching.

Now a days, NoC designs are based on Packet switching [1]. Packet switching is again classified as Store and Forward (SAF), Wormhole (WH) and Virtual Cut Through (VCT). But all these techniques having a problems of Head-on-Line (HoL) blocking problem, which give input buffering contention in routers. To make a solution with these problems in router switching techniques, researchers have proposed different buffering allocation techniques, micro architectural buffer structures, and effective buffer arbitration algorithms.

In [13] J. Dally introduced the virtual channel for deadlock-free routing for networks. Dally and Towels illustrate the basic virtual-channel router architecture [11] and showed that virtual-channel router works in a pipeline to decrease router delay. In [15] authors introduced low latency virtual-channel router in which a single flit can travel through VC router within only one cycle. In Virtual channel router is used to solve network deadlock by adopting adaptive routing[16].

A multi-priority arbiter which has priorities that assigned in a probabilistic manner; rather than absolute manner. It proved worthy in improving global fairness in the network.[6] Many researchers are developing various arbitration schemes to achieve an efficient allocation and

reduce packet latency. A lot of arbiters have been developed in the routers of NoC such as round robin arbiter, fixed priority arbiter and so on. Round robin arbiter gives the fair treatment with each input port and guarantees fairness in scheduling. Fixed priority arbiter always assigns the highest priority to the requiring input port when contention happens. The input ports with lower priority may rarely be authorized which results in extremely unfair.[5] There are three new arbitration techniques using the concept of round robin mechanism for the NoC router. They determined the scheduling sequence based on the previous experience to heuristic information. So all the packets input ports are detected and their routing paths are to be calculated before delivering them, which is time consuming and a complex task. For every clock cycle, it detected the loads of input ports and each input port gets its priority being adjusted dynamically[6].

3.0 NOC ROUTER ARCHITECTURE

NoC architecture which is composed of two parts: router and data link. Router which can store and forward data, and the data link can transmits the signals from one router to its neighbor. The architecture of router is as shown in figure.

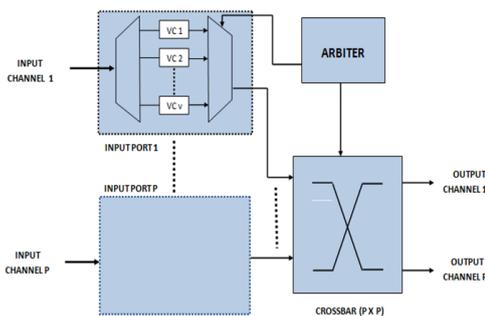


Fig.1 NoC Router Architecture

It consists of mainly Virtual channel, Arbiter, Crossbar.

3.1 Virtual Channel

It is logical channel which is obtained when physical channel is divided into a multiple number of logical channel. A virtual channel has its own queue, but it shares the bandwidth of the physical channel in a

time multiplexed fashion. Virtual channels give the flexibility, better channel utilization and improve network throughput and reduce the effect of blocking.

3.2 Arbiter

Arbiter which receives requests from input buffers and allocates virtual channels to requests and then provides grant signals to request initiators. The techniques required for coordinating mediate access between multiple agents through a shared resource is called arbitration and the agent which follows this logic so as to assign output port to input port is called arbiter. The significance of arbiter is to enhance the connectivity between the input and output ports, determining the implementation sequence of the routing paths so as to overcome the problem of conflicting requests of the input ports for the same outputPort[6].

3.3 Crossbar

The crossbar switches granted input requests and forwards the request data to data link, and then the request data is transmitted to the next hop router through data link[8].

4.0 ARBITRATION TECHNIQUES

In this, we will be analyzing various recently used arbitration techniques and their problems. But firstly we need to understand some basic problems:

Congestion:- Many input ports are requesting for the same output port.

Starvation:- It is type of unfairness in which all the input ports do not give an equal chance of accessing the output port.

Deadlock:- Output port can't be accessed by an input port as other input port does not release the resource.

Livelock:- Packets from the input port are moving but they cannot reach the desired output port.

Head-of-Line Blocking:- It happens in the case that two input ports request for the same output port but it is already being occupied by one of the input port, so another input port and the coming input port requests can't advance to the desired output port, so the network performance is degrading.

4.1 Fixed Priority Arbiter

Fixed arbiter is a simple which has a predetermined priority order grants access to a shared resource based on which it grants access to a shared resource. It has a linear array of basic bit cells which results in the generation of the grant G_i if both the input request R_i and incoming priority signal C_i are asserted. In case, R_i can't be asserted, it is transferred to the next cell. This arbitration technique is not very complex and is easy to implement[6]. The Problems in this scheduling is as -

- 1) Critical path delay grant is linearly proportional to the number of inputs.
- 2) The starvation problem is as only occurred as one input port is provided with a highest priority while all other with a low priority to access the output port i.e. an unfair arbitration technique.

4.2 Round-Robin Arbiter

This arbiter gives high degree of fairness with the agents by treating each input port fairly and guaranteeing fairness in scheduling. Therefore each input port gives an equal chance to access the output port and because of that the starvation problem can be solved [1]. The arbitration technique can be summarized as -

(Fixed Priority+ Bit Cell+ Priority Select Input) wrapped around the last bit cell's priority output to the first bit cell's priority output. State registers drive them to be rotated by one bit position. So, the requesting agents get the grant one after another.

The problems obtained are as -

- 1) This arbiter may degrade the efficiency for some input ports.

- 2) It is a somewhat time-consuming and is mainly contributed by the Input Selector to grant the requests, which also determines critical path delay.

The complexity of the Input Selector can be observed in two types:

- (a) The changing priority: The priority of the inputs get changed by grading the inputs and the request (RPT) value. Hence, the Input Selector take all possible priority settings.
- (b) The circular priority order: As there is increase in complexity of the priority processing as the priority order is circular. This gives the result in two separate conditions for the grant decisions: for requests at or below the Request [RPT] and requests above Request [RPT]. So, the critical path gets enhanced by this two parted decision.

4.3 Matrix Arbiter

Matrix arbiter is used when the packets are destined for the same output port with the same priority [17]. The access to the output port is achieved by using a pairwise precedence in between the entire request by input ports. As the transmission of the last packet is over then the output port is released. The matrix is created by the matrix arbiter with the help of input and output port. In this matrix, set the bits for the input port which have requested for the same output port. After checking is completed for the inputs, the priorities are assigned correspondingly. A control signal is generated by the matrix arbiter and the source packet is transmitted to destination through a particular select line. It has two types of delay: grant (grant generation delay grows logarithmically with the number of inputs) and update (update delay)[6]. The below is the figure of Matrix arbitration where H = High & L = Low.



Fig.2 Matrix Arbiter

The problems of this scheduling are -

1) As it update the status of precedence values $2 \times n$ registers from each grant signal for the fan-out are required, hence there is increase in the update delay by a large fraction.

2) No. of resister require are more, so Complexity is high to hold the precedence matrix and updating it a large number of registers. It is observed that the cost is almost four times taking into account the allocator's implementation cost.

4.4 Dynamic Adaptive Arbiter (DAA)

DAA is operated on the buffer status and can change the priority of input port dynamically. The input port having highest priority signal is assigned when the buffer signal is full. This input port firstly occupy the desired output port prior to other requests, and result in a deduction in the buffer pressure of the input port and improves the NoC performance. To remove the problem of starvation, it applies the principle of round robin mechanism. Arbiter in case of contention determines the output sequence. Output contention happens when the packets need to be released at the same output port but packet arrives at different input port. For solving the contention, an arbitration mechanism is necessary to allow only one input port to access the output port. According to the buffer status of the port, a dynamic arbitration priority is assigned for each input to reduce the problem of head-of-line blocking problem. This arbiter keep track of the buffer status of these input ports and check whether their buffers are full for the input ports which simultaneously request the same output port. Buffers are easily full for heavy-load input ports. As the delivery of packets in the full buffer is not done immediately, this results into latency and the packets will be halted. So they are assigned with high priority. In such cases, fairness is to be maintained. For light-load distribution there might be input port whose buffer is full. All input ports therefore are requested with an equal chance to Occupy. Starvation problem is removed for the low priority input ports, DAA incorporate a counter and a comparator. The counter function keeps the record of

the number of controlling times for high priority input ports. All input ports are scheduled with equal priority if the number of authorizing times is bigger than $T_{threshold}$. $T_{threshold}$ depends on the traffic characteristics. The request signal is disabled for one Round Robin arbiter but when the other is working. This gives the result that in each half cycle no request signal is missed by DAA. The corresponding buffer full signal of one Round Robin Arbiter is disabled, when the other RRA is in working. The corresponding buffer full signal will be disabled when a buffer of input port is not full. The packets with light loaded input ports which will accumulate more and more packets with the advance of time with the fact that they have no chance to make access to the desired output port and occupy it. Since the time passes with accumulation of more and more packets, their corresponding buffers become full. So, they will be assigned with high priority to occupy the desired the output port. For some input ports whose buffers are never full, they have a chance to occupy their desired output port by overcoming the $T_{threshold}$ [1].

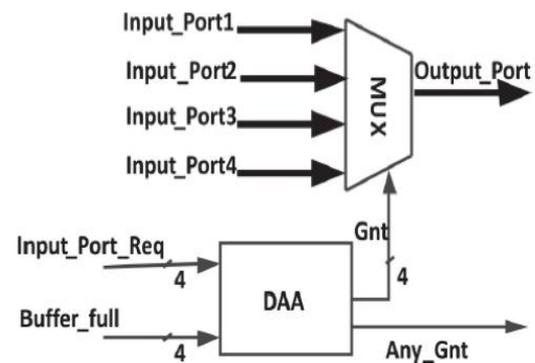


Fig.3 Block diagram for one output port DAA

This has a scheduling as follows -

(Input Port Request + Buffer Signal) passed through AND Gate = Buffer request signal (x). Now (x + Input set) are applied to a MUX which gets to one of the RRA finally. For 2nd RRA, Input port request+ Input set (coming across an (AND gate (counter (here $T_{threshold}$ is introduced) and a comparator)) + NOT

Gate) acts as inputs to the MUX which then is finally give to the 2nd RRA. (RRA + RRA) passed through an OR gate= Grant generated; (RRA + RRA) passed

through AND gate= Waiting Signal is generated. This method has a problem that it cannot overcome the problem of livelock i.e. the packets are moving but they cannot towards the desired destination.

5.0 CONCLUSION

In a NOC design, we should consider the need for an arbiter to resolve conflicts on shared resources (i.e., bus or equivalent communication channels) among multiple bus masters (e.g., processors). In a bus-based system, processors could be stalled because of bus conflicts. Therefore a high-performance arbiter is needed to solve bus contentions among bus masters; such a fast arbiter can also reduce processor stall time by shortening arbitration delays. In this, we have studied the impact of different scheduling techniques on multiprocessor System-on-chip. We have also studied the different arbitration techniques that required in designing of NoC with congestion free, low latency, high speed scheduler and also the problems of starvation, heavy inputs load, better trade-off between performance and implementation cost so that it provides guaranteed service and best-effort service. Thus we can use DAA as the base of operation.

REFERENCES

[1] Design of NoC Router Architecture using VHDL Minakshi M. Wanjari Electronics Engineering Dept. PCE Nagpur, India Pankaj Agrawal Electronics and Comm. Engineering. Dept.RCOEM Nagpur, India R. V. Kshirsagar Electronics Engineering Dept. PCE Nagpur.

[2] Design And Analysis Of Five Port router For Network On Chip Swapna S Department of Electronics and Communication Engineering, National Institute of Technology, Rourkela, Odisha Email: s.swapna.pai@gmail.com Ayas Kanta Swain Department of Electronics and Communication Engineering, National Institute of Technology, Rourkela, Odisha Emailswain.ayas@gmail.com Kamala Kanta Mahapatra Department of Electronics and Communication

Engineering, National Institute of Technology, Rourkela, Odisha. \Email:kkm@nitrkl.ac.inshirsagar,IJCA, volume, 115-no.

[3] Review on Network on Chip Router Design V.A.Bute, D.S. Chaudhari E&TC, GCOEA, Amravati, 2 E&TC, GCOEA, Amravati Maharashtra, India.

[4] Design and Analysis of On-Chip Router for Network on Chip Anuprita.S. Kale, Prof. M.A.Gaikwad Electronics Department, Bapurao Deshmukh college of engineering, Wardha M. S. India Professor, Electronics Department, Bapurao Deshmukh college of engineering, Wardha M.S India anupritakale@gmail.com gaikma@rediffmail.com

[5] A dynamic adaptive arbiter for Network-on-Chip Yanhua Liu^{1,2}, Jie Jin^{2,3}, Zongsheng Lai¹ ¹Institute of Microelectronics Circuit & System, East China Normal University, Shanghai, China ²Jiangsu Provincial Key Lab of ASIC Design, Nantong University, Nantong, China ³Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai, China.

[6] Problems Encountered in Various Arbitration Techniques Used in NOC Router: A Survey Kunj Jain¹, Sandeep K Singh², Alak Majumder³, Abir J Mondal³ ¹B-Tech Final Year, Department of ECE, NIT Arunachal Pradesh, Yupia, India - 791112 ²M-Tech Final Year, Department of CSE, NIT Arunachal Pradesh, Yupia, India - 791112 ³Assistant Professor, Department of ECE, NIT Arunachal Pradesh, Yupia, India. W. J. Dally, "Virtual-channel flow control," IEEE Trans. Parallel Distrib. Syst., vol. 3, no. 2 Mar. 1992, pp. 194-205.

[7] A.S. Kale and M.A.Gaikwad, 'Design and Analysis of On-Chip Router for Network On Chip,' International Journal of Computer Trends and Technology, vol. 9, no. 6, pp. 182-186, 2011.

[8] "An Innovative Power-Efficient Architecture for Input Buffer of Network on Chip" Kun Huang Jun Wang Ge Zhang, Key Laboratory of Computer System and Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijingss.

[9] "Zero-Efficient Buffer Design for Reliable Network-on-Chip in Tiled Chip-Multi-Processor" Jun

Wang¹, Hongbo Zeng¹, Kun Huang¹, Ge Zhang¹, Yan Tang¹, Institute of Computing Technology, Chinese Academy of Sciences; The Ohio State University, wangjun, bzeng,kunhuang,gzhang@ict.ac.cn;2tangya@cse.ohio-state.edu.

[10] “Energy-Efficient Input Buffer Design using Data-Transition Oriented Model” Jun Wang, Kun Huang, Ge Zhang, Weiwu Hu, Feng Zhang Key Laboratory of Computer System and Architecture, ICT, CAS, Beijing, 100080, China,{wangjun, huangkun, gzhang, hww, zhangfeng}@ict.ac.cn.

[11] 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.

[12] 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.

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

[14] 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.

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

[16] 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.