S. Sundara Mahalakshmi et al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 4, Issue 4, Dec 2017, pp.212-215

Evaluation of Arbitration Techniques for SoC Design

S.Sundara Mahalakshmi [1], S.Cammillus[2]

[1]PG scholar, Department of ECE, National Engineering College, Kovilpatti

[2] Assistant Professor, Department of ECE, National Engineering College, Kovilpatti.

Abstract:- Network on chip (NoC) is part of a System on Chip (SoC) and Network on chip is used for on chip communication. It acts like router switches. NoC router made up of FIFO buffer to store the data, Crossbar switch (multiplexing and De-multiplexing), and Arbitration technique. Arbiter is the most important part in the Network on Chip and it is used when number of input port are request for same output port. In this work, a Modified Index - based Round Robin arbiter has been proposed. The attitude and architecture of proposed arbiter gives on to lower power consumption, less chip area and low latency than aforementioned arbiters. The design of Modified Index based round robin arbiter is coded in VHDL, synthesized and simulated in XILINX 13.2 version. Finally, the performance of Round Robin arbiter and Matrix arbiter are compared with the results of proposed work in terms of priority.

Keywords--- Matrix Arbiter, System-on-Chip, Network-on- Chip, Index-based Round Robin.

I. INTRODUCTION

Very-large-scale integration (VLSI) is the process of combining thousands of transistors into a single chip called integrated circuit (IC). One billion transistors placed on a single chip is possible with the current CMOS technology. A system on a chip or system on chip combines the electronic circuits of various computer components onto a single chip. A SoC is heterogeneous, in addition to classical digital components: processor, memory, bus, etc.; It may contain digital, analog and radio-frequency functions on a single substrate [2]. Network on chip is part of a System on Chip and Network on chip is used for on chip communication. NoC router made up of FIFO buffer to store the data, Crossbar switch (multiplexing and De-multiplexing), and Arbitration technique. Arbitration of different ports controlled by the arbiter. Arbiters are used to access shared resources such as a buffer, channel or switch-port. From different input ports packets may have go to the same output port. Arbiter decides to which ports packets will be served in which order. Arbitration works only when there is more than one In this paper, explore the arbiters used in NoC Router. Arbiter plays an important role in NoC router. An NoC Router facilitates communication among IP cores of an SoC. It includes a network of switches (routers) that are interconnected by communication links as shown in Figure 1.a. Figure 1.b shows a typical NoC router that consists of some input and output ports, an arbiter and a crossbar switch. Normally centralized switching techniques is preferred when number of cores are less in a SoC. In the case of 2D Mesh, switches at each node contains routers and some of I/O ports and packet need to travel using different router and hops to reaching its final target on the network.

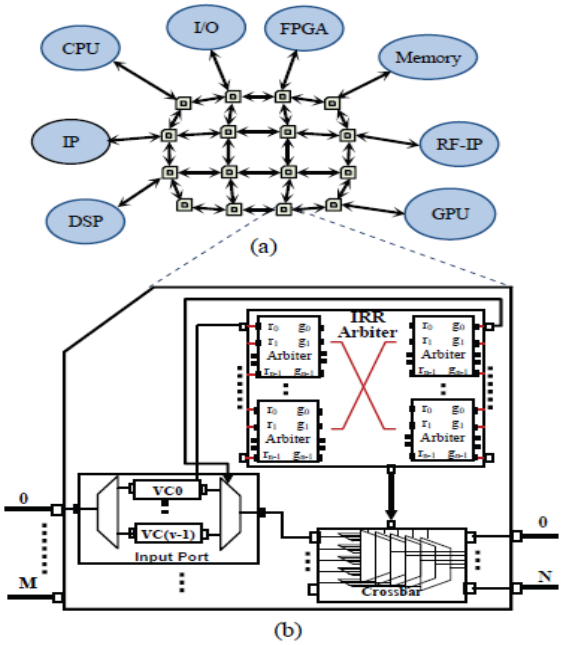


Figure 1. (a) 2D SoC mesh. (b) NoC router.

request for the same port otherwise data flow occurs normally. The data paths of routers are made up of buffers and switches. The control parts of routers are largely composed of arbiters and allocators. Allocators are nothing but used to allocate virtual channels to packets and to allocate switch cycles to flits. Arbiters, which resolves multiple request for the single resource, they form the fundamental block for the allocators that match multiple requests with multiple resources. To grant the resource at a time, arbiter is required. whenever there are resources such as buffer, channel, or crossbar switch ports is shared by

many agents. Use of virtual channel(VC) mechanism, makes the arbiter's structure more complex.

A key property of an arbiter is its fairness. A fair, arbiter is one that provides equal service to the different request's. The exact meaning of equal, can vary from application to application. The useful definitions of fairness are:

Weak fairness: Every request is eventually served,

Strong fairness: Requests will be served equally often.

FIFO fairness: Requesters are serviced in the order they made their requests There are a number of benefits of SoC. Application of SoC requires less memory, provides freedom in design, reduces chip count and also requires less space. Consequently, this leads to reduction in cost to consumer. Further, reliability and performance will be highly enhanced as the system is now is on the chip.

II.METHODOLOGY

A. The mechanism of Round-Robin Arbiter

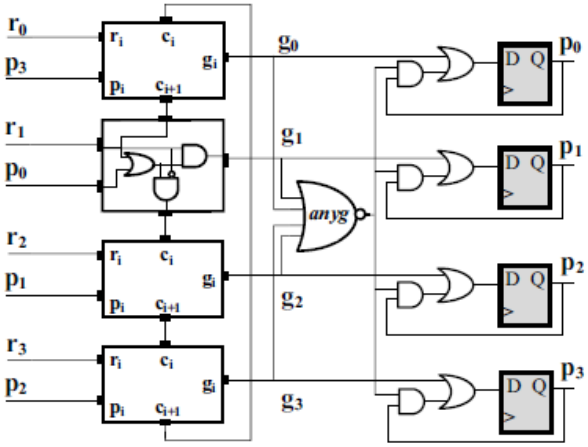


Figure 2. RoR arbiter architecture.

The functionality of a round-robin arbiter can be explained as a request that is just granted will have the lowest priority on the next arbitration cycle [3]. The round robin arbiters are simple, easy to implement, and starvation free. When the input requests are large in numbers, the structure of round robin arbiter grows that leads to large chip area, higher power consumption, and critical path delay. In an NoC design, the critical path delay of arbiter usually dominates among the critical path delays of input-port and crossbar switch due to the architectural complexity of arbiter as compared to those of port and crossbar switch. Therefore, the arbiter circuit determines the maximum frequency (or the speed), Fmax of an NoC router. The critical impact of arbiter on the performance of the NoC system and the characteristic behaviour of round robin architectures have created a lot of interest of NoC researchers.

B. The mechanism of Matrix Arbiter

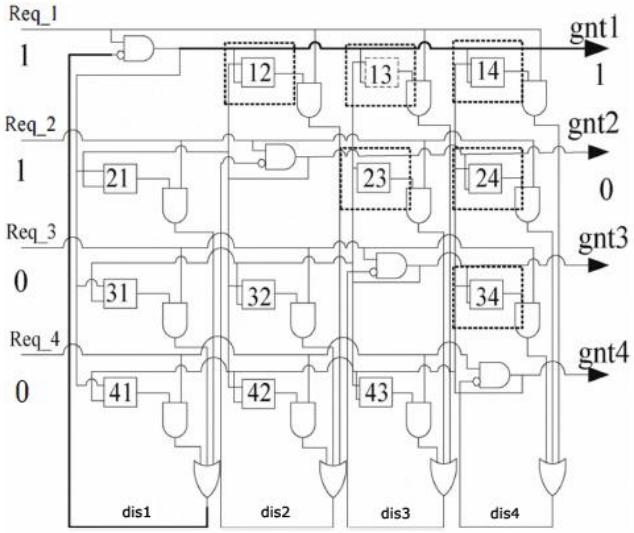


Figure 3. Matrix-arbiter architecture

Matrix Arbiter works on the principle of least-recently-served policy. It is the strong fairness arbiter. Matrix arbiter has four inputs. Solid boxes denoted the six flip flops maintained the state in the upper triangular portion of matrix. Each of the shaded boxes represents the inverted output of the diagonally symmetric solid boxes in the lower triangular portion of matrix. Matrix Arbiter implement a least recently served order of priority and it maintains a triangular array of state bits pmn. The element pmn in row m and column n indicates that request m takes priority over request n [3]. The 4-input gate architecture of a matrix arbiter is shown in figure3. In the upper triangular portion of the matrix, each block with dotted line describes the S-R latch. The state is maintained in the 6 S-R latches denoted by dotted line blocks. The complementary output of the diagonally symmetric solid box represented in each of the dotted line blocks in the lower triangular portion of the matrix. For example, r0 request is asserted and bit p02 is set then the signal dis 2 will be asserted for disable the lower priority request 2 as shown in Figure 3. If it is not disable, the request travel to the corresponding grant output through a single and gate.

III. PROPOSED WORK

A. The mechanism of Fixed Priority arbiter

For a fixed priority arbiter, the priority of requests is established in a linear order. Figure 4 illustrates a 4-input fixed-priority arbiter where r0 has the highest and r3 has the least priority. Fixed priority arbiters provide weak fairness arbitration because when a request is continuously asserted,

none of its following requests will ever be served.

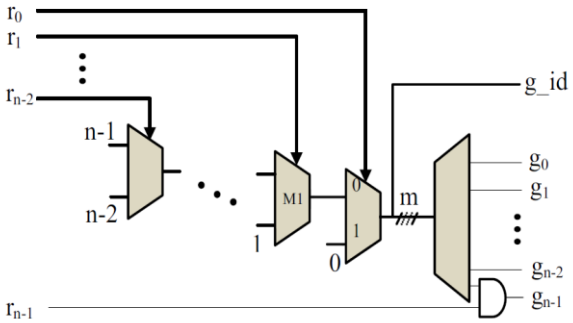


Figure 4. Fixed priority arbiter

B. The mechanism of Variable Priority arbiter

In order to have a fair iterative arbiter, variable priority arbiter is used as illustrated in Figure 5. An OR gate and a priority input signal, pi is added to each cell of the fixed priority arbiter shown in Figure 5. When p1 is set, its corresponding request, r1 has high priority and the priority decreases from that point cyclically around the circular carry chain.

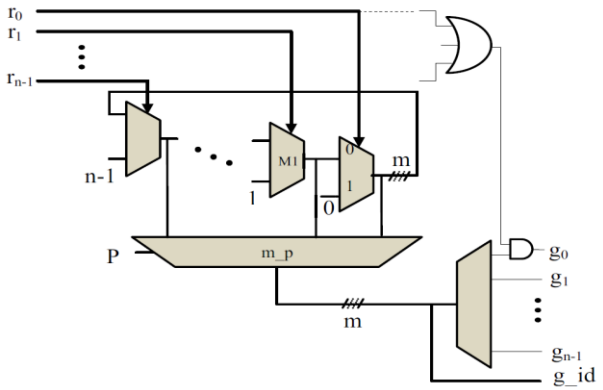


Figure 5. Variable priority arbiter

C. The mechanism of MIRR Arbiter

When we use a fixed priority arbiter in NoC design, there is no limit to how long a lower priority request should wait until it receives a grant. An arbiter is a logical element serving to select the order of access to a shared resource. An arbiter would typically employ a scheduling algorithm to decide which one on several requestors would be serviced. The round robin arbitration, in its basic form, is a simple time slice scheduling, allowing each requestor an equal share of the time in accessing a memory or a limited processing resource in a circular order. For more complicated applications of round robin arbitration, such as packet switching, using a fixed time slice for each requestor is inefficient as the processing time of each data element, impacts the fairness of the arbitration. Therefore, there are several flavors of round robin arbitration, each suited for a different application. Finally, we have to choose index based round robin arbiter with decoder reuse technique for achieving high speed with less delay overhead. Figure 6 is the circuit diagram of Modified index based round robin arbiter. Each input is in the index format which is applied to each individual multiplexer. Number of outputs from all individual multiplexers is sorted out to a single output by a mux MP. In a Modified IRRA, instead of demux for decoding of output decoder is used. Decoder gives the error free output when there is no input request whereas demultiplexer was sending with some error of the previous output. Any_r signal is the ORing of all the inputs. This design of Modified IRRA reduces the area, delay and power consumption of the previous design.

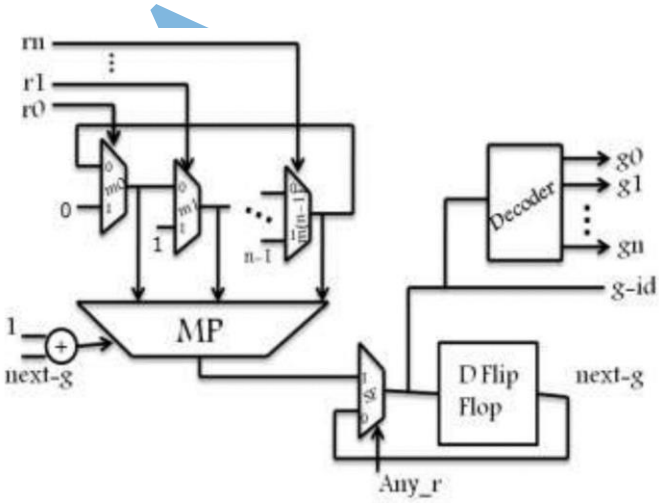


Figure 6. Modified IRR Arbiter

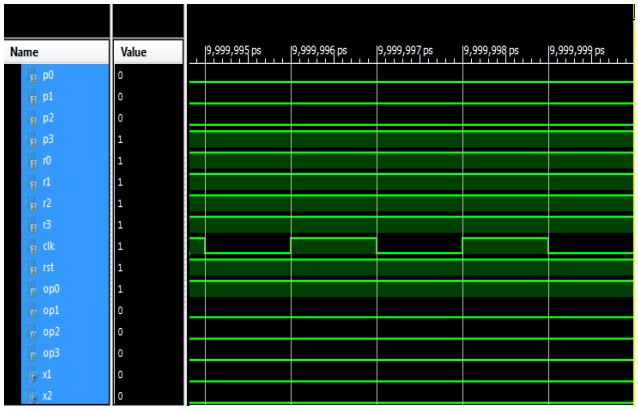
IV. ANALYTICAL COMPARISION

Hardware analysis performed to compare the expected performance and hardware overhead of the aforementioned Round robin and Matrix arbiter with MIRR arbiter. The MIRR routers have less chip area, less power consumption, and faster frequency as compared to the Matrix arbiter and Round Robin arbiter. The main advantage of an arbiter circuit are speed, area and power consumption. The maximum clock frequency (Fmax) of an arbiter circuit is the measure for speed. The clock frequency of an arbiter depends on the critical path delay between two registers clocked at the same time. In this paper no algorithm is used to optimize the circuits and calculated the summation of the areas and powers of all the cells of each arbiter to estimate their power and area. The power includes both static and dynamic powers. For speed estimation, the critical path delay between two registers of each circuit is calculated.

V. RESULTS

Following are the obtained results of Modified IRRA and comparison table.

A. Simulation result



B. Comparision table

DESIGN	DELAY (ns)	AREA (μm^2)	POWER CONSUMPTION (W)
MIRR	0.645	35	0.034
Matrix	1.070	44	0.092
RoR	1.305	74	0.115

VI.CONCLUSION

There are several arbitration mechanisms can be chosen when designing NOC, such as Round-robin arbiter and Matrix arbiter. However, in related literatures, there is lack of analysis, especially in resource, performance and power consumption of them. In this paper, three mechanisms are emulated on FPGA platform. Comparing the difference of them is meaningful for designing arbiters. The experiment details and the results are presented after comparing MIRR with other NoCs utilizing some previous buffering and arbitration approaches. The MIRR routers have less chip area, less power consumption, and faster frequency as compared to the Matrix arbiter and Round Robin arbiter. Moreover, the MIRR NoC performance shows much higher throughputs and lower average latencies than those of Matrix arbiter and Round Robin arbiter NoCs for various traffic patterns.

REFERENCES

[1] Jagrati Gupta1 and Nidhi Goel2, “Efficient Bus Arbitration Protocol for SoC Design,” International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp.396-400, 6-8 May 2015.

[2] Gopika Santhosh, “Design of Network on Chip with an Arbiter”, International Journal of Innovative Research in Computer and Communication Engineering(IJIRCCE), Vol. 3, Issue 9, September 2015.

[3] J. Arjana, “Low Latency NoC with Dynamic Priority based Matrix Arbiter,” Indian Journal of Science and Technology, Vol 9(29), August 2016.

[4] Rajeev Kamal and Juan M. Moreno Arostegui, "RTL Implementation and Analysis of Fixed Priority, Round Robin, and Matrix Arbiters for the NoC’s Routers," International Conference on Computing, Communication andAutomation (ICCCA2016).

[5] Z. Fu and X. Ling, "The design and implementation of arbiters for Network-on-chips," in Proc. Int. Conf. Industrial & Information Systems, pp. 292-295, Dalian, 2010.

[6] K. A. Helal, S. Attia, T. Ismail, H. Mostafa, "Priority-select arbiter: An efficient round-robin arbiter," in Proc. IEEE 13th Int. Conf. on New Circuits and Systems, pp. 1-4, Grenoble France, June 2015.

[7] Chang Hee Pyoun, Chi Ho Lin, Hi Seok Kim, Jong Wha Chong, “The Efficient Bus Arbitration Scheme in SoC Environment”, Proceedings of The 3rd IEEE International Workshop on System-on-Chip for Real-Time Applications, 2003.

[8] Ruqaiya Khanam; Himanshu Sharma; Srisht Gaur (2015), “Design a low latency arbiter for on chip communication architecture”, International Conference on Computing, Communication & Automation, Pages:1421-1426.

