ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2017.0051

DESIGN OF FIVE PORT PRIORITY BASED ROUTER WITH PORT SELECTION LOGIC FOR NoC

Meenu Anna George[1], Aravindhan A[2] and Lakshminarayanan G[3]
[1,2]*Department of Electronics and Communication Engineering, Saintgits College of Engineering, India*
[3]*Department of Electronics and Communication Engineering, National Institute of Technology, Tiruchirappalli, India*

Abstract

Network-on-chip (NoC) is a relatively new technology to signaling that enables not only more efficient interconnects but also more efficient design and verification processes for modern SoCs. The communication through the NoC is performed by enabling processing element (PE) to send and receive packets through the network fabric composed of switches/routers connected together through physical links or channels. For effective global on-chip communication, routers provide efficient routing with comparatively low complexity and high performance. Communication deadlock may appear at the router network and can cause performance degradation and system failure. Based on these studies, a five port priority based router is proposed. Port selection logic is used for selecting the ports for data transmission to selective ports. The proposed router shows better performance when tested in Mesh and Torus topology. Round Robin Algorithm is used in arbiter, which handles the process with priority and has low power consumption. The designed router is implemented in Artix 7, Spartan 6 and Virtex 7 using Xilinx ISE 14.7 design tool and power consumption of five port router is taken in Synopsis VHDL and power compiler tool.

Keywords:

Network-On-Chip, Router, Communication Deadlock, Port Selection Logic, Round Robin Algorithm

1. INTRODUCTION

The number of transistors on an Integrated Circuits (IC) is doubling approximately every two years as predicted by Moore's law which has been followed for half a century. Due to the advancements in CMOS technology, it is easy to achieve billion of transistor on a single chip. A regular increase in the number of transistors on a single chip has increased over the years and more complex systems are developed on a single chip. Integration of all the components of a system on a single chip is known as System on Chip (SoC). The increased transistor densities, high operating frequencies and reduced product life cycle, describe today's semiconductor industry [1]. Due to the thermal density constraints and power consumption the IC have not been keeping up with the operating frequencies for the past years. Designers introduced multicore and much core architecture due to the increase in transistor density on chip which causes an increase in performance without a proportional increase in frequency. The whole system is integrated by adopting intellectual property (IP) cores to have an effective communication with each other by connecting it directly to each other. These IP cores can be a processor, Input/Output controllers, DSPs, memory, video controllers etc. They can be reused in different communication systems [2]. The increase in number of cores on a chip will increase the complexity of the chip. The IP cores can be directly connect to each other or buses are used for connecting the cores. As the complexity on a chip increases, the number of core connected to a bus is limited and multiple cores finds it difficult to communicate at the same time. Adaptive pipeline bus structure was proposed [3] for interlayer communication for improving the performance by reducing the complexity and congestion in bus arbitration. As the complexity increases so does the power usage per communication, leading to a higher capacitive load [4].

Arbitration problem exist for shared buses in multicore architecture on a chip [5]. Increase in number of links will increase the area and routing problems on SoC. Shared bus designs are efficient for communication but has underlying limitations. This is one of the major arising problems in SoC. Direct connections between cores will have less scalability. Based on these studies a shift in implementation style was much in need by using an idea which was proposed a decade before. This paradigm shift is called "Network on Chip" (NoC). It is defined as the front end solution to the backend problem [1]. A solution for global interconnects by using on-chip packet switched network. The switches are introduced on to the chip instead of direct connection between the cores. In [6] five port routers for NoC is used as a decoding logic in which the router had a delay for an amount of input data packets available at the input. To overcome this issue a five port priority based router is proposed which have port selection logic for identifying the output port, data error correction and availability of output ports.

The paper is organized as given below. Section 2 contains a background of NoC, Section 3 gives details about the communication architecture used in NoC, section 4 contains the details of basics of NoC router and section 5 contains the proposed design of priority based router, section 6 contains the experimental results and simulation of the implemented router in FPGA, section 7 contains the conclusion of the paper.

2. NETWORK ON CHIP – AN OVERVIEW

NoC aim for a more efficient interconnects and communication between IP cores over an on-chip network. An example for the NoC basic interconnection network is shown in Fig.1. It consists of four basic functional blocks. These blocks are IP cores, Network Interface (NI), routing node and the links. The router helps in communication between the IP cores in a NoC. It is the main functional component in NoC interconnection network. The overall performance is determined by its area and speed of communication [7]. The NI provides the connection between the IP cores and the network adapters which helps in decoupling the cores from the communication network. The communication through NoC is done by enabling the IP cores to send and receive the data through the switches/routers. Network designs are scalable and can be changed according to the application requirement [8]. A Bidirectional NoC (BiNoC)

architecture with dynamically self-reconfigurable bidirectional channel can break the conventional NoC's [9].

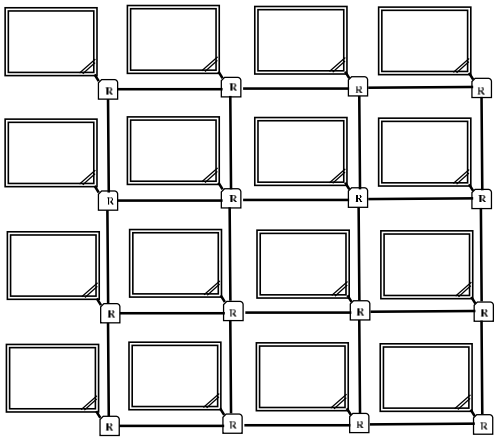

Fig.1. Basic NoC building block

3. NoC COMMUNICATION ARCHITECTURE

At first the topology for the network is determined and routing algorithm will be based on the topology and design constraints. The flow control scheme, the router and the link design are designed after the routing algorithm is selected.

3.1 NETWORK TOPOLOGY

The topology refers to the structure and the way in which the routers are connected to the network. It determines the connection between different nodes on the chip. The topology is selected in such a way that the area is minimized and maximum throughput is gained without causing bottlenecks [7]. Topology selection is identified by the device architect. The topology selection is application specific where the communication requirements are satisfied and minimum cost is selected. Performance, cost and scalability are the main factors considered in selection of network topology [1].

For a regular topology some routers will remain non-fully utilized but for irregular topology each routers will have different number of ports. The most common topologies of on chip interconnects are shown in Fig.2(a) Shared Bus, Fig.2(b) Ring, Fig.2(c) Crossbar, Fig.2(d) Mesh and Fig.2(e) Torus. 2D mesh is the simplest and most commonly used topology for NoCs. It assumes that all the links have the same length. Torus is another direct topology which consists of an n-dimensional grid with k nodes in each dimension. Ring topology suffers large end to end delay and star topology suffers central bottleneck.

The switching technique defines the way in which the input channel of the router/switch is connected to the output channel. The data are transmitted as messages, which are then splitted into packets, which in turn are splitted into flow control units (flits) and then to physical units (phits). Physical units is the unit of data transferred through the physical link (i.e.), the number of bits transmitted between the routers for a single clock cycle [1]. Flow control units are the unit of synchronization between routers. Packets are a set of flits with the same destination. Messages are a set of packets that completely transfer between nodes. There are two types of switching techniques in NoC network, circuit and packet switching. In circuit switching, a fixed path is established between the sender and the receiver, and this connection is reserved till the message is transferred. The packet switching transfers by segmenting the messages into smaller data packets and forward it from the sender to the receiver with different routers and delays for each packet.

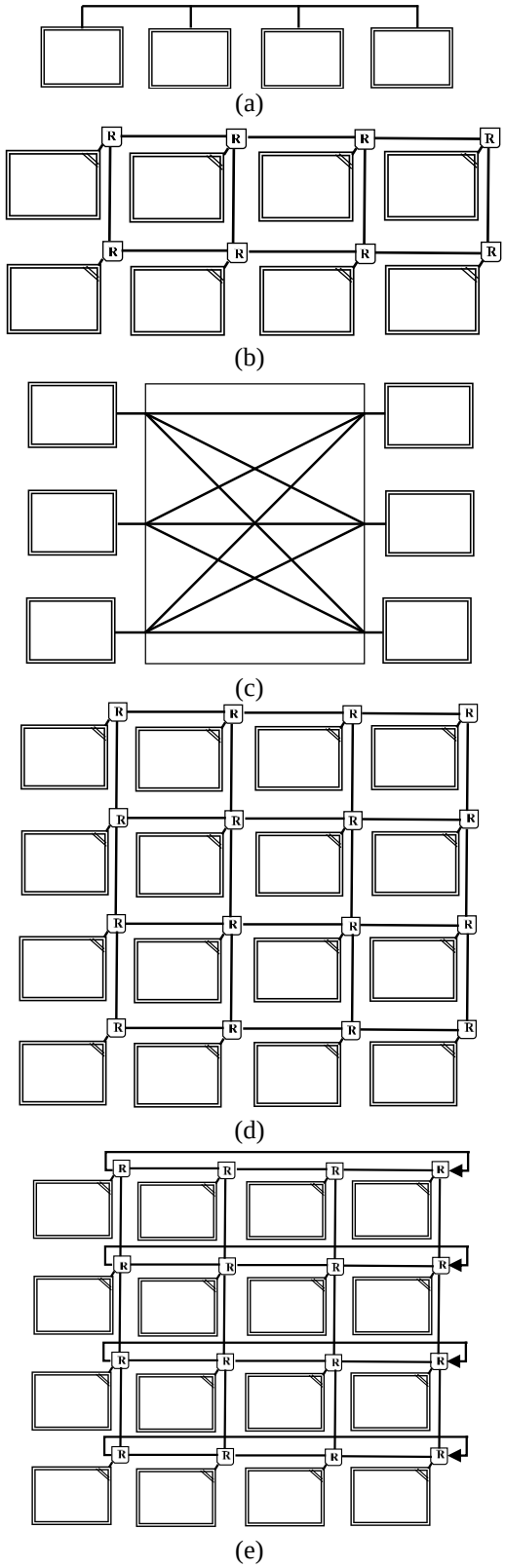

Fig.2. Common Topologies of NoC. (a). Shared Bus (b). Ring (c). Crossbar (d). Mesh (e). Torus

3.2 ROUTING ALGORITHM

The routing scheme for a network focuses in the low-level transfer of data [7]. There are several aspects to be considered for routing on network on chip which can be classified as deterministic or adaptive routing schemes. In deterministic routing scheme the path between the given source and destination pair is predetermined and is independent of the current network status. The popular deterministic routing scheme for mesh is XY routing [10]. In this routing a packet is routed along a row first, then routed along column to reach the destination. This routing is very simple and robust against deadlock. Round Robin algorithm is another most commonly used because it is fair and prevents starvation [11]. In adaptive routing scheme, routing decisions are made at each routing node on per-hop basis which considers the network status for avoiding the congestion and balancing network load. An adaptive arbiter based on round robin mechanism is proposed [12] which show improvement in packet injection rate and throughput of NoC. Adaptive routing is more complex for implementation than deterministic routing [13]. Partially adaptive routing schemes offer both characteristics of deterministic and adaptive routing. Routing schemes can be further defined as distributed and source routing. In Distributed routing the packet header contains the destination address. In source routing all the routing decisions are embedded into the packet header for the given destination. Application specific routing algorithm provides higher adaptivity and also superior performance when compared to the other algorithms for traffic cases [14].

4. NoC ROUTER

In general the packet travels through several routers to reach the destination. The routing algorithm decides the route for forwarding the packet to the destination router. A simple algorithm makes the router design less complex and faster. The block diagram of a typical 5 port switch/router is shown in the Fig.3. The ports of the router are designated as North port, South port, East port, West port and Local port.

The basic components of the router as follows,

i. Input /Output buffer for storing flits.
ii. Output port allocation logic for selection of output ports for each packet.
iii. Switch fabric for physical connection from input port to output port.
iv. Control Logic for overall synchronization.

The common tile based 2D Mesh Topology is shown in the Fig.4.

Designing buffer is very critical for achieving the overall good performance and area. NoC uses small registers for buffering. Each input and output buffers have temporary registers for storing data. In regular topologies all routers and ports within a router will have same buffer size. NoC router which supports shared bus architecture is proposed [15] which show a better performance when compared to input-buffer architecture. Crossbar switch facilitates the connection between the input port and the output port. Pass transistors and multiplexers are commonly used for implementation of the crossbar switch. Crossbar switch as pass transistors helps in reducing area and power consumption while the multiplexers can be automatically synthesized from RTL description and can be implemented in FPGA also [6]. The area of a crossbar increases with the square of number of ports. The arbiter and control block gives signal for controlling the crossbar switch.

Arbiter and the routing unit is the most important component of the router. The routing of packets from input port to output port is controlled by the arbiter. It helps in resolving the conflicts of simultaneous request to same output port. When the output port is ready for receiving the packet, arbiter sends selection signal to crossbar and control signal to input buffer for forwarding the packet through the crossbar switch to desired output port.

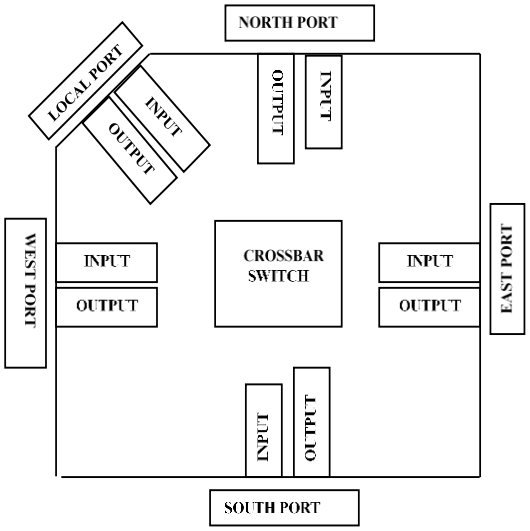

Fig.3. Block diagram of Mesh based Router

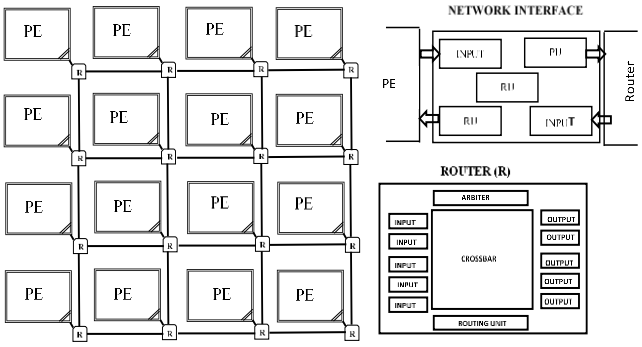

Fig.4. Tile based 2D Mesh topology

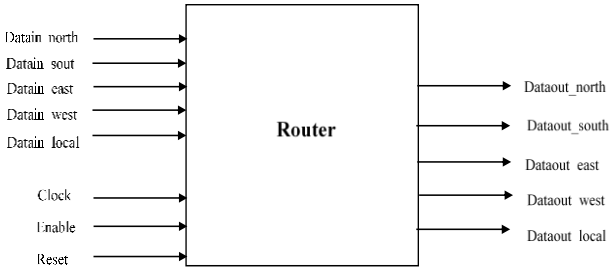

Fig.5. RTL Schematic of 5 Port Priority based Router

5. PROPOSED PRIORITY BASED ROUTER ARCHITECTURE

The RTL schematic of designed Router is shown in Fig.5. It consist of five data input ports (Datain_north, Datain_south, Datain_east, Datain_west, Datain_local), five data output ports (Dataout_north, Dataout_south, Dataout_east, Dataout_west, Dataout_local), and three global control signals Clock, Enable and Reset signal. The data used is 8 bit and the last three bits are used for selection of output port. The arbiter of the router uses the Round Robin Algorithm for the selection of output ports. The designed Router is implemented and tested in a 2×2 Mesh and 2×2 Torus topology.

The basic building blocks of the designed router are

- Port Selection Logic
- Registers
- Demultiplexers
- Buffers
- Arbiter

The internal architecture of the designed Router is shown in Fig.6.

5.1 PORT SELECTION LOGIC

The Port Selection Logic is used for determining the destination port. This logic will verify the destination address and is forwarded to the input registers for temporary storing of data. This logic eliminates the need for checking the destination port address at each components of Router. The Port Selection Logic has Clock, Enable and Reset signals in synchronous with the router signals. The last three bits of the data are selected and the corresponding output port is verified. This module is also used for the checking the availability of output ports and for data error correction. The faulty ports are identified using this module. The schematic of the Port Selection logic is shown in Fig.7.

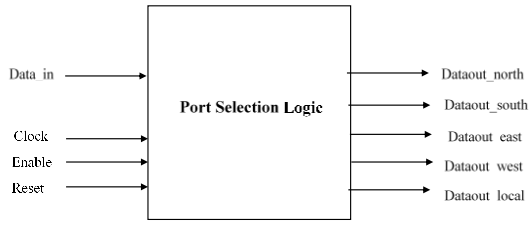

Fig.7. Schematic of Port Selection Logic

5.2 REGISTERS

Registers are used for temporary storage of input data received from the input ports. This 8-bit Registers have a Clock, Enable and Reset signals for operation. The data are stored during the positive edge of clock cycle. The output of the Port Selection Logic is stored in the Register. The input data is transferred to the output port at the positive edge of the clock of the Register if and only if the enable is 1 and the reset is 0. If the reset signal is 1, then the output port of the register is all set to zeros. If the enable signal is 0, then the output port keeps its current value. The schematic of register is shown in Fig.8.

5.3 DEMULTIPLEXERS

Demultiplexers forward the data to the buffer according to the selected port. This router uses 1 to 8 eight bit demultiplexers. These demultiplexers are used for placing the data to the desired address in buffer. The demultiplexers have a Clock, Enable and Reset signal which are in synchronous with the Router signals. It is operated during the positive cycle of the clock. If the Enable is 1 and Reset 0, the data is transferred from the register to the respective area in the buffer. The schematic of demultiplexer is shown in Fig.9.

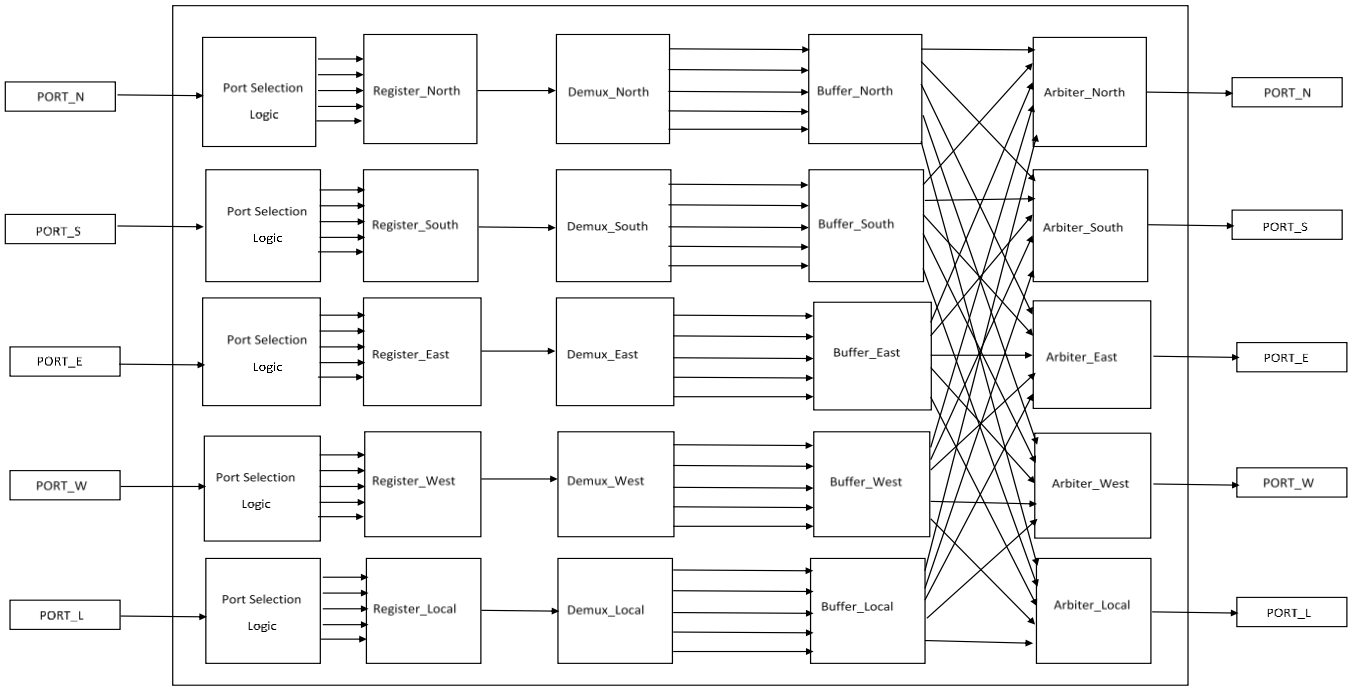

Fig.6. Proposed Router Architecture

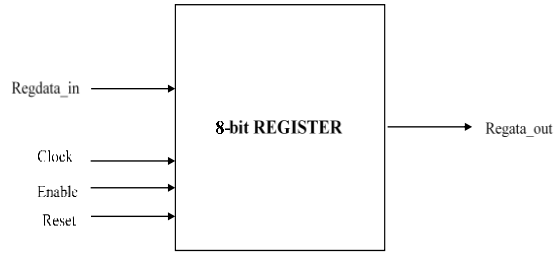

Fig.8. Schematic of Register

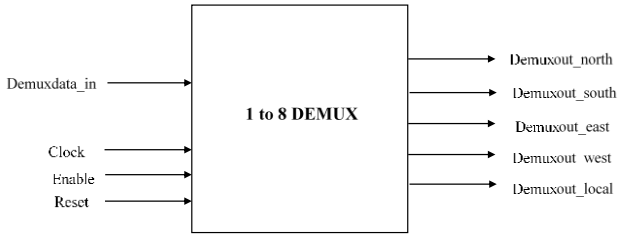

Fig.9. Schematic of Demultiplexer

5.4 BUFFER

The buffer is used for storing the data forwarded from the input ports. The buffer has five different address areas stored for the five ports. The bottom most areas is reserved for the local port and the top most for the north. The Port Selection Logic determines the output port and the data received from the register is transferred to the corresponding First-out (FIFO) and the pointer is used for selecting the corresponding port area to store the packet. The buffer has a Clock, Enable and Reset signal. The data will enter and leave the buffer at the rising edge of the clock. Write Request signal is used for storing the data and Read request signal is used for reading the data from the buffer. The schematic of buffer is shown in Fig.10.

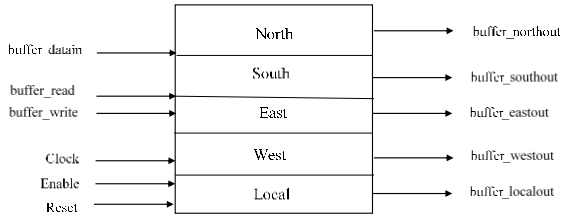

Fig.10. Schematic of Buffer

5.5 ARBITER

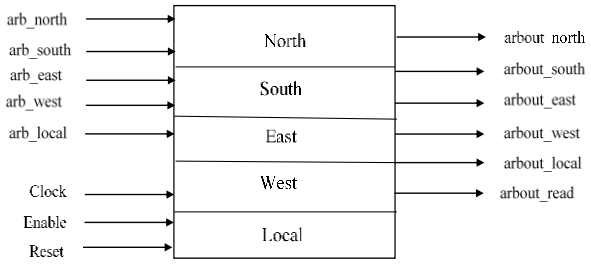

Fig.11. Schematic of Arbiter

The Arbiter in this designed router uses Round Robin Algorithm. In this router each port will be given a priority. The priorities are implemented in clockwise fashion, i.e., if the last input port serviced was north, then during next service, the North port will be given the least priority. This algorithm decides which input channel is to be given access to that output channel when many channels are requesting for the same output. The arbiter selects the data from the respective area of the buffer for the selected output port. The schematic of arbiter is shown in Fig.11.

6. EXPERIMENTAL RESULTS AND SIMULATION

Table.1. Single Router Implementation

Parameter	Artix 7	Spartan 6	Virtex 7
No. of Slice Registers	32	32	32
No. of Slice LUTs	23	23	24
No. of bonded IOBs	48	48	48
No. of LUT flip-flop pairs	30	29	29
Delay	1.724ns	2.603ns	1.806ns

Table.2. Mesh and Torus Topology Implementation

Parameter	Mesh 2×2			Torus 2×2		
	Artix 7	Spartan 6	Virtex 7	Artix 7	Spartan 6	Virtex 7
No. of Slice Registers	24	24	24	15	15	15
No. of Slice LUTs	13	16	14	9	9	8
No. of bonded IOBs	43	43	43	19	19	19
No. of LUT flip-flop pairs	22	22	22	15	16	15
Delay	1.385 ns	2.120 ns	1.524 ns	1.326 ns	2.009 ns	1.496 ns

Table.3. Five Port Single Router Power in milliwatts (mW)

Frequency	Power
500Mhz	95.379
200Mhz	48.134
100Mhz	18.989
50Mhz	9.612

The designed router is implemented in Artix 7, Spartan 6 and Virtex7 and simulated in Xilinx ISE 14.7 software. Simulation refers to the verification of design, its performance and function. A test bench is written to test the routing pattern from various ports. The output port will receive the data during the positive half of each clock cycle. Each output port will have a certain time slot to transfer the whole data to the respective routers. This prevents the livelock. The Synthesis & Simulation results of Single Router implemented on Artix 7, Spartan 6 and Virtex 7 are shown in

Table.1 and Synthesis & Simulation results of the designed router in Mesh and Torus topology is shown in Table.2.

The power consumption of the designed NoC router, are modeled in VHDL .The power consumption are calculated using Synopsys Design Compiler™, VHDLSIM™ and Power Compiler™ tools. The *analyze* and *elaborate* commands read the RTL design into an active memory and convert the design to GTECH design format which is done by the Design Compiler tool. Then Switching Activity Interchange Format (SAIF) file is generated using *rtl2saif* command. This forward-annotated file contains directives for tracing the design elements during simulation. The forward-annotated SAIF file with VHDL test bench is fed into the simulator to generate a back-annotated SAIF file. The back annotated SAIF file contains information about the switching activity of the synthesis-invariant elements in the design. This file is then used with the gate-level net-list file (Data-Base (DB) file) to calculate the power consumption of the router which is produced by the Design Compiler tool. Power Compiler is used to calculate the power. The Power Consumption of the single router is verified for various operating frequencies as shown in Table.3.

The Fig.12 and Fig.13 shows the verification of the designed router in 2×2 Mesh and Torus topology. The Fig.14 shows the simulation results of a single router.

7. CONCLUSION

The five port Priority based router is implemented on various FPGA families like Artix-7, Spartan-6 and Virtex-7. Synthesis and Simulation results are verified through VHDL codes using Xilinx ISE 14.7 software. Synopsis tool in 180 mm technology is used for calculating the power consumption of Single router. The time delay for Artix-7 is found to be much lesser than Spartan-6 and Virtex-7. Based on the experimental results it is evident that the power consumption is less for 50MHz. On the other hand, the power consumption is more for higher frequencies.

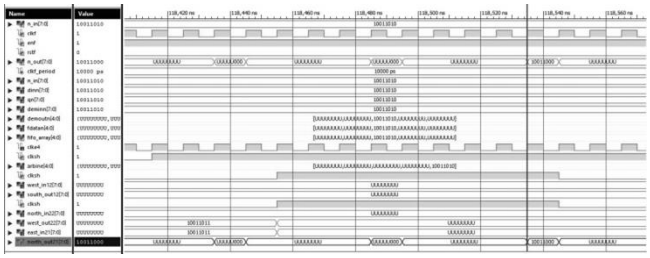

Fig.12. Router simulation Results in 2×2 Mesh topology

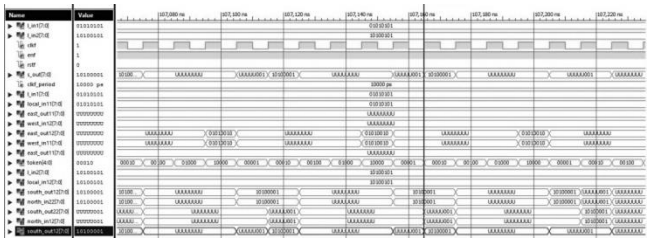

Fig.13. Router simulation Results in 2×2 Torus topology

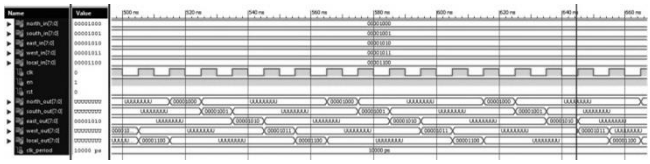

Fig.14. Single Router simulation Result

ACKNOWLEDGEMENT

The authors would like to thank Facility for Advanced Computing, testing and Simulation of Electronic circuits (FACTS ElCi) for providing lab facility for our research work.

REFERENCES

[1] Konstantinos Tatas, Kostas Siozios, Dimitrios Soudris and Axel Jantsch. "*Designing 2D and 3D Network-on-chip Architectures*", Springer, 2014.

[2] William J Dally and Brian Towles, "Route Packets, not Wires: On-Chip Interconnection Networks", *Proceedings of Design Automation Conference*, pp. 684-690, 2001.

[3] Masoud Daneshtalab, "Exploring Adaptive Implementation of On-Chip Networks", Ph.D Dissertation, Department of Information Technology, University of Turku, 2011.

[4] Tobias Bjerregaard, and Shankar Mahadevan. "A Survey of Research and Practices of Network-on-Chip", ACM Computing Surveys, Vol. 38, No. 1, pp. 1-51, 2006.

[5] Luca Benini and Giovanni De Micheli, "Networks on Chips: A New Paradigm for Systems on Chip Design", *Proceedings of Design, Automation and Test in Europe Conference and Exhibition*, pp. 1-2, 2002.

[6] S. Swapna, Ayas Kanta Swain and Kamala Kanta Mahapatra, "Design and Analysis of Five Port Router for Network on Chip", *Proceedings of IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics*, pp. 51-54, 2012.

[7] Maurizio Palesi and Masoud Daneshtalab, "*Routing Algorithms in Networks-on- Chip*", Springer, 2014.

[8] T.A. Bartic, J.Y. Mignolet, Vincent Nollet, Theodore Marescaux, Diederik Verkest, Serge Vernalde and Rudy Lauwereins, "Topology Adaptive Network-on-Chip Design and Implementation", *IEEE Proceedings-Computers and Digital Techniques*, Vol. 152, No. 4, pp. 467-472, 2005.

[9] Wen-Chung Tsai, Ying-Cherng Lan, Yu-Hen Hu and Sao-Jie Chen, "Networks on Chips: Structure and Design Methodologies", *Journal of Electrical and Computer Engineering - Special Issue on Networks-on-Chip: Architectures, Design Methodologies, and Case Studies*, Vol. 2, No. 2, pp. 1-16, 2012.

[10] Aline Vieira de Mello, Luciano Copello Ost, Fernando Gehm Moraes and Ney Laert Vilar Calazans. "Evaluation of Routing Algorithms on Mesh Based NoCs", Technical Report Series, Faculdade de Informatica, pp. 1-11, 2004.

[11] Suyog K Dahule and M. A. Gaikwad, "Design and Simulation of Round Robin Arbiter for NoC Architecture", *International Journal of Engineering and Advanced Technology*, pp. 2249-8958, 2012.

[12] Yanhua Liu, Jie Jin and Zongsheng Lai, "A Dynamic Adaptive arbiter for Network-on-Chip", *Journal of*

Microelectronics, Electronic Components and Materials, Vol. 43, No. 2, pp. 111-118, 2013.

[13] Ville Rantala, Teijo Lehtonen and Juha Plosila, "Network on Chip Routing Algorithms", Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.8910&rep=rep1&type=pdf.

[14] Rickard Holsmark, Maurizio Palesi and Shashi Kumar, "Deadlock Free Routing Algorithms for Irregular Mesh Topology NoC Systems with Rectangular Regions", *Journal of Systems Architecture*, Vol. 54, No. 3-4, pp. 427-440, 2008.

[15] Yaniv Ben-Itzhak, Israel Cidon, Avinoam Kolodny, Michael Shabun and Nir Shmuel, "Heterogeneous NoC Router Architecture", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 9, pp. 2479-2492, 2015.