DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

EFFICIENT ROUTER DESIGN FOR NETWORK ON CHIP

SWAPNA S

2013

EFFICIENT ROUTER DESIGN FOR NETWORK ON CHIP

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

In

VLSI DESIGN AND EMBEDDED SYSTEM

By

SWAPNA S

Department of Electronics and Communication Engineering

National Institute Of Technology

Rourkela

2011 – 2013

EFFICIENT ROUTER DESIGN FOR NETWORK ON CHIP

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology
In
VLSI DESIGN AND EMBEDDED SYSTEM

by

SWAPNA S

211EC2122

Under the Guidance of
Prof. Ayas Kanta Swain

Department of Electronics and Communication Engineering

National Institute Of Technology

Rourkela

2011 – 2013

Dedicated to my parents

DEPARTMENT OF ELECTRONICS AND
COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA
ORISSA, INDIA-769008

CERTIFICATE

This is to certify that the Thesis Report entitled "**EFFICIENT ROUTER DESIGN FOR NETWORK ON CHIP**", submitted by **SWAPNA S bearing roll no. 211EC2122** in partial fulfilment of the requirements for the award of **Master of Technology in Electronics and Communication Engineering** with specialization in "**VLSI Design and Embedded Systems**" during session 2011 - 2013 at National Institute of Technology, Rourkela is an authentic work carried out by her under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any Degree or Diploma.

Place: Rourkela
Date:

Prof. AYAS KANTA SWAIN
Dept. of E.C.E
National Institute of Technology
Rourkela – 769008

ACKNOWLEDGEMENTS

With deep regards and profound respect, I avail this opportunity to express my deep sense of gratitude and indebtedness to Prof. Ayas Kanta Swain, Department of Electronics and Communication Engineering, NIT Rourkela for his valuable guidance and support. I am deeply indebted for the valuable discussions at each phase of the project. I consider it my good fortune to have got an opportunity to work with such a wonderful person.

Sincere thanks to Prof. K. K. Mahapatra, Prof. S. Meher, Prof. S. K. Behera, Prof. S. K. Das, Prof. Samit Ari, Prof. A. K. Sahoo and Prof.Poonam Singh for teaching me and for their constant feedbacks and encouragements. I would like to thank all faculty members and staff of the Department of Electronics and Communication Engineering, NIT Rourkela for their generous help.

I would like to thank all my friends, especially my classmates without whom the project would not have reached its completion.

Last but not least I also convey my deepest gratitude to my parents and family for whose faith, patience and teaching had always inspired me to walk upright in my life.

Finally, I humbly bow my head with utmost gratitude before the God Almighty who always showed me a path to go and without whom I could not have done any of these.

ABSTRACT

A Network-on-chip (NoC) is a new paradigm in complex system-on-chip (SoC) designs that provide efficient on chip communication networks. It allows scalable communication and allows decoupling of communication and computation. The data is routed through the networks in terms of packets. The routing of data is mainly done by routers. So the architecture of router must be an efficient one with a lower latency and higher throughput. In this project we designed, implemented and analyzed three different router architectures for a network on chip communication. The routers have five ports, four ports connected to other ports in four different directions and the fifth port connected to the processing element through a network interface. The first architecture is a basic router with demultiplexer and scheduler. The second architecture consists of crossbar switch and arbiter. The third architecture uses the CDMA technology that is popular in wireless communication. The three architectures were analyzed for their performance in terms of delay, throughput and latency and we concluded that CDMA router performs better than the other two.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

NoC	Network-on-Chip
SoC	System-on-Chip
CDMA	Code Division Multiple Access
IC	Integrated Circuit
CMOS	Complementary Metal Oxide Semiconductor
ITRS	International Technology Roadmap for Semiconductors
MAC	Medium Access Control
PCI	Peripheral Component Interconnect
IP	Intellectual Property
FPGA	Field Programmable Gate Array
VLSI	Very Large Scale Integration
ASIC	Application Specific Integrated Circuit
NI	Network Interface
OSI	Operation Systems Interconnect
FIFO	First In First Out
RAM	Random Access Memory
RR	Round Robin
FCFS	First Come First Serve
TX	Transmitter
MOD	Modulator

CHAPTER 1

INTRODUCTION AND OVERVIEW

1.1 INTRODUCTION

A billion transistors, one million gates, thousands of circuits, hundreds of designs on a single IC chip; such intricate designs pose innumerable challenges to IC designers. The most successful IC designers overcome all such challenges to provide functionally correct and reliable operation of the IC's. As the integration increases the cost effectiveness is also a major area of concern in IC designs.

Reduced cost is one of the big attractions of integrated electronics, the cost advantage continues to increase with the evolution of technology toward the production of larger and larger circuit functions on a single semiconductor substrate; proposed by Gordon E Moore in his paper Cramming more components onto integrated circuits, 1965, the paper in which he proposed the well-known Moore's law[1]. By 2025 the physical dimension of CMOS transistors are expected to cross the 10 nm threshold, according to a 2012 report of International Technology Roadmap for Semiconductors[2]. The graph below shows the transistor integration on a single chip over the past two decades. Now two billion transistors are integrated onto a single chip.

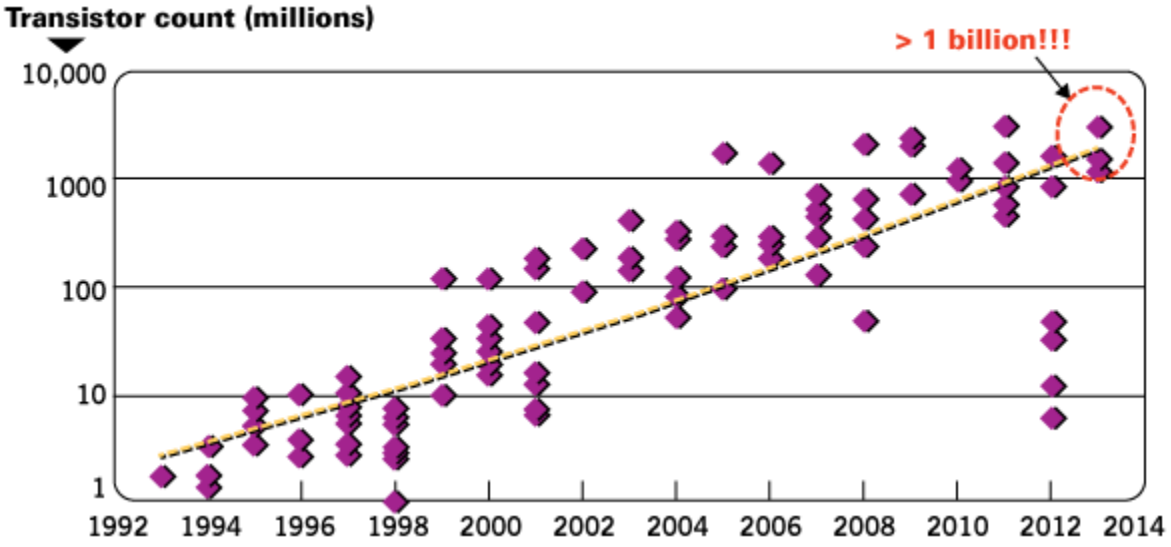

Figure 1.1. Evolution of transistor integration on a chip

It is to keep pace with such intricate levels of integration that the design engineers have come up with a new design methodology called System-on-Chip (SoC)

The SoC is a technology where maximum technology is crammed into the smallest possible space. The design of system on a chip is impacted strongly by the so called intellectual property (IP) core. An integrated circuit core is a predesigned, preverified silicon circuit block[3]. The core usually contain at least 5,000 gates, that can be used in building a larger or more complex application on a semiconductor chip. Examples of cores are memory controllers, processors, or peripheral devices such as MAC Ethernet or PCI bus controllers. In the semiconductor industry IP core is the property of any single vendor. The ip cores are the building blocks of various system on chip designs for implementing larger and complex embedded system applications. The system on chip can hold hardwares like processors, memories peripherals, controllers, digital signal processors and various custom logic blocks and softwares for controlling the hardwares.

The main advantage of system on chip is low power consumption, lower cost and higher reliability than the multi-chip systems it has replaced. But the transition to system on chip technology was faced with many challenges. Firstly, scalability of the system. It is really an enormous task to scale down large computer systems to the size of silicon die. The physical dimensions of various components, their inductive and capacitive effects on other components need to be taken care of. Secondly, it is difficult to maintain global synchronization as different systems will be using different clock signals. Thirdly, the heterogeneity of the whole package where different systems with different library files, packages, coding languages and dimensions need to be packaged together. Fourthly, the issues in interconnection of the systems[4].

Cores do not make up SoCs alone. The cores must include an interconnection architecture and interfaces to peripheral devices. The interconnection architecture consists of physical interfaces and communication mechanisms. This allows the communication between SoC components to take place.

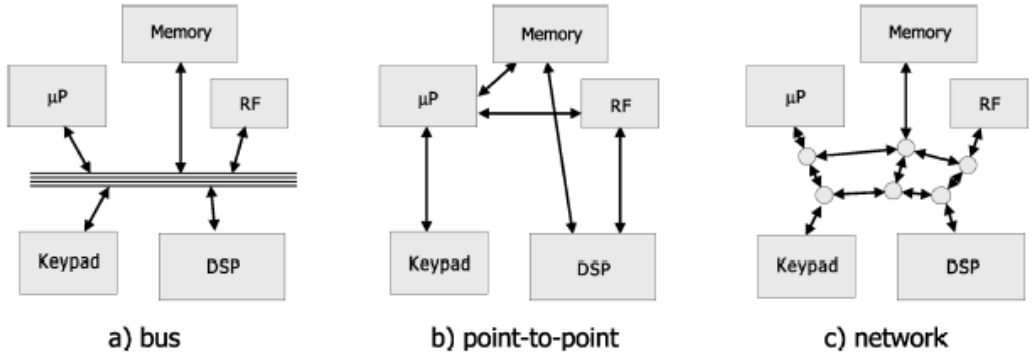

Fig. 1.2 Communication structures in SoC a) traditional bus based communication, b)dedicated point to point links c)network on a chip [5]

Usually, the interconnection architecture is based on dedicated wires or shared busses. If a system has a limited number of cores then dedicated wire architecture is effective. As the system complexity grows the number of wires around the core also increases. Therefore, dedicated wires have poor reusability and flexibility. A shared bus is a set of wires which is common to multiple cores. The approach of shared bus is more flexible and is totally reusable, but it allows only one communication transaction at a time, all cores share the same communication bandwidth in the system and its scalability is limited to few dozen IP cores. Thus scalability is a major problem with buses. It is the issues in interconnection that paved the way for new paradigm in communication called the Network-on-chip (NoC). NoC architecture has been proposed as a high performance, scalable and power efficient alternative to the bus based architecture. It solves the scalability problem by supporting multiple concurrent connections with various systems. As system becomes more complex, more and more integration is possible to the

existing system with ease without any constraints. It can reduce the wire routing congestion to a great extent. The systems that are interconnected with a network on chip can be easily interchanged with other systems with any ip cores of any vendor available in the market. The NoC separates the communication part from the computation part for system simplicity and is ideally suited for integrated systems. NoC can take care of the communication part with utmost ease without any interference in the computation part [6].

After the configuration and interconnection the other major area of concern in the SoC design is the implementation. Nowadays the implementation is done using Field Programmable Gate Arrays (FPGAs). The advantages of FPGAs are the lower time to market, lower development cost, less manufacturing steps and highly suitable for research activities.

1.2 NETWORK ON CHIP (NoC)

NoC is a technology that is intended to solve the short coming of buses. It is an approach to design the communication subsystem between intellectual property cores in a SoC design. The communication strategy in system on chip uses dedicated buses between communicating resources. This will not give any flexibility for the needs of the communication, in each case, have to be thought of every time a design is made. Another possibility is the use of common buses, which have the problem that it does not scale very well, as the number of resources grows. NoC is intended to solve the shortcomings of these, by implementing a communication network of switches/micro routers and resources [7,8]. The NoC design paradigm has been proposed as the future of ASIC design [9]. The major driving force behind the transition to NoC based solutions is the inadequacy of current day VLSI inter-chip communication design methodology for the deep sub-micron chip manufacturing technology [10]. The negative effect of technology scaling on global interconnects, increased dependence on fault-tolerant mechanisms as feature

size reduces, increasing use of parallel architectures are the reasons why NoC is becoming popular. The NoC based system on chips imposes various design issues on the fabrication of such integrated chips. Firstly, the suitable topology for the target NoCs such that the performance requirements and design constraints are satisfied. Secondly, the design of network interfaces to access the on chip network and routers provide the physical interconnection mechanisms to transport data between processing cores. Thirdly, the selection of communication protocols which are suitable for on chip interconnection networks. Finally, as technology scales and switching speed increases, future network on chips will become more sensitive and prone to errors and faults. Fault tolerance is becoming critical for on chip communications [11]. Today's SoCs need a network on chip IP interconnect fabric to reduce wire routing congestion, to ease timing closure, for higher operating frequencies and to change IP easily. Network on chips are a critical technology that will enable the success of future system on chips for embedded applications. This technology of network on chip is expected to dominate computing platforms in the near future.

NoC design space is considerably larger when compared to a bus based solution, as different routing and arbitration strategies can be implemented as well as different organizations of the communication infrastructure. The NoC paradigm is highly suited to provide SoC platforms scalable and adaptable over several technology generations. NoC platforms may allow the design productivity to grow as fast as technology capabilities and may eventually close the design productivity gap [12]. In addition, NoCs have an inherent redundancy that helps tolerate faults and deal with communication bottlenecks [13].

1.3 NoC ARCHITECTURE

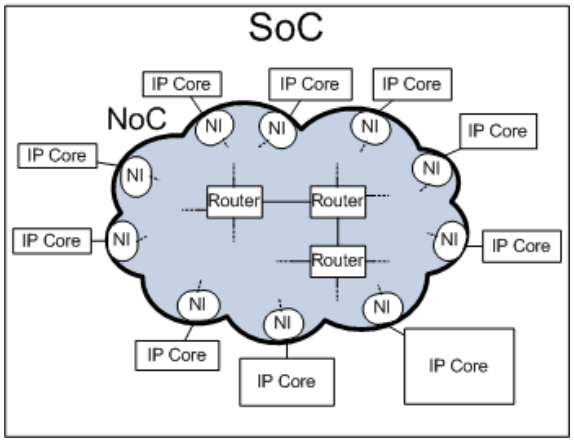

Figure 1.3. General Architecture of NoC

The figure 1.3 shows the general architecture of NoC. Routers, network interface (NI) and links are the main components of a NoC architecture [14]. A router is responsible for routing information from a source port to its destination port. The network interface separates the computation part from the communication part and acts as a mediator between the router and the processing element. A link connects different routers in the network according to the chosen topology.

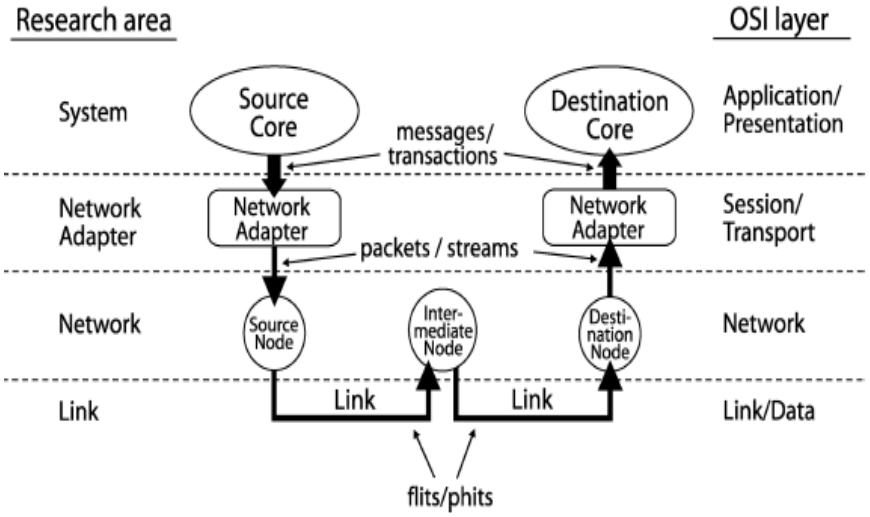

Figure 1.4. Flow of data packets from source node to destination node Links [5]

1.4 LINKS

Links are used to transmit packet between routers. It physically connects the nodes and enables the communication in the network. It consists of a set of wires that connect the routers in the network. A NoC link has two physical channels making a full-duplex connection between the routers. The number of wires per channel is uniform throughout the network and it is known as channel bit width. But if the links become too long then it can cause wiring delay in the network. To overcome such problem the NoC pipelines long wires in interconnects by partitioning the wires into smaller segments [15].

1.5 NETWORK INTERFACE

The network interface or network adapter makes the logical connection between the IP core and the network. It can be divided into two parts: a front end and back end. The requests from the IP core are handled by the front portion and it is unaware of the existence of the network. The back end part is connected directly to the network which handles the network protocol, ordering and reordering the packets, buffers and help the router in terms of storage.

1.6 ROUTER

A router is the most important component in a NoC [16]. It is the communication backbone of a NoC system. So it should be designed for maximum efficiency and throughput. A router is used in a network for directing the traffic from source to destination. It coordinates the data flow which is very crucial in communication networks.

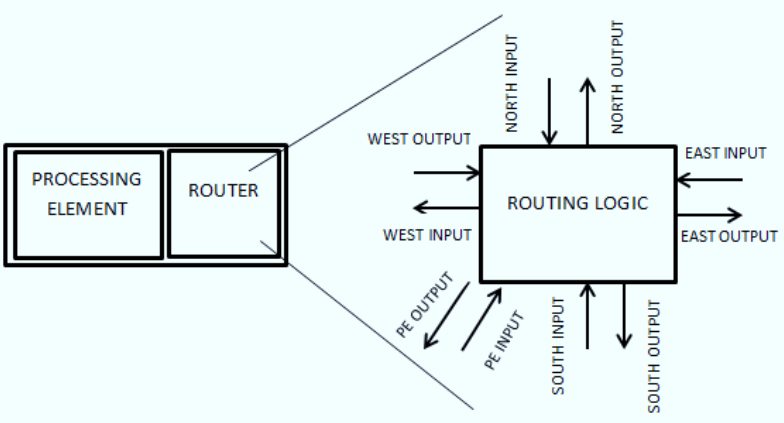

Figure 1.5 A typical routing node

The architecture of a router consists of an input port, an output port, a switching matrix to connect the input port to the output port, and a local port that connects the router to the corresponding IP core. Routers are intelligent devices that receive incoming data packets, inspect their destination and figure out the best path for the data to move from source to destination. A router's architecture determines its critical path delay which affects per hop delay and network latency [17]. So the design of the router should be such that it meets the required latency and throughput requirements amidst tight area and power constraints. The design efficiency of the router determines the performance of the network. A router decodes the information provided by the incoming message based on the routing function and destination of the message. A router is built according to the OSI model of NoC. Each layer has its own specific functions to perform [18].

1.7 NETWORK TOPOLOGY

The physical layout and connections between nodes and channels in the network is determined by the on chip network topology. A topology determines the number of hops (or routers) a message must traverse as well as the interconnect lengths between hops, thus

influencing network latency significantly. As traversing routers and links incurs energy, a topology's effect on hop count also directly affects network energy consumption. Furthermore, the topology dictates the total number of alternate paths between nodes, affecting how well the network can spread out traffic and hence support bandwidth requirements. The first decision designers have to make when building an on-chip network is, frequently, the choice of the topology. Figure 1.6 shows the commonly-used on-chip topologies [19].

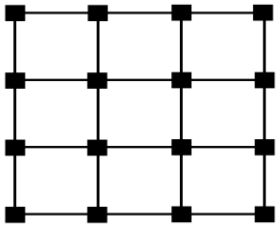

Figure 1.6. (a) Mesh Network

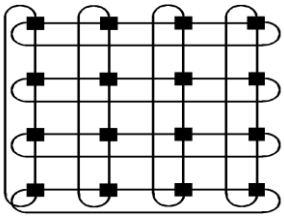

Figure 1.6. (b) Torus Network

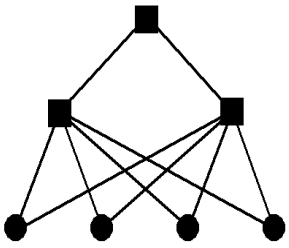

Figure 1.6. (c) Fat Tree Network

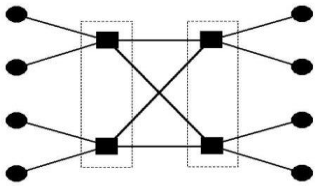

Figure 1.6 (d) Butterfly Network

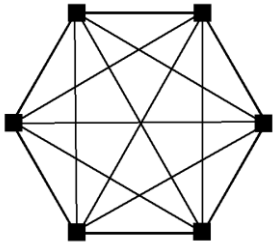

Figure 1.6 (e) Polygon (Hexagon) Network

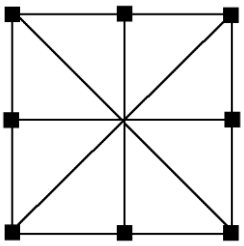

Figure 1.6 (f) Spidergon Network

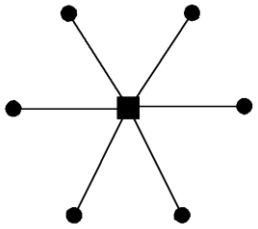

Figure 1.6 (g) Star Network

1.7.1 MESH

Mesh topology is favored by many research groups because of its layout efficiency. It has good electrical property and can address the on-chip resources in a simple manner [10]. A mesh-shaped network consists of m columns and n rows. The routers are situated in the intersections of the two wires and the computational resources are near routers. Addresses of routers and resources can be easily defined as x-y coordinates in mesh. The regular mesh network is also called as Manhattan Street network. A mesh-shaped network consists of m columns and n rows. The routers are situated in the intersections of the two wires and the computational resources are near routers. Addresses of routers and resources can be easily defined as x-y coordinates in mesh [20].

1.7.2 TORUS

A Torus network is an improved version of basic mesh network. A simple torus network is a mesh in which the heads of the columns are connected to the tails of the columns and the left sides of the rows are connected to the right sides of the rows. The path diversity of torus is better than the mesh network, and it also has more minimal routes.

1.7.3 TREE

In a tree topology nodes are routers and leaves are computational resources. The routers above a leaf are called as leaf's ancestors and correspondingly the leaves below the ancestor are its children. In a fat tree topology each node has replicated ancestors which mean that there are many alternative routes between nodes.

1.7.4 BUTTERFLY

A butterfly network is uni- or bidirectional and butterfly-shaped network typically uses a deterministic routing. Packets arriving to the inputs on the left side of the network are routed to the correct output on the right side of the network.

1.7.5 POLYGON

The simplest polygon network is a circular network where packets travel in loop from router to other. The network becomes more diverse when chords are added to the circle. When there are chords only between opposite routers, the topology is called as spidergon.

1.7.6 STAR

A star network consists of a central router in the middle of the star, and computational resources or sub networks in the spikes of the star. The capacity requirements of the central router are quite large, because all the traffic between the spikes goes through the central router. That causes a remarkable possibility of congestion in the middle of the star.

1.8 METRICS FOR COMPARING TOPOLOGIES:

Degree: The degree of a topology refers to the number of links at each node. For instance, for the topologies in figure 1.6, a ring topology has a degree of 2 since there are two links at each node, while a torus has a degree of 4 as each node has 4 links connecting it to 4 neighbouring nodes. A higher degree requires more ports at routers, which increases implementation complexity.

Hop count: The number of hops a message takes from source to destination, or the number of links it traverses, defines hop count. This is a very simple and useful proxy for network latency, since every node and link incurs some propagation delay, even when there is no contention. The maximum hop count is given by the diameter of the network; For instance, in the figure 1.6 shown, assuming bidirectional links and shortest-path routing, the maximum hop count of the ring is four, that of a mesh is also four, while a torus improves the hop count to two.

Maximum channel load: This metric is useful as a proxy for estimating the maximum bandwidth the network can support, or the maximum number of bits per second (bps) that can be injected by every node into the network before it saturates. Here, it is defined as being relative to the injection bandwidth. So, when the load on a channel is 2, it means that the channel is loaded with twice the injection bandwidth [21].

1.9 THESIS GOALS

Network on chip is an active research field. Many aspects of NoC still need further exploration and understanding. In this thesis our main concentration is to implement an efficient router for network on chip applications. The router is the main component that determines the latency, throughput, reliability and efficiency of the entire network on chip design.

1.10 OBJECTIVES

- To develop a deep understanding of FPGA-based NoC architectures and parameters to provide further exploration of their design space
- To design an efficient and reliable router which is an important part of the entire NoC
- The performance analysis of the routers are done to develop a deep understanding of the architectures

1.11 THESIS OUTLINE

The general approach of the thesis is to go through the different architectures for routers in network on chip. The thesis is organized as follows:

Chapter 2 discusses the router architecture I which is the basic router with a demux and a scheduler along with its simulation results.

Chapter 3 discusses the router architecture II, the router with a crossbar switch and arbiter and its simulation results.

Chapter 4 discusses the router architecture III (CDMA router) along with a brief introduction on CDMA technique and its simulation and synthesis results.

Chapter 5 compares the performance of the three architectures with each other.

Chapter 6 concludes all the contributions of the thesis with a brief on future research plans.

CHAPTER 2

DESIGN OF A FIFO BASED ROUTER

2.1 FIFO BASED ROUTER ARCHITECTURE

Figure 2.1 shows the entity of the designed router. It consists of five data input ports (datai1, datai2, datai3, datai4, datai5), five data output ports (datao1, datao2, datao3, datao4, datao5), five packet available indicators (Wr1, Wr2, Wr3, Wr4, Wr5), a clock (Clock) and a reset(Reset) signal.

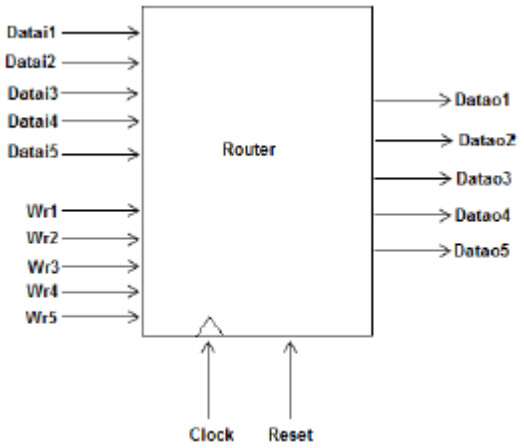

Figure 2.1. The router ports for FIFO based router

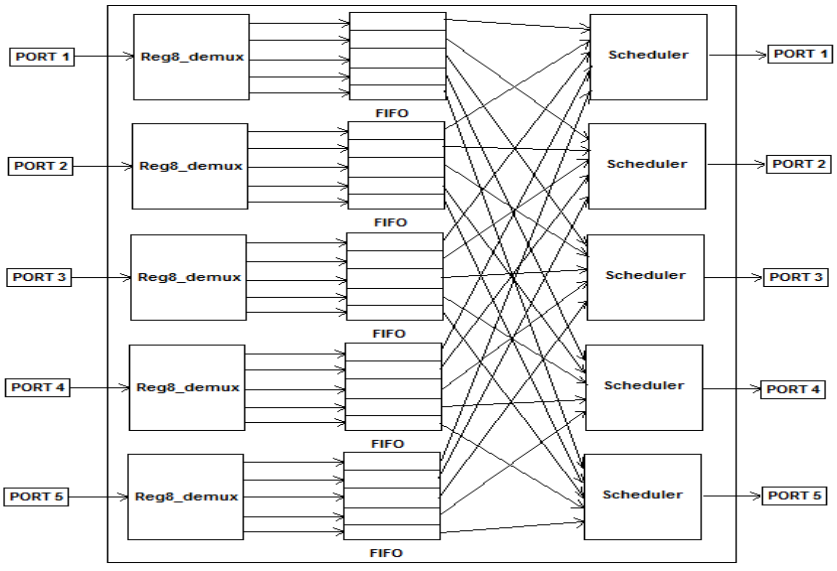

Figure 2.2. The FIFO based router architecture

Figure 2.2 shows the internal architecture of the designed router. The building blocks of router consist of mainly three parts:

1. Registers and demultiplexers
2. First In First Out Registers
3. Schedulers.

2.2 REGISTER AND DEMULTIPLEXER

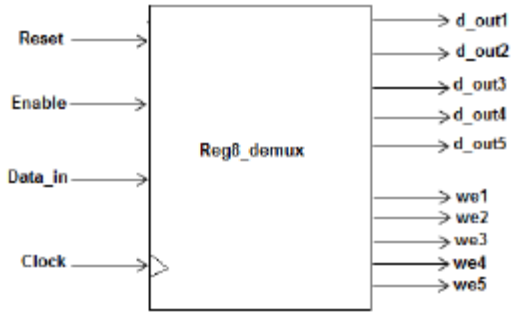

Figure 2.3. Combined register and demultiplexer

Figure 2.3 shows the combined register and demux design. The register is used for storing the input data in the form of a buffer. The demultiplexer will direct the input to the appropriate output port.

1) 8-bit register: The register has a positive edge clock, an active high clock enable and an active high asynchronous reset. The output of the register is the input of the demultiplexer. The data input to the register is transferred to the output port at the positive edge of the clock if and only if the enable is 1 and the reset is 0. If the reset is 1, then the output port of the register is set to zeros. If the enable is 0, then the output port keeps its current value.

2) 1-to-8 8-bit demultiplexer: The demultiplexer directs the input to the proper output port according to the select signal. The select signal is taken as the first three bits of the input data. The demultiplexer also has an enable. If this enable is set to 1, then the input data is transferred to the appropriate output port and the corresponding write enable is set to 1, while the other output ports and write enables are set to zeros. If the enable signal is 0, then the output ports and the write enables are all set to zeros.

2.3 FIFO UNIT

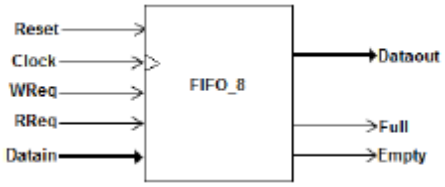

Figure 2.4 FIFO Unit

The FIFO unit, as shown in figure 2.4 consists of two parts; RAM memory and FIFO Control. The FIFO receives read and write requests from RReq and WReq signals respectively. When the FIFO is full, write operations are disabled and when it is empty, read operations are disabled. The FIFO empty flag is set to high when the FIFO is empty and full flag is set to high when the FIFO is full.

1) RAM: The write and read operations in the memory are synchronized with the memory clock input (Clk). If (Clk) is rising and (Wr Enable) is 1, then the input data word (D in) is written to the memory location with address Addr. If (Clk) is rising and (Rd Enable) is 1, then the output data word (D out) is read from the memory location with the address (Addr). The input bus (Datain) of the FIFO is the input bus of the RAM (D in), while the output bus (Dataout) of the FIFO is the output bus of the RAM (D out). The width of all the data ports is

eight bits. When the memory is reset asynchronously by the reset signal (Rst), all its locations become zeros and its output (D out) is set to zeros as well.

2) FIFO Controller: The FIFO controller receives read and write requests from the (R Req) and (W Req) signals respectively. It checks the validity of the read or write operations and generates valid signals on (Read En) or (Write En) ports. Then it outputs the corresponding read or write address on (Add Output). The (Read En) and (Write En) ports are connected to the (Rd Enable) and (Wr Enable) ports of the RAM, respectively. The (Add Output) port is connected to the (Addr) port of the RAM.

2.4 SCHEDULER

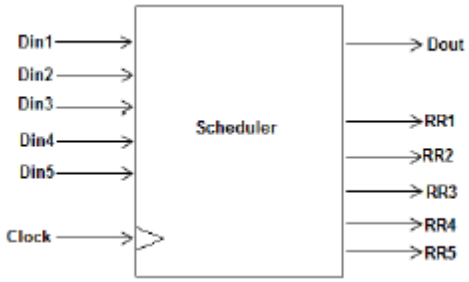

Figure 2.5. Scheduler

The scheduler used here (figure 2.5) is a round robin scheduler which uses the round robin algorithm. It assumes that all data are equally valid for selection. The algorithm lets every active data flow that has data packets in the queue to take turns in transferring packets on a shared channel in a periodically repeated order. The port through which the data comes at the present instance should have the lowest priority at the next round of scheduling. The outputs RR1, RR2, RR3, RR4 and RR5 are used to indicate the next port to be read and they are further connected to the read request ports in the corresponding FIFO.

2.5 ROUND ROBIN SCHEDULING ALGORITHM

Round Robin (RR) is one of the oldest, simplest and fairest and most widely used scheduling algorithms. It is designed especially for time sharing systems. It is similar to first come first serve (FCFS) but preemption is also added to it. It is widely used in NoC because of its simplicity and practicability [23].

2.6 SIMULATION RESULTS

Figure 2.6. Simulation waveform of router architecture I

Figure 2.6 shows the simulation result of five port router. The reset (rst) signal must be kept low during normal operations. The write enable (wr) signals, when high, will enable the corresponding demultiplexers. The input signals (datai) are the data packets given to the router. The routing pattern is observed from the output signals (datao). The output signals follow the round robin scheduling algorithm. The data at the input ports is directed to the output port depending on the first three bits of the input data which act as the select lines of the demultiplexer. The input data 68 from port 1 is directed towards port 4 of the output since the first three bits of the data is 100. At that instance the round robin scheduling signal of r2 will be high which indicates the port to be read in the next instance. Likewise the other data packets are also routed. From the simulation it is also observed that, depending on the amount of input data packets the router will have a delay before the output data becomes error free.

2.7 SYNTHESIS RESULTS

The synthesis result of router architecture I is as given in table 2.1. The synthesis result provides a summary and analysis of netlist generation of the design. The device used for implementation is Spartan 3E with a speed grade of -4 and the results given are obtained after the FPGA implementation.

TABLE 2.1. PERCENTAGE OF RESOURCE UTILIZATION OF FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

RESOURCE	**USED**	**AVAILABLE**	**UTILIZATION PERCENTAGE**
Slices	1464	4656	31
4 input LUTs	1901	9312	20
Bonded IOBs	87	232	37
GCLK	1	24	4

2.8 TIMING SUMMARY

The timing summary of the router architecture I design is given in table 2.2.

TABLE 2.2. TIMING SUMMARY OF PROPOSED ARCHITECTURE I ON FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

Maximum frequency	144.196 MHz
Minimum period	6.935 ns
Minimum input arrival time before clock	7.599 ns
Maximum output required time after clock	4.283 ns

CHAPTER 3

DESIGN OF A CROSSBAR BASED ROUTER

3.1 CROSSBAR BASED ROUTER ARCHITECTURE

The router architecture consists of fifo, crossbar switch and arbiter. This architecture has less number of fifo buffers compared to the previous architecture. So it takes very less area. Each block of the architecture is as explained below.

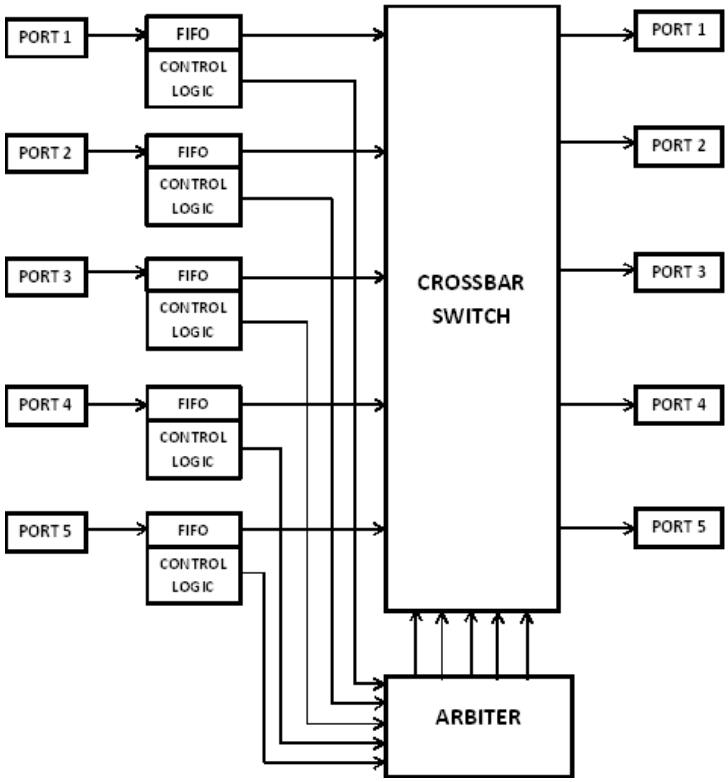

Figure 3.1. Router architecture II

3.2 FIFO BUFFER

In this architecture the packets in transit are stored in a buffer. Each input channel has a FIFO buffer of depth 4 and 8 bit data width and control logic. The FIFO controller receives the packet from the output port of the adjacent router, stores them in buffer and manages the flow

control between adjacent routers. The complete transmission of data occurs when the buffer of that channel is not full. The output of the FIFO buffer is given to the crossbar switch that switches the data to the corresponding output port. The FIFO controller selects the last three bits of the input data and that is given to the arbiter which will grant the data depending on the various signals.

3.3 CROSSBAR

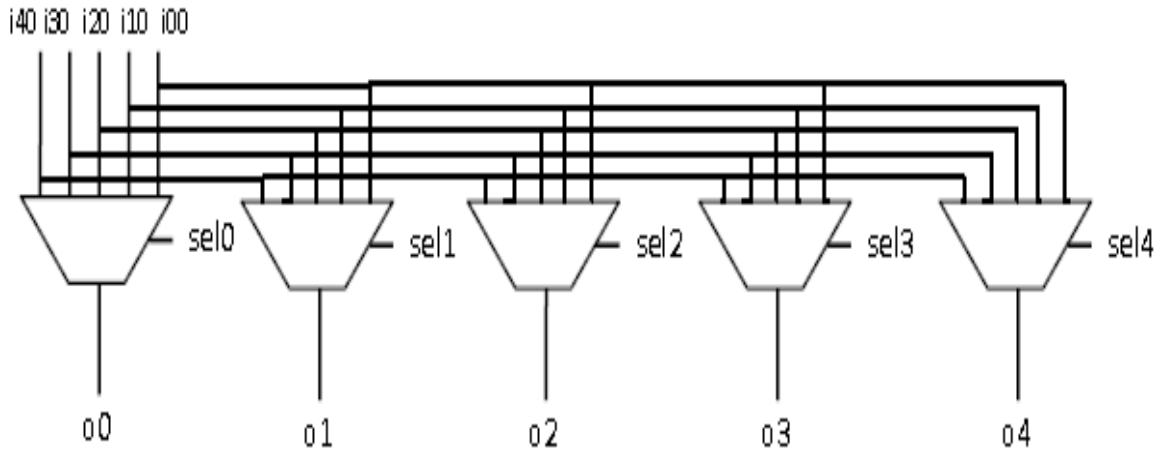

Figure 3.2. Internal structure of crossbar

The crossbar switch of a router is the heart of the router datapath. It switches the data from the input port to the output port doing the essence of the router function. The internal structure of a crossbar consists of an array of multiplexers. In this architecture it consists of five 5:1 multiplexers. The five multiplexers have five forty bit data as input. All the five inputs are connected to all the five multiplexers. The data to be forwarded to the output depends on the select lines. The select lines are generated by the arbiter depending on the request signals. There are five select lines of three bit width to select one of the five ports. So the output of each multiplexer depends on the select line of that multiplexer.

To design a crossbar that switches at high frequency and low power is one of the main challenges of router design. The crossbar can be made to work faster by clocking it at a higher frequency than the rest of the router design. By making the speed of the crossbar faster more data packets can be sent.

3.4 ARBITER

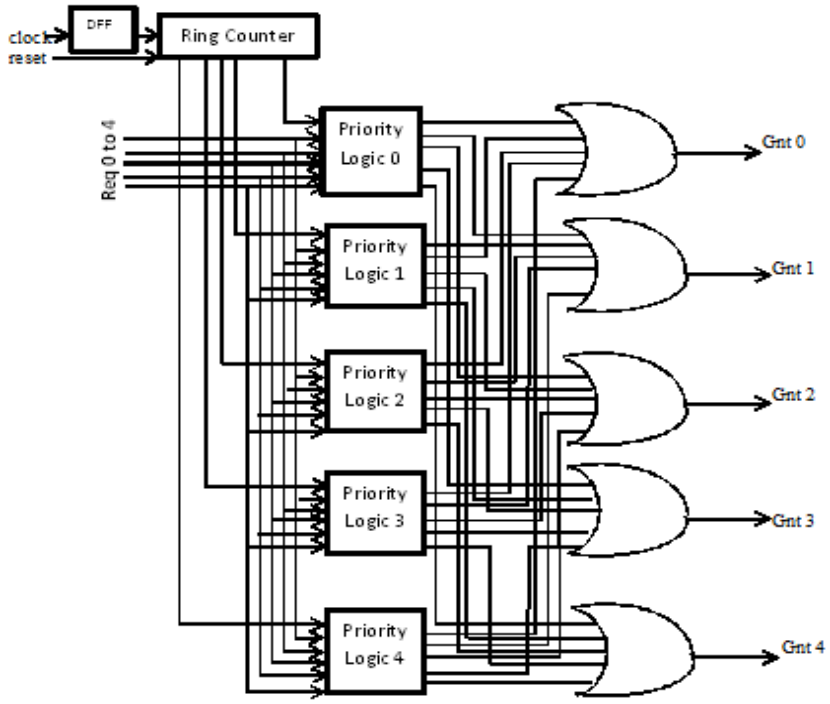

Figure 3.3. Internal structure of arbiter

The arbiter is for controlling the arbitration of ports and to resolve the contention issues [24]. It knows the current status of all the ports, which ports are free, which ports are communicating with each other and in which ports the data contention can occur. Packets of same priority and destined for the same port is scheduled by a round robin algorithm. The arbiter can release the output port which is connected to the crossbar once it finishes the data

transmission in that particular port. Then the port will be assigned to the next awaiting port in the queue. The arbiter generates an output signal of three bit which is given to the select line of the crossbar for selecting the corresponding port. It is these three bits that determines the data that comes in the output port of the router.

The internal structure of the arbiter has a D flip flop, ring counter, five priority logic blocks and five input OR gates. In logic designs flip flops are used to create simple finite state machines. The ring counter is a counter where simple shift registers are connected in cascade to each other. The output of the last flip flop is connected as the input to the first flip flop. The data pattern will circulate as long as the clock pulses are applied. In the arbiter it is the ring counter that selects the incoming requests in round robin way and gives it to the priority logic block. The functionality of the priority logic block is similar to a priority encoder without output encoding. Since each block has different order of inputs the priority of chosen signals varies with the chosen block. With a round-robin arbiter, the last request to be serviced will have the lowest priority in the next round of arbitration.

3.5 SIMILATION RESULTS

Figure 3.4 (a) shows the input waveform given to the router entity. The reset must be kept low for normal operations. There are five enable signals for each input and the enable signals must be kept high. Figure 3.4 (a) shows the input data. The output data is obtained after some time delay. For the data contention where all the ports are trying to pass data to a single port round robin algorithm is used as shown in figure 3.4 (b).

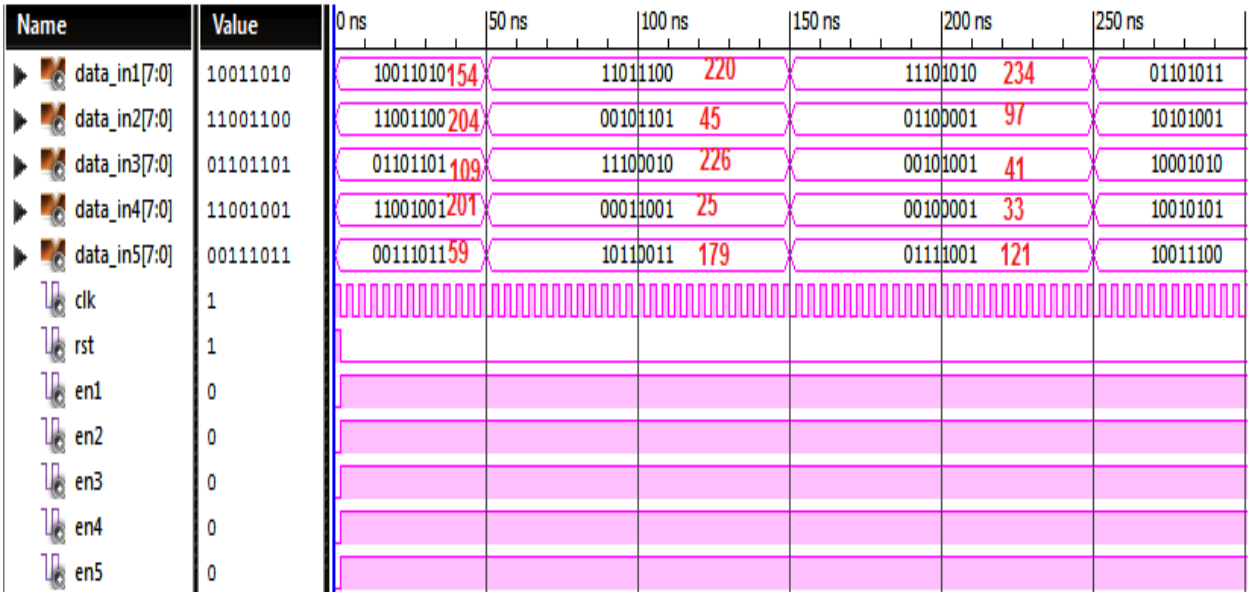

Figure 3.4. (a) Simulation waveform of Router Architecture – II of input

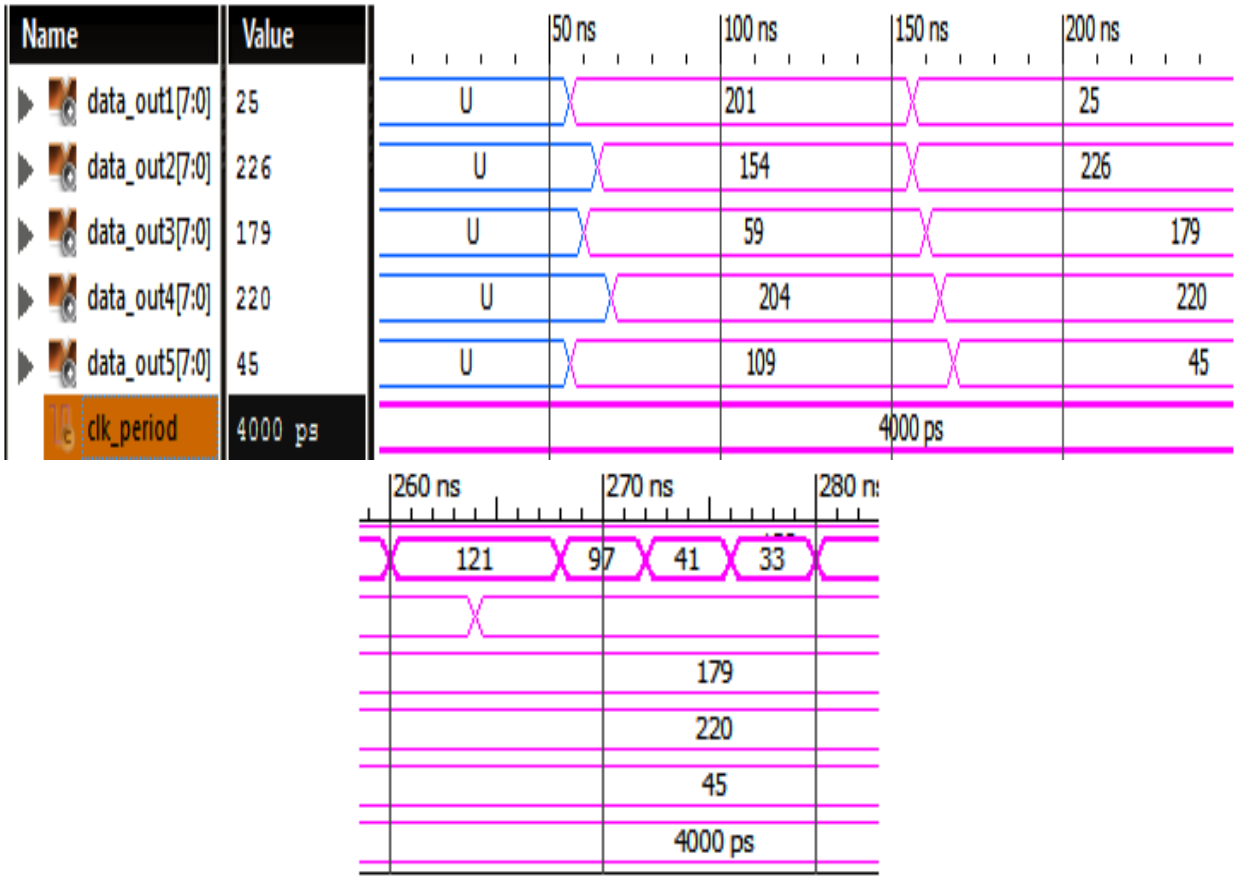

Figure 3.4. (b) Simulation waveform of router architecture – II of outputs

3.6 SYNTHESIS RESULTS

The synthesis result of router architecture II is as given in table 3.1. The synthesis result provides a summary and analysis of netlist generation of the design. The device used for implementation is Spartan 3E with a speed grade of -4 and the results given are obtained after the FPGA implementation.

TABLE 3.1. PERCENTAGE OF RESOURCE UTILIZATION OF FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

RESOURCE	**USED**	**AVAILABLE**	**UTILIZATION PERCENTAGE**
Slices	418	4656	8
4 input LUTs	2007	9312	21
Bonded IOBs	87	232	37
GCLK	6	24	25

3.7 TIMING SUMMARY

The timing summary of the router architecture I design is given in table 3.2.

TABLE 3.2. PERCENTAGE OF RESOURCE UTILIZATION OF FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

Maximum frequency	212.089 MHz
Minimum period	4.715 ns
Minimum input arrival time before clock	12.909 ns
Maximum output required time after clock	4.368 ns

CHAPTER 4

DESIGN OF A CDMA BASED ROUTER

4.1 CDMA BASED ROUTER ARCHITECTURE

The third architecture uses the CDMA technique that has been widely used in wireless communication systems because it has great bandwidth efficiency and multiple access capability. CDMA is a spread spectrum technique that encodes the data prior to transmission, then transmits the encoded data to the transmission medium so that simultaneously the medium can be used for separate information streams. The basic idea of this technique is that by using appropriate interconnections and CDMA encoding the interconnect wiring can be made very less. CDMA technique relies on the principle of code word, orthogonality, such that when multiple code words are summed they do not interfere completely with each other at any point in time and can be separated without loss of information [24].

In a packet switched network when the data has to be routed to different destinations or to the same destination through different ports, if point to point network connection is used then the data packet transfer latency varies. In CDMA technique this difference in latency can be eliminated. The CDMA NoC can transfer data packets from different sources to their destinations directly and concurrently. Multiple data packets can be transferred at the same time using this technique. This makes use of the entire bandwidth of the system. The constant data transfer latency in CDMA NoC helps to provide a guaranteed communication service for an on-chip system.

4.2 CDMA TECHNIQUE

The principle of the CDMA technique is illustrated in figure 4.1 for encoding the data at the sending end a set of orthogonal spreading codes are used [25]. The encoded data from different senders are added together for transmission without interfering with each other because

of orthogonal property of spreading codes. The orthogonal property means that the normalized autocorrelation value and cross-correlation value of the spreading codes are 1 and 0 respectively. Autocorrelation of spreading code refers to the sum of the products of a spreading code with itself, while cross-correlation refers to the sum of the products of two different spreading codes. At the receiving end, because of the orthogonality property, the data can be decoded from the received sum signals by multiplying the received signals with the spreading code used.

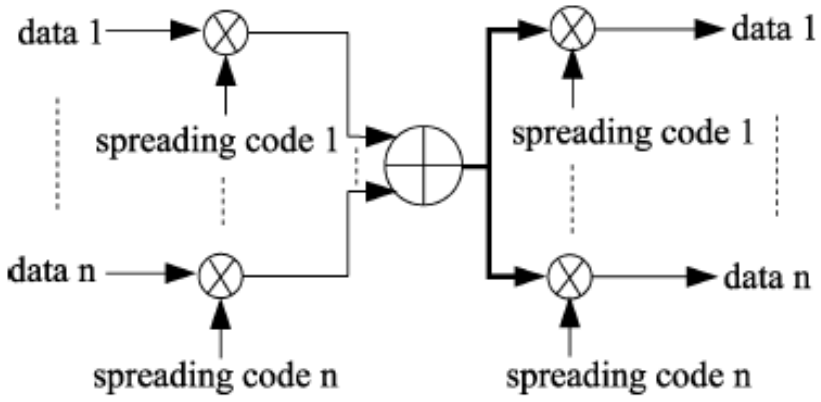

Figure 4.1. CDMA technique and principle

The spreading codes in CDMA are assigned through various protocols.

1) Common Code Protocol (C protocol): All users in the network use the same spreading code to encode their data packets to be transferred.

2) Receiver-Based Protocol (R protocol): Each user in the network is assigned a unique spreading code used by the other users who want to send data to that user.

3) Transmitter-Based Protocol (T protocol): The unique spreading code allocated to each user is used by the user himself to transfer data to others.

4) Common-Transmitter-Based Protocol (C-T protocol): The destination address portion of a data packet is encoded using C protocol, whereas, the data portion of a packet is encoded using T protocol.

5) Receiver-Transmitter-Based Protocol (R-T protocol): It is the same as the C-T protocol except that the destination address portion of a data packet is encoded using R protocol.

6) Transmitter-Receiver-Based Protocol (T-R protocol): Two unique spreading codes are assigned to each user in the network, and then a user will generate a new spreading code from the assigned two unique codes for its data encoding.

Among the introduced spreading code protocols, only T protocol and T-R protocol are conflict-free if the users in the network send data to each other randomly. Because the T-R protocol has the drawback of using a large amount of spreading codes and complicated decoding scheme, T protocol is preferred in the CDMA NoC. However, if T protocol is applied in the network, a receiver cannot choose the proper spreading code for decoding because it cannot know who is sending data to it. In order to solve this problem, an arbiter-based T protocol (A-T protocol) is developed for the CDMA NoC. In a CDMA NoC which applies A-T protocol, each user is assigned with a unique spreading code for data transfer. When a user wants to send data to another user, he will send the destination information of the data packet to the arbiter before starting data transmission. Then, the arbiter will inform the requested receiver to prepare the corresponding spreading code for data decoding according to the sender. After the arbiter has got the acknowledge signal from the receiver, it will send an acknowledge signal back to the sender to grant its data transmission. If there is more than one user who wants to send data to the same receiver, the arbiter will grant only one sender to send data at a time. Therefore, by applying the proposed A-T protocol, spreading code conflicts in the CDMA NoC can be eliminated.

4.3 CDMA ROUTER ARCHITECTURE

The CDMA router architecture has six basic functional blocks.

1 .FIFO Buffer

2. Walsh Code Generator

3. Scheduler

4. Modulator

5. Code Adder

6. Demodulator

The input data packets from different input ports are stored in FIFO buffer of depth four. The data packet which is of eight bits consists of three bit destination address, three bit source address and then the data. By selecting the Walsh code of higher width the number of cores attached to the router can be increased. This makes the CDMA router more scalable [26, 27, 28].

4.4 FIFO BUFFER

While many network switches use output buffering to avoid head-of-line (HOL) blocking, input buffering is used in this design because of its simplicity. Input buffering has lower complexity and lower cost of implementation. For an N by N switch matrix, in the case of input buffering the switch matrix and the memory need to run as fast as the line rate, while the output buffering has to run N times as fast as the line rate. The width of each buffer is equal to

the packet length and each buffer can hold four packets. Store and forward routing is used for simplicity of implementation.

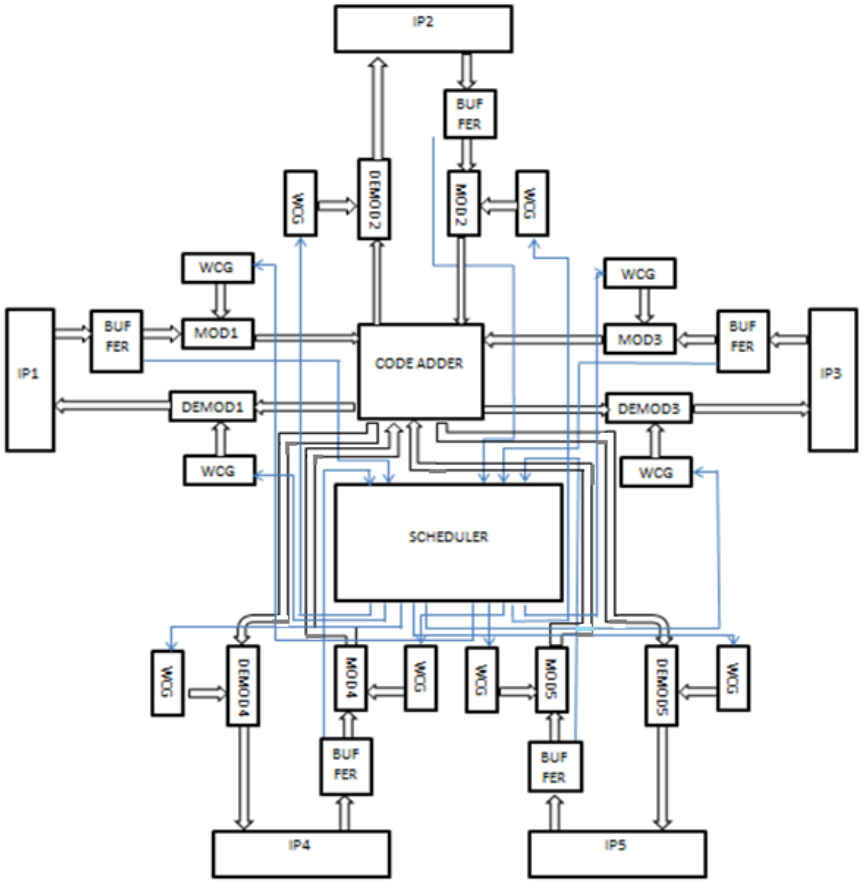

Figure 4.2. CDMA Router architecture

4.5 WALSH CODE GENERATOR

The spreading code used in this design is the 8-chip walsh code. It is the only code that satisfies both orthogonal and balance properties. So it is used widely in CDMA based NoC architectures. One way to generate the Walsh code is to use a Hadamard matrix with recursive procedures which is given by

$$H_1 = [0]$$

$$H_2 = \begin{bmatrix} H_1 & H_1 \\ H_1 & \bar{H}_1 \end{bmatrix}$$

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & \bar{H}_N \end{bmatrix}$$

TABLE 4.1. TRUTH TABLE FOR 8 BIT WALSH CODE

INPUT	**OUTPUT**
000	11111111
001	10101010
010	11001100
011	10011001
100	11110000
101	10100101
110	11000011
111	10010110

4.6 SCHEDULER

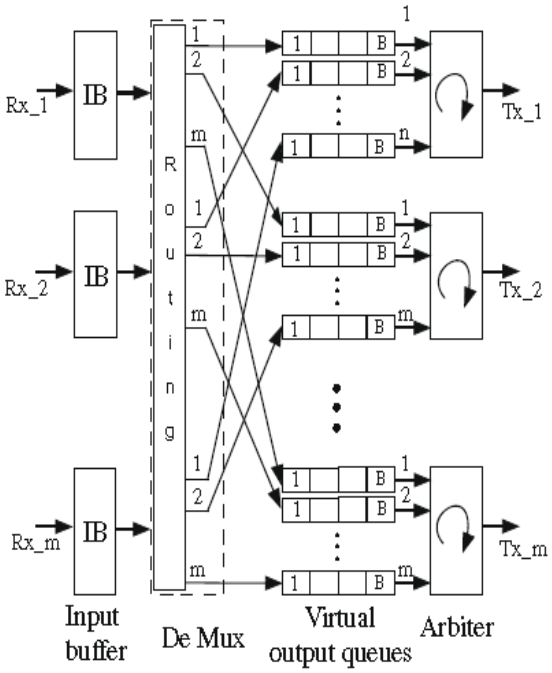

Figure 4.3. (a) Internal structure of scheduler

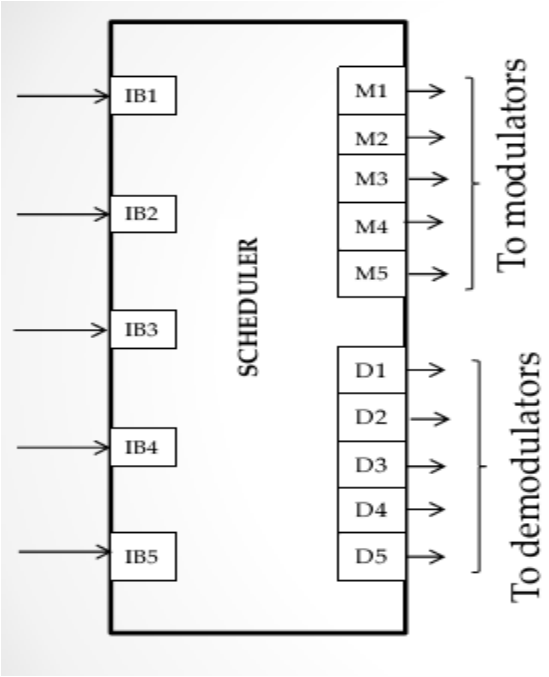

Figure 4.3. (b) Conceptual diagram of scheduler

Figure 4.3 shows the internal structure of scheduler and its conceptual diagram [29]. The six bit headers from all the resources that include three bit source address and three bit destination address are input to the scheduler. As per the scheduling algorithm, scheduler outputs source addresses and destination addresses to the Walsh Code Generators both at source and destination sides.

The scheduler uses the concept of virtual output queue. The virtual output queue is used to address the problem of head of line blocking that occurs in communication systems.

The virtual output queue means that the queue will actually be at the input but it seems as if the queues are at the output. In switch architectures the queue can be either at the input side or the output side. But the input queue suffers from head of line (HOL) blocking which limits the

throughput of the architecture. When a single queue is maintained in the input port, the packet in the front of the queue can block the traversal of the packets behind it if its output port is busy. So even if the output ports of the other packets are free they cannot traverse in the network because the packet at the head of the queue will be blocking them.

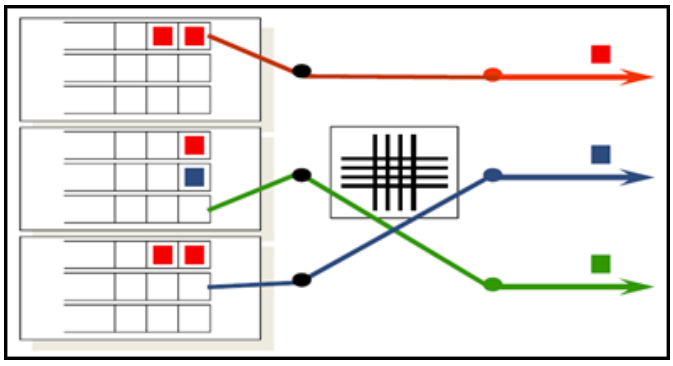

Figure 4.4. Illustration of a virtual output queue

In virtual output queue each input port maintains a separate queue for each output port [30]. In place of one queue N queues are maintained at each input port if there are N input ports. For an N port switch the buffer memory speed must operate at N times the link speed in order to prevent packet loss. The case of N times occurs when N-1 input ports all simultaneously transfer a packet to the single output port. So the throughput of the network increases. The virtual output queue can achieve 100% throughput performance with an effective scheduling algorithm.

Figure 4.4 shows the main blocks of an m-port output-queuing router. The modeling methodology adopted here could be easily applied to other router architectures. In output-queuing routers, packets arrive at the input of the router asynchronously. Then the packet header of each incoming packet, which contains the destination address, is examined by the routing module. Based on the routing table, the demultiplexers are enabled to direct the incoming packets to the corresponding output queues. As shown in figure 5.4, there are m queues for each

output port serving as FIFO buffers. Finally, the output arbiter uses a round robin scheduling algorithm to serve backlogged queues one after another at each output port in a fixed order.

4.7 MODULATOR

The TX block receives a packet from the buffer and examines its destination field. TX then selects the Walsh codeword that corresponds to this destination. The MOD block modulates the payload bits with the selected codeword. In other words, each payload bit is spread by modulation with the codeword. The specific form of CDMA modulation that is used is given in the following algorithm:

Modulation Algorithm:

if data is 0 **then**

assign codeword itself

else if data is 1 **then**

assign inverted codeword

end if

4.8 CODE ADDER

All of the modulated data from the seven resources are summed together in the code adder. The summation range of each code word chip is thus from 0 to 7. The internal architecture consists of cascaded full adders. The summation result is then sent to the demodulator.

4.9 DEMODULATOR

The demodulator recovers the original data from the summed and spread data. The details of the demodulation procedure are as given in the following algorithm. For example, assume that resource 4 (R4) wants to send a bit 0 with Walsh code C4, which is [0 0 0 0 1 1 1 1], and that the other six resources also send 0 or 1 simultaneously in a similar manner. After the code adder sums all of the modulated signals coming from all seven resources, the summed value P is [3 0 3 2 2 3 4 3]. The demodulator module first doubles each digit, resulting in [6 0 6 4 4 6 8 6]. The bits of code word X[i] determine how the decision will be made. If the bit of the codeword is '0', 2P-N is used in the decision, whereas -2P+N is used when the codeword bit is '1'. In this example, these steps would result in [-2 -8 -2 -4 4 2 0 2]. Then, upon adding up all of these values, we have a result of -8, which we divide by N, i.e. 8 in our case. Therefore, the final value is -1. From the demodulation algorithm, we would correctly determine that the original data was a '0' because it is equal to -1. By repeating this process, we can recover all of the original data that was sent.

Demodulation Algorithm:

Let

$$X[i] = \begin{cases} (2P-N) & \text{if codeword[i] is 0} \\ -2P+N & \text{if codeword[i] is 1} \end{cases}$$

Where N is the size of codeword

P is the sum of all the modulated values

Let

$$\lambda = \sum_{i=0}^{N-1} X[i] / N$$

if $\lambda = 1$ then

demodulated data value is 1

else if λ = -1 then

demodulated value is 0

end if

4.10 SIMULATION RESULTS

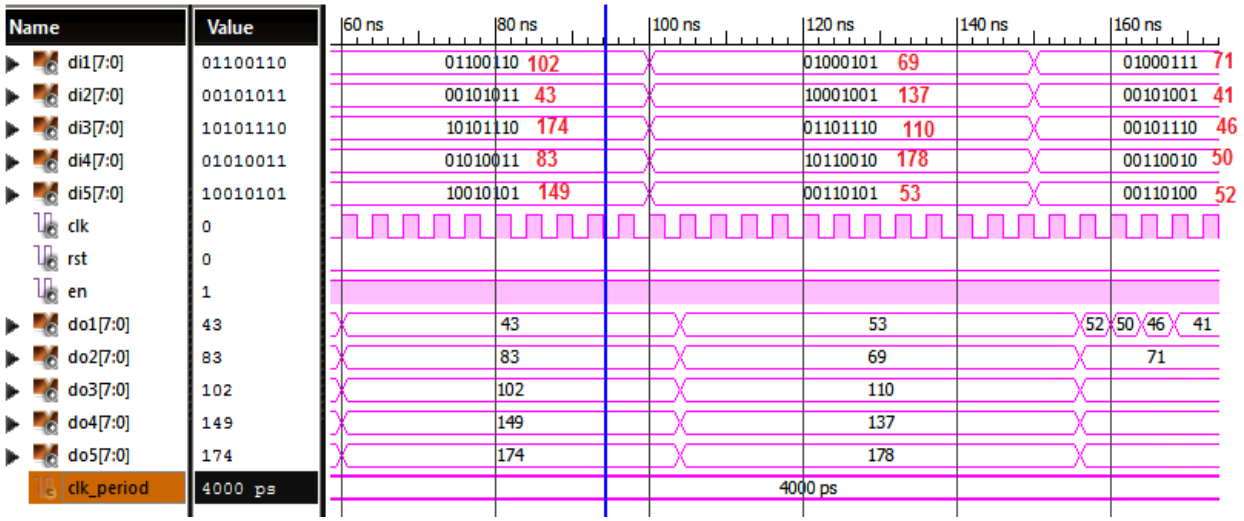

Figure 4.5 Simulation waveform of CDMA router

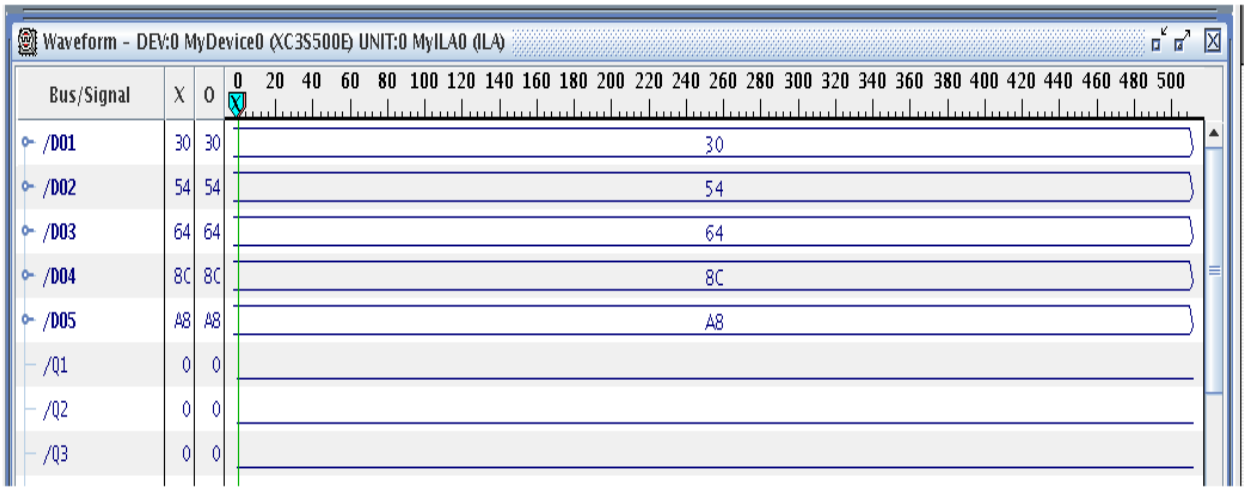

Figure 4.6. Chipscope pro analyzer waveform

The simulation waveform of CDMA router is as shown in figure 4.5. From the figure we can conclude that the output arrives at all the output ports at the same time which makes it a concurrent transmission and the latency is also same for all the output ports. There is no dependency on external enable signal. The output depends only on clock and reset signal. For the router functioning to take place the reset signal must be kept low. Round robin algorithm is used if there is data contention. Figure 4.6 shows the chipscope pro analyzer waveform after it is dumped in Spartan 3E FPGA kit.

4.11 SYNTHESIS RESULTS

The synthesis result of router architecture III is as given in table 4.2. The synthesis result provides a summary and analysis of netlist generation of the design. The device used for implementation is Spartan 3E with a speed grade of -4 and the results given are obtained after the FPGA implementation.

TABLE 4.2. PERCENTAGE OF RESOURCE UTILIZATION OF FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

RESOURCE	**USED**	**AVAILABLE**	**UTILIZATION PERCENTAGE**
Slices	1987	4656	42
4 input LUTs	4132	9312	44
Bonded IOBs	43	232	18
GCLK	1	24	4

4.12 TIMING SUMMARY

The timing summary of the router architecture I design is given in table 4.3.

TABLE 4.3. TIMING SUMMARY OF PROPOSED ARCHITECTURE III ON FPGA-XILINX (DEVICE SELECTED – 3S500EFG320-4)

Maximum frequency	189.430 MHz
Minimum period	5.279 ns
Minimum input arrival time before clock	6.571 ns
Maximum output required time after clock	8.720 ns

CHAPTER 5

COMPARISON OF DESIGNED ROUTERS

5.1 PERFORMANCE ANALYSIS OF THE THREE ARCHITECTURES

The three architectures are compared with each other for various performance parameters and the results are as discussed below.

5.1.1 LATENCY

Latency is the measure of time delay in the system. In a packet switched network it is the time taken by the data packet to move from the source port to the destination port. In a network a packet will be forwarded over many links. Each link will not forward a packet until it has received all the data packets. This causes delay in transmission of a packet. Then queuing delay occurs when different packets are vying for the same port. Then processing delay also occurs to a packet in traversal when it has to go through different entities. All these contribute to the latency incurred to a data packet. Latency limits the total bandwidth of a network [31, 32].

In the three architectures all of them have different latencies. In the first architecture the latency is different for different ports. The latency varies from 24 ns, 28 ns, 32 ns, 36 ns and 40 ns. For the second architecture the latency variation is from 52 ns, 56 s, 60 ns, 64 ns and 68 ns. In the third architecture the latency is constant for all the output ports and the value came out to be 10 ns.

5.1.2 THROUGHPUT

Throughput of a network is the amount of data in the input that arrives at the output. It is an average rate of successful message delivery. In routers packets can get dropped due to congestion. So an effective scheduling algorithm is required for maximum throughput. Throughput is calculated using the formula No of resources * No. of payload bits * Clock

frequency. Throughput is obtained in Gbps. The payload vs throughput and the throughput vs delay graph of the three architectures is given in figure

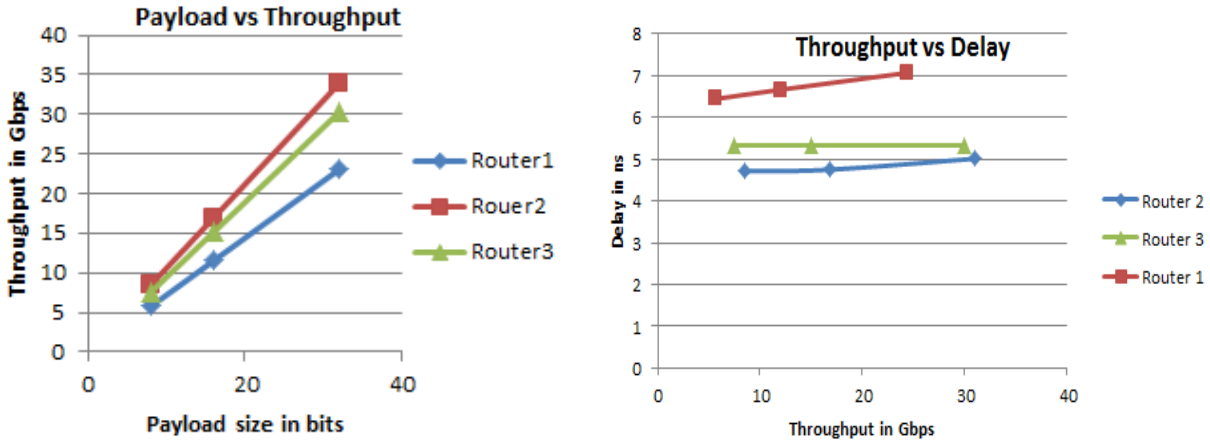

Figure 5.1. Comparison of three designed architectures

TABLE 5.1. COMPARISON OF AREA AND POWER

Architecture	Area (Number of slices)	Power (mW)
Router I	1464	97.26
Router II	418	90.37
Router III	1987	84.12

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 CONCLUSION

The various challenges faced by researchers in SoC design forced them to look for new alternatives which paved the way for Network-on-chip technology. The NoC is a vast and emerging research area that is still in its initial stages. The NoC area has a significant influence in the design of next generation SoC or multicore architectures. In our project we went through the various research aspects of NoC and details of network topology and routing algorithms were explored. We tried to contribute in the research of NoC by exploring the design space of NoC routers which is a dominant component of the network.

The main focus of our current research was aimed at an efficient design of a router for NoC applications. The router is the most important component since it determines various network parameters like latency, throughput and delay. In this project we went through three different router architectures. All the three router designs were of five input and five output port architecture. The designing has been done using the hardware description language VHDL in XILINX ISE tool. Its FPGA implementation is done and its functional model is also verified.
After analyzing the three architectures we concluded that the CDMA router architecture performs better than the other two. It has constant delay, constant latency, high throughput. Moreover it has concurrent transmission which gives it more flexibility over the other two architectures and it is less error prone. But it has more area than the other two.

6.2 FUTURE WORK

The ultimate goal of this project is to develop a 4x4 NoC. The work conducted so far is the first part of the whole project. Future work includes the extension of the router architectures and to construct an efficient NoC. The FPGA implementation of the NoC will also be done.

REFERENCES

[1] Gordon E Moore, "Cramming more components onto integrated circuits," *Electronics*, Vol. 38, No. 8. (19 April 1965), pp. 114-117.

[2] International Technology Roadmap for Semiconductors, report 2012.

[3] Resve Saleh, Shahriar Mirabbasi, AlanHu, Mark Greenstreet, Guy Lemieux, Partha Pratim Pande, Cristian Grecu, and Andre Ivanov, "System-on-Chip: Reuse and Integration," *Proceedings of the IEEE*, vol. 94, no. 6, Jun. 2006.

[4] R. Rajsuman, *System-on-a-Chip Design and Test*, Boston, MA: Kluwer, 2000.

[5] 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.

[6] Luca Benini and Giovanni De Micheli, "Networks on chips: a new SoC paradigm," *Comput.*, vol. 35, no. 1, pp. 70-78, 2002.

[7] Rickard Holsmark and Magnus Hgberg, "*Modelling and Prototyping of a Network on Chip*," Master of Science Thesis, 2002 Electronics.

[8] Muhammad Ali, Michael Welzl, and Martin Zwicknagl, "Networks on Chips: Scalable Interconnects for Future Systems on Chips," *4th European Conference on Circuits and Systems for Communications*, pp. 240-245, 2008.

[9] M. Pirretti, G. M. Link, R. R. Brooks, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "Fault tolerant algorithms for network-on-chip interconnect," *Proceedings. IEEE Computer society Annual Symposium on VLSI*, pp.46-51, Feb. 2004.

[10] Israel Cidon and Idit Keidar, "Zooming in on network-on-chip architectures,"*16th international conference on Structural Information and Communication Complexity*, 2009.

[11] Xinan Zhou, "*Performance evaluation of network-on-chip interconnect architectures*," Master of Science Thesis, 2009 Electrical Engineering.

[12] A. Jantsch and H. Tenhunen. *Networks on Chip*, Kluwer Academic Publishers, 2003.

[13] W.J. Dally and B. Towles "Route packets, not wires: On-chip interconnection networks," *Proc. DAC*, 2001.

[14] David Atienza, Federico Angiolini, Srinivasan Murali, Antonio Pullini, Luca Benini, and Giovanni De Micheli, "Network-on-Chip design and synthesis outlook," *INTEGRATION, the VLSI journal* vol. 41, pp. 340–359, 2008.

[15] Davide Bertozzi and Luca Benini Xpipes "A Network-on-Chip Architecture for Gigascale Systems-on-Chip," *IEEE CIRCUITS AND SYSTEMS MAGAZINE*, SECOND QUARTER, pp. 18-31, 2004.

[16] 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.

[17] Li-Shiuan Peh and Natalie Enright Jerger, "*On-Chip Networks Synthesis,"* Lectures on Computer Architecture, 2009.

[18] Everton Carara, Ney Calazans, and Fernando Moraes, "A new Router Architecture for High-Performane Intrachip Networks," *Journal Integrated Circuits and Systems*, vol. 3, no. 1, pp. 23-31, 2008.

[19] Ville Rantala, Teijo Lehtonen, and Juha Plosila, "Network on Chip Routing Algorithms," *Journal of Systems Architecture*, TUCS Technical Report No 779, pp. 5-7, Aug. 2006.

[20] 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.

[21] Brett Stanley Feero, and Partha Pratim Pande, "Networks-on-Chip in a Three-Dimensional Environment: A Performance Evaluation," *IEEE Transactions on Comput.*, vol. 58, no. 1, Jan. 2009.

[22] Xingang Ju, and Liang Yang, "NoC Research and Practice: Design and Implementation of 2×4 2D-Torus Topology," *Int. J. Information Technology and Computer Science*, vol. 4, pp. 50-56, 2011.

[23] Eung S. Shin, Vincent J. Mooney III and Georgey Riley, "*Round-robin Arbiter Design and Generation*," Technical Report GIT-CC-02-38, College of Computing, Georgia Institute of Technology, Jul. 2002.

[24] Xin Wang, Tapani Ahonen, and Jari Nurmi "Applying CDMA Technique to Network-on-Chip," *IEEE Trans. VLSI Syst.*, vol. 15, no. 10, Oct. 2007.

[25] Ahmed A. El Badry and Mohamed A. Abd El Ghany, "A CDMA Based Scalable Hierarchical Architecture for Network-On-Chip," *IJCSI International Journal of Computer Science Issues*, vol. 9, Issue 5, no. 2, Sep. 2012.

[26] Anant W. Hinganikar, Mahendra A. Gaikwad and Rajendra M. Patrikar, "PERFORMANCE EVALUATION OF CDMA ROUTER FOR NETWORK - ON – CHIP," *International Journal of VLSI design & Communication Systems (VLSICS)*, vol. 3, no. 3, Jun. 2012.

[27] Daewook Kim, Manho Kim and Gerald E. Sobelman, "CDMA Based Network On Chip Architecture," *IEEE Asia Pacific Conference on Circuits and Systems*, pp. 137-140, 2004.

[28] Daewook Kim, Manho Kim and Gerald E. Sobelman, "Design of high performance scalable CDMA router for on-Chip Switched Network" *Proc. ISSOC*, pp.32–35, 2005.

[29] Haytham Elmiligi, Ahmed A. Morgan, M. Watheq El-Kharashi, and Fayez Gebali, "Power optimization for application-specific networks-on-chips: A topology-based approach," *Microprocessors and Microsystems*, vol. 33, pp. 343–35, 2009.

[30] Yoshigoe, K. and Christensen, K.J. "An evolution to crossbar switches with virtual output queuing and buffered cross points," *IEEE Network*, pp. 48-56, Sep./Oct. 2003.

[31] Emo Salminen, Ari Kulmala, and Timo D. H¨am¨al¨ainen, "*Survey of Network on Chip proposals*," White Paper, OCP-IP, March 2008.

[32] A.Alhussien, C.wang, and N.Bgherzadeh, "Design and evaluation of a high throughput robust router for Network on chip," *Computers & Digital Techniques*, IET, vol. 6 , no. 3, May 2012.

PUBLICATIONS

[1] Swapna S, Ayas Kanta Swain, and Kamala Kanta Mahapatra, “Design and Analysis of 5-Port Router for Network on Chip,” *IEEE PrimeAsiaProc.,* pp. 51–55, Dec. 2012.