A

DISSERTATION REPORT

ON

Router Architecture using XY routing

is submitted in partial fulfillment for the award of degree of

MASTER OF TECHNOLOGY

IN

EMBEDDED SYSTEMS

BY

GARIMA KUMARI

(2015PEB5358)

UNDER THE GUIDANCE OF

Dr. Lava Bhargava

Associate Professor, ECE Department

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

MALVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

JUNE 2017

Department of Electronics and Communication Engineering

Malaviya National Institute of Technology, Jaipur

Certificate

This is to certify that this Dissertation report entitled **"Router Architecture Using XY Routing"** by **Garima Kumari** (**2015PEB5358**), is the work completed under my supervision and guidance, hence approved for submission in partial fulfillment for the award of degree of **Master Of Technology** in **EMBEDDED SYSTEMS** to the Department of Electronics and Communication Engineering, Malaviya National Institute of Technology, Jaipur in the academic session 2016-2017 for full time post-graduation program of 2015-2017. The contents of this dissertation work, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Lava Bhargava

Associate Professor
Department of Electronics and Communication

MNIT Jaipur

Declaration

I, hereby declare that the work which is being presented in this project entitled **"Router Architecture Using XY Routing"** in partial fulfillment of degree of Master of Technology in **Embedded Systems** is an authentic record of my own work carried out under the supervision and guidance of **Dr. Lava Bhargava** in Department of Electronics and Communication, Malaviya-National Institute of Technology, Jaipur.

I am fully responsible for the matter embodied in this project in case of any discrepancy found in the project and the project has not been submitted for the award of any other degree. I also confirm that I have consulted the published work of others, the source is clearly attributed and I have acknowledged all main sources of help.

(Garima Kumari)

Acknowledgement

I am grateful to my supervisor Dr. Lava Bhargava for his constant guidance and encouragement and support to carry out this work. His excellent cooperation and suggestion provided me with an impetus to work and made the completion of work possible. He has been great source of inspiration to me, all through. I am very grateful to him for guiding me how to conduct research and how to clearly & effectively present the work done.

I would like to express my deepest sense of gratitude and humble regards to our Head of Department Prof. K. K. Sharma for giving encouragement in my endeavors and providing necessary facility in the Department. I am very thankful to Mr. Amit Joshi , PG Coordinator and all other faculty members of ECE, MNIT for their valuable assistance and advise.

I would like to thank Mr. Yogendra Gupta for timely help in this project work, I would also like to thank my friends for their support in discussions which proved valuable for me. I am indebted to my parents and family for their constant support, love, encouragement and sacrifices made by them so that I could grow up in a learning environment. Finally, I express my sincere thanks to all those who helped me directly or indirectly to successfully complete this work.

(Garima Kumari)

Abstract

The network on chip approach for complex SoC designs is a new paradigm which gives efficient on chip communication. In today's industry complexity of each component is increasing rapidly as transistor density is increasing , higher operating frequencies and shorter time to market and multi processor system on chip etc are required which use conventional bus based architecture. Wire delay is also increasing as compared to gate delay as transistor size is shrinking. So we need more structured and scalable architecture to fit with increasing complex elements on a single chip. Conventional bus based architecture and point to point connection doesn't allow efficient on chip communication requirements as these architectures are not scalable, flexible that is may not allow reuse of design. Noc allows scalable communication and also supports design reuse.

The NoC architecture uses links, routers and network interface and it may increase cost, area and power consumption as many on chip resources are required. In Noc, router plays an important role as data is routed through it in terms of packets. So the architecture of router must be efficient such that it provide low latency and high throughput. In this thesis we have described pipelined router architecture and its implementation in verilog. This router architecture has been implemented using xy routing. Further we have modified xy routing as now if there is any blockage in the path of packet transmission. Then packet will traverse a path which is free from blockage. After analysing the results, I have made some result and concluded my experiments. At the end, this report summarizes the finished work and gives some more developing directions of router architecture.

Contents

Declaration ii

Acknowledgement iii

Abstract iv

List of figures viii

1 Introduction 1

1.1 Introduction 2

1.1.1 Bus based architecture 2

1.1.2 Dedicated point to point communication 3

1.1.3 Network on chip communication 3

1.2 Motivation 3

1.3 Thesis organization 4

2 An overview of NoC 5

2.1 NoC building blocks 5

2.1.1 Link 5

2.1.2 Network interface 5

2.1.3 Routing node 6

2.2 Network topology 6

2.2.1 Direct topology 7

2.2.1.2 Mesh 7

2.2.1.3 Torus 8

2.2.2 Indirect topology 8

2.2.2.1 Spidergon 9

2.2.2.2 Butterfly topology 9

2.2.2.3 Star 10
2.2.3 Metric for comparing network topology 10
2.3 Switching technology 10
2.3.1 Circuit Switching 11
2.3.2 Packet switching 11
2.3.2.1 Store and forward 12
2.3.2.2 Virtual cut through 12
2.3.2.3 Wormhole Switching 12
2.4 NoC evaluation matric 12
2.4.1 Latency and throughput 12
2.4.2 Area 13
2.4.3 Energy and power consumption 13
2.4.4 Quality of service 13
3 Basic architecture of NoC router 14
3.1 Types of routing algorithm 14
3.1.1 Oblivious routing 14
3.1.2 Adaptive routing 15
3.2 Arbiter 16
3.3 Crossbar network 16
4 Design of router using XY routing algorithm 17
4.1 Allocator 18
4.1.1 XY routing algorithm 19
4.1.2 Packet format 20
4.2 Arbiter 21
4.3 Router implementation 22

5 Simulation results 23

6 Conclusion and future work 29

Reference 30

List of figures

1.1 Bus based communication 1

1.2 Point to point communication 2

1.3 NoC interconnection 3

2.1An NoC architecture 6

2.2 Mesh 8

2.3Torus topology 8

2.4 4X4 Spidergon topology 9

2.5 Butterfly topology 9

2.6 Star topology 10

3.1 Diagram of oblivious routing 14

3.2 Router with single node 16

3.3 5X5 Crossbar 17

4.1 Allocator 18

4.2 XY routing 20

4.3 Packet format 21

4.4 Block diagram of arbiter 21

4.5 Block diagram of arbiter 22

5.1 Simulation result 23

5.2 Simulation waveform of allocator 24

5.3 DEMUX 25

5.4 Channel buffer 26

5.5 Simulation result for XY routing 27

5.6 Output of arbiter module 27

5.7 Waveform showing data at buffer of destination 28

Chapter 1

Introduction

1.1 Introduction

According to Moore's law, As transistors which can be used on an IC is increasing exponentially, this number is becoming double approximately after two years. A large number of transistors, gates and many circuits on a single IC chip is becoming a great challenge to IC designers. So they have emerged with a new design methodology known as System on-Chip (SoC). The system on chip includes hardwares like micro processors, peripherals, micro controllers, digital signal processors and various custom logic block and softwares for controlling the hardwares.

Various communication structures in SoC are :

1.1.1 Bus based communication

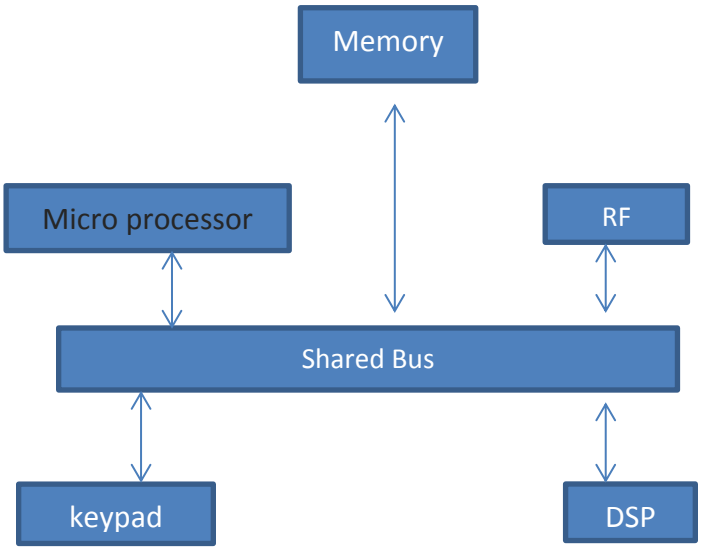


Fig 1.1: Bus based communication system

Generally architecture of interconnecting components is based on shared bus or dedicated wires. If any system has limited components or cores then bus based architecture is really effective.

When complexity of system is increased, number of core or components are also increased. Thus dedicated wires or bus based communication have less flexibility and reusability.

A shared bus is a collection of wires, they are commonly used by multiple cores and cores share the bandwidth. This approach is more flexible and completely reusable. This scheme suits for Master-Slave communication, in master slave structure peripherals (slaves) wait for data which is coming from more complex processing element (master). But this approach allows only one communication at a time. Its scalability is a big issue as it has limitation of few IP cores. Thus scalability becomes a great problem for systems using bus based communication.

1.1.2. Dedicated point to point communication

In this connection there is an independent path available from one node to an other node. Each node is connected with other nodes in a direct path as shown in figure 1.2. The performance of this communication is high. Whenever there is data available at a node, data can be diretly transferred without any delay, as there is a dedicated path between every source node and destination, thus this communication offers low latency. The disadvantage of this type of communication is when data is not communicated between the nodes a path will still be there between the nodes which is not desired.

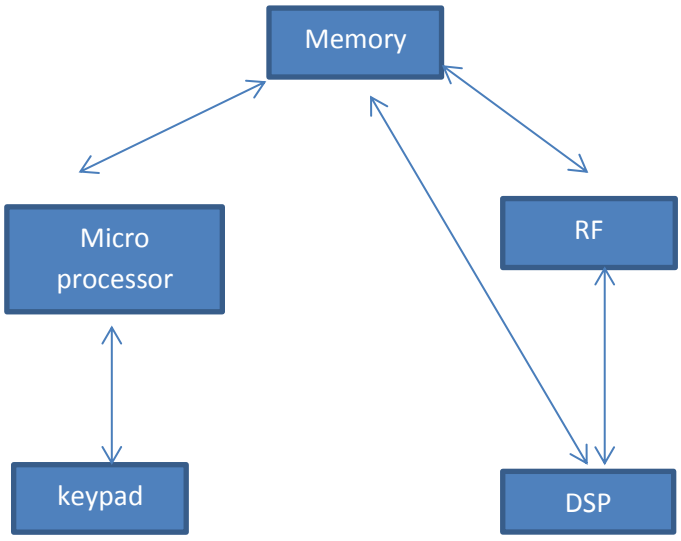


Fig 1.2: Point to point communication

For n nodes, network requires $(n-1)/2$ wires between the node. Which causes increase in area and power as the value of n increases.

1.1.3. Network on chip

As with reducing size of transistors, wire delay has been increasing when we compare it with gate delay. So we need an architecture which is more structured and scalable, which fits with the increasing complexity of an IC. Any processor can be seen as having computational units which is used to pass the information between communication blocks. Computation time is concerned with gate delay but communication time is concerned with wire delay. With the size of transistors which is reducing, gate delay is also decreasing as compared to wire delay. So wire delay becomes more dominating as compared to gate delay. This translates computational centric design into communication centric design.

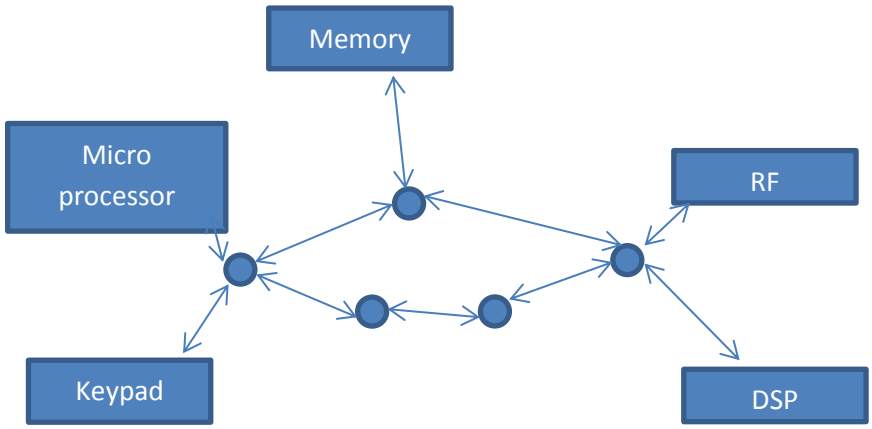


Fig 1.3: Typical NoC

NoC architecture is advantageous in terms of performance, scalability and power efficiency as compared with the bus based architecture. It also provides a solution for the the issue of scalability which was major problem of bus based architecture.

1.2 Motivation

Any digital system consists of three basic blocks which are Logic, memory and communication. Logic deals with data such as it manipulates data, memory is used to store the data for future purpose and communication block is used to transfer data between different modules. Communication system mainly decides performance, speed and reliability of the system. NoC is a communication sub system between IP cores and SoC. In which using large number of functional node limits the performance of NoC. Now a days, the number of processors and memories in a single multi-core system are increasing. So communication using channel of wires causes high latency, high power consumption and low throughput. Thus for a good communication system having a high performance digital system with low power consumption is required which causes high throughput, low latency.

1.3 Thesis Organization

In his thesis, We have discussed basic components of NoC and router microarchitecture of NoC. In Chapter 1 we have discussed different types of communication system and their drawbacks and motivation for choosing work on NoC.

Chapter 2 has overview of NoC. In this we have described network building blocks, network topologies, matric for comparing topologies, different switching technique and NoC evaluation matric. In chapter 3 we have discussed basic router architecture and its various components. Chapter 4 discusses design of pipelined router architecture. Chapter 5 discusses simulated output results and utilization summary.

In Chapter 5, we have discussed simulation results of routing architecture with and without consideration of blockage nodes (routers). Future direction and the conclusion of this work is discussed in Chapter 6.

Chapter 2

An Overview Of NoC

2.1 NoC Building Blocks

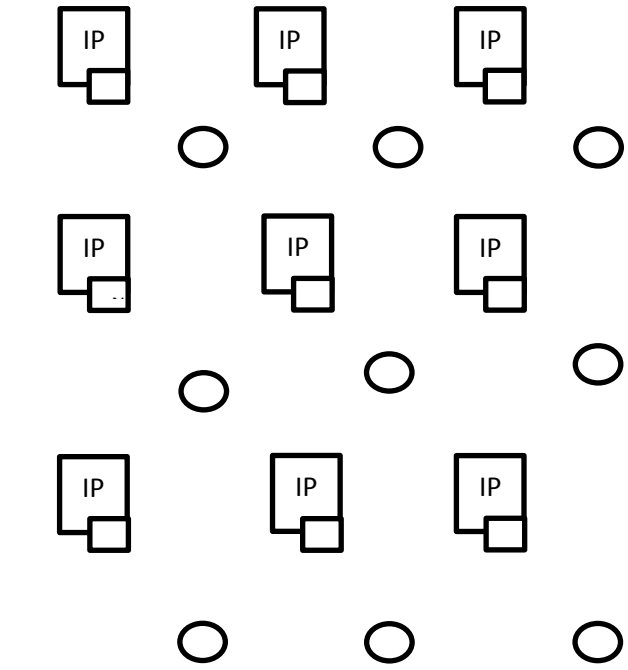
NoC allows computational blocks (IP cores) to communicate over an on-chip network. An example of a NoC interconnection network is shown in Figure 2.1, which includes four basic functional blocks. These blocks are IP cores, routing nodes, and link. Generally IP cores are specific to the application thus not considered part of the NoC design.

2.1.1 Link

To send packets from source to destination routers, links are used. It provides a physical connection between the nodes. It is basically a segment of wires which are used to provide connection between the routers in the network architecture. It may also provide buffer and separate control lines for connection establishment. Link decides channel bit width. The number of wires per channel is known as channel bit width.

2.1.2 Network Interface

Network interface module transforms the data packets into fixed length flits. Network interface is used to provide logical connection between IP core and network. It has two parts: front end and back end. Front end is used to handle the request from the IP core and it does not have any idea of the remaining network. The back end is connected to the network that administers ordering and reordering the packets, buffers and IP core address .



Indicates Router, dicates network interface

Fig 2.1: NoC interconnection

2.1.3 Routing Node

Router is the main component in NoC architecture. It forwards a flit received from a source router to a downstream destination router by using routing computation algorithm. So its area and speed plays an important role in the whole network.

2.2 Network Topologies

A network is combination of wires and routers. Topology determines how the routers and channels are connected to each other. Topologies should be chosen such that area is minimized

while maximizing utilization. Topology determines the count of nodes (or routers) which are traversed by a message and the length of interconnection between nodes. Thus it influences latency of the network. As routers which are being traversed and the links sustains energy, then effect of topology on the count of router has also direct effects on energy consumption of the network. The topology also decides the possible number of different paths between the routers or nodes, which affects traffic of the network so topology also support bandwidth requirements. The first decision of the designer is to make the choice of the topology.

A network topology can be divided into two parts as direct and indirect one. With a direct topology, each terminal node has a router, so all routers in this topology are sources and destinations of packet while they are in traffic. In this topology, nodes can be switched through the traffic from the other routers or nodes. Nodes are separated from terminal or end nodes in an indirect topology. End nodes are only the nodes which become sources and destinations, Nodes which are intermediate simply used to transfer the traffic to and from end nodes. In a direct network, packets are forwarded directly between terminal nodes but in an indirect network, packets are transferred indirectly through a number of intermediate switch nodes between the source and the destination. Generally most of the designers use direct networks.

2.2.1 Direct topology

2.2.1.2 Mesh

In Mesh each node has network interface which is used to provide connection to the remaining four routers with the help of channel, routers and processing elements. A channel has 2 unidirectional links between 2 routers or between a router and a resource.

In a mesh, there are m number of columns and n number of rows which is simplest topology among all other. So its just a m*n matrix in which routers are placed at the intersection point of two wires and the computational blocks are placed near the routers. Routers address and resources address are given by x-y coordinates. The another name for mesh network is also Manhattan Street network. Diagram for mesh is given figure number 2.2:

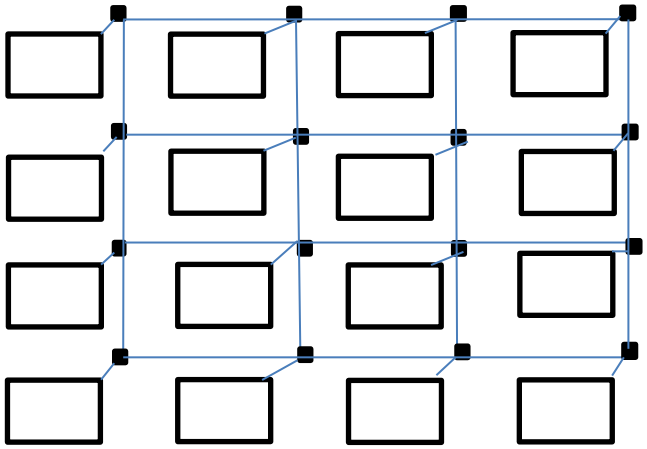


Fig 2.2: Mesh

2.2.1.3 Torus

An higher version of the mesh network is torus. Generally in a torus network there is a mesh present in which we connects heads and tails of columns and left sides of the rows are connected to the rows which are in right side through wrap around channels. There are five ports in each router. One of them has connection with the resource which is local, and the remaining has the connection with the nearest neighboring routers. There is better path diversity in torus network as compared with mesh and it also has more shortest routes.

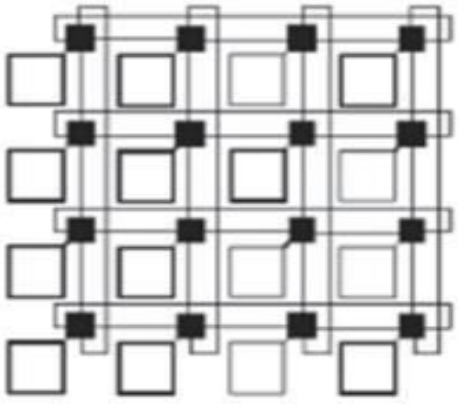


Fig : 4X4 Torus topology

2.2.2 Indirect topology

In this topology, processors has connection through one or more than one nodes which are intermediate, these switching node are used to perform the routing task and arbitration of packets. There is no direct path or connection present in between the processors.

2.2.1.3 Spidergon topology

Spidergon is a topology which is point to point and also regular topology. This is symmetric with respect to the vertex and the edge. Due to this symmetric structure all routers have information about whole network, which helps in simple routing. It connects only even number of nodes as bidirectional ring. So network implemented using this topology is simple and routing decisions are also fast. This topology is a type of polygon topology. The simplest form of polygon network is a network which is circular where packets are used to travel in a loop from one router to the other router. This network is more manifold when multiple chords are joined to the circle. When chords are available only between the opposite routers, then this topology is known as spidergon topology.

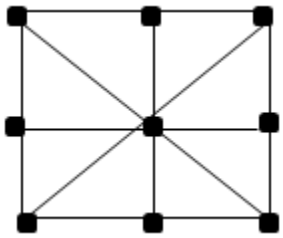


Fig 2.4: 4X4 Spidergon topology

2.2.2.2 Butterfly topology: In this topology only one path or route is present from each source node to destination node. So there is no path diversity which is its drawback. It cannot be realized without long wires. There is absence of path diversity i. e, there is only one routing path between each and every source and destination node. Long wires become its necessity for its realization. Because of presence of these lengthy wires it becomes less popular for the networks which are moderate in size and have large interconnection.

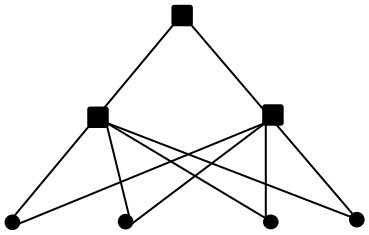


Fig 2.5: Butterfly fat tree topology

2.2.2.3 Star

In star network a central or main router is present in the middle of the whole star network and resources are present at the corners of the star.

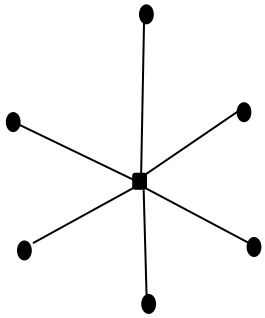


Fig 2.6: Star topology

2.2.3 Metrics for comparing network topologies

Degree : Number of links associated with each node is known as degree. Higher degree of a node requires more number of ports at router which increases complexity and cost also.

Hop count: while traversing from source to destination the number of hops or nodes taken by a message is known as hop count. Value of the maximum count of hops is calculated by the diameter in the network. This is useful for determining latency.

Maximum channel load : Maximum number of bits per second provided by every node in the network before there is saturation level. It is helpful parameter in bandwidth determination. As

first of all it determines which link in the network will be maximum congested for a particular traffic.

Path diversity: Topology in which maximum multiple shortest path are present between source and destination, provides greater path diversity as compared to topology which provides single path between the source node and destination node.

2.3 Switching technologies

In communication systems, generally there are multiple path available between the source node and destination node through which packets can be transferred. So this allocating function to allocate a particular path available out of different multiple path is performed by switching. NoC is a network which is switched, in which minimum of two links has connection through a switch. There are two different types of switching network, which are circuit switching and packet switching.

2.3.1 Circuit switching

In circuit switching there is always a dedicated connection established between two ends. If there is no data transmission then also there is a dedication connection establishment. Switch is used to make this connection active or inactive. Circuit switching first allocates a link in the network. A message is sent to the network to reserve the required link. Once acknowledgement is received by source from destination then only it starts transferring bits. So latency wise circuit switching has benefit. But it suffers from poor b1andwidth requirement.

For n node network it requires nC2 channels. As number of nodes increases area and power consumption also increases. In this technique, resources are underutilized due to dedicated connections.

2.3.2 Packet switching

In this technique data is transferred in terms of segments of message between two ends. So a single path is available which is shared by multiple number of users. There are two types of this switching. In the first type, each packet carries source and destination address and algorithms

are used to forward them. In second type, a connection between source and destination is established first. This establishment of connection can be either connection less or connection oriented. In connection less path a connection is not required in between the source and destination nodes. In connection less type of packet switching, ordering of packet is main requirement. In the mechanism of connection oriented switching first a connection is required between the source node and the destination for transmission of packet. Once packet is transferred and reached to destination, connection is released. There are various packet switching techniques available such that wormhole switching, virtual cut through switching and store and forward packet switching.

2.3.2.1 Store and forward switching

In this switching, a node must have the entire packet before it is transmitted to other nodes. Main disadvantage of this switching is that it increases delay as a node waits for entire packet transmission before sending a new packet. Buffer is also required of large size at every node to store entire packet. If a packet transmission is dropped in middle due to any reason then whole packet needs to be re- transmit.

2.3.2.2 Virtual cut through switching

In this switching, nodes do not need to wait till entire packet is stored unlike store and forward switching. In this switching delay is reduced as when a node receives a part of packet then this is forwarded to the next node if space in buffer is available.

2.3.2.3 Wormhole switching

In this switching, whole packet is segmented into flits which are known as digits of flow control. Flits are the basic smallest units on which flow control mechanism is performed. Like virtual cut through mechanism, the whole packet is not stored in this switching. So buffer size is less which causes reduction in area and power of network. Latency in this switching is also less as there is no need for routers to wait till the entire packet has arrived. This switching involves in connection oriented mechanism so there is no need of ordering of packet. In this switching flow control mechanism is done at the level of flits and at packet level switching is done.

2.4 NoC Evaluation Matric

2.4.1 Latency and throughput

There are multiple ports present in NoC router through which data can be sent, received and processed simultaneously. Therefore data transaction time plays an important role. When speed is measured in terms of delay then its referred as latency. Latency is calculated as overall run time, it can be calculated as an average of multiple divided intervals such as packet or flit latency.

We can measure speed in terms of bandwidth, which is known as throughput. Throughput is amount of data transferred per unit of time. Throughput can be measured in packet or flit per cycle, which is reciprocal of time and can be converted as a function of maximum clock frequency.

2.4.2 Area

Overall area required on a single chip should be minimum. We can measure area in terms of number of resources available.

2.4.3 Energy and power consumption

Power consumption is directly affected with area.

2.4.4 Quality of service

To provide quality of service, an NoC network must be free of the following failures

1. Livelock : When data is trapped into a cyclic path then it could not reach to destination node, which is known as livelock. XY routing avoids this failure of data being routed around their destination node by allowing it to travel through shortest path.
2. Starvation : When data can not reach to destination because any resource in the path does not grant access, then this situation is known as starvation.

3. Deadlock: When data is unable to reach at destination because some intermediate resources blocks it, then this situation is known as deadlock. So deadlock mainly occurs when packet is being blocked continuously. This problem of network can not be solved easily because data remains blocked while waiting for an event to occur. This problem in NoC can be avoided by restricting channel reservation.

Chapter 3

Basic Architecture of NoC Router

Generally there are five components in NoC router, they are input buffer, route computation logic or routing algorithm, allocator, arbiter and crossbar switch.

3.1 Types of routing algorithm

3.1.1 Oblivious routing

In this routing message can be sent randomly along either X-Y path or along Y-X path. Deterministic routing comes under this category. Dimension ordered algorithm is the example of deterministic routing algorithm in which message is traversed through dimension by dimension. In 2D mesh, dimension ordered algorithm sends packet along first X dimension then Y dimension. For example a packet which is travelling from source node (0, 0) to destination node (2, 3) will traverse X dimension first up to (2, 0) then it will traverse Y dimension upto (2, 3).

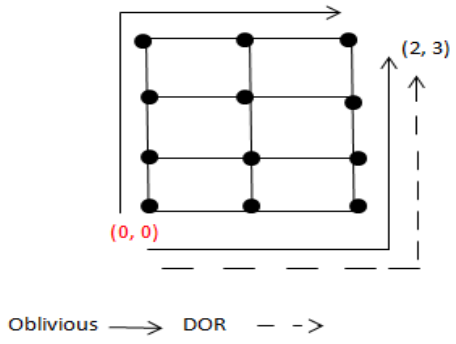


Fig 3.1: Oblivious routing

In two dimensional mesh for example all turns of links are allowed as shown in example, these turns may be limited in DOR X-Y and DOR Y-X. When all turns are allowed then there may be deadlock because of cyclic dependencies. In DOR X-Y when a message is travelling in east or west direction then turns in north and south direction are allowed. But if message is travelling in

north or south direction, turns in east or west direction is not allowed. So here deadlock is not possible as cycle is not present. Similarly in DOR Y-X when a message is travelling in north or south direction, it is allowed to take turn in east or west direction but if it is travelling in east or west direction then turns in north or south direction are not allowed. So here also there is no cyclic dependency so no deadlock possible. So we will choose between X-Y or Y-X routing depending upon in which direction more number of nodes are present. There are few advantages of dimension order routing which are as follows, simple and deadlock free etc. It eliminates path diversity, which makes routing algorithm unable to route around faults. It allows only one path between each source and destination.

3.1.2 Adaptive routing

In this routing path traversed by a message depends on traffic situation of network. For example, a message travelling the X-Y route finds traffic at any node's link in between then the message will choose other link of the same node towards destination. In adaptive routing message coming from source may not arrive at destination in the same order so it may be problematic. So it may require an ordering mechanism at destination node.

Routing algorithm can be divided into two categories as minimal and non minimal routing algorithm. Minimal routing algorithm select only that path in which minimum number of hops between the source node and the destination node are required. In non-minimal routing algorithms, any path can be selected in which number of hop between source and destination node could be increased. Non minimal routing may cause increased latency and power consumption. In router architecture, there is input port and output port available and a mechanism to connect them and a local port which is used to connect the router to the corresponding IP core. Performance of a network is based on how efficiently a router is designed.

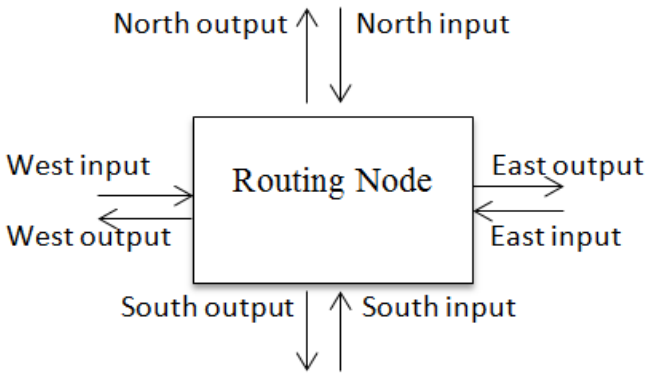


Fig 3.2: Router with single node

3.2 Arbiter

Arbiter has an important role in designing router. When two or more than two packets from different input ports want to access a single output port then it solves the contention between these by deciding which packet will access the output port. It maintains an information about ports which are free and which ports are busy. So mainly arbiters perform arbitration that is it solves dispute between multiple packets wanted to arriving at the same output port by using any scheduling algorithm. Arbitration should be done such as it allows only one packet for shared resources and it should avoid starvation problem.

3.3 Crossbar network

Crossbar is a non-blocking network, which is useful in reducing latency of the network. Any network is called non-blocking when it connects all inputs to all outputs in network. In starting non-blocking networks used in telephone lines. In non-blocking networks nodes are connected such that unused input and output pair are connected through unused nodes. Crossbar network is just like a NXM matrix in which N inputs are directly connected to M outputs without any intermediate stage. This NXM crossbar switch has M number of N: 1 multiplexers. Each multiplexer for one output is required. Diagram of 5X5 crossbar is shown as below

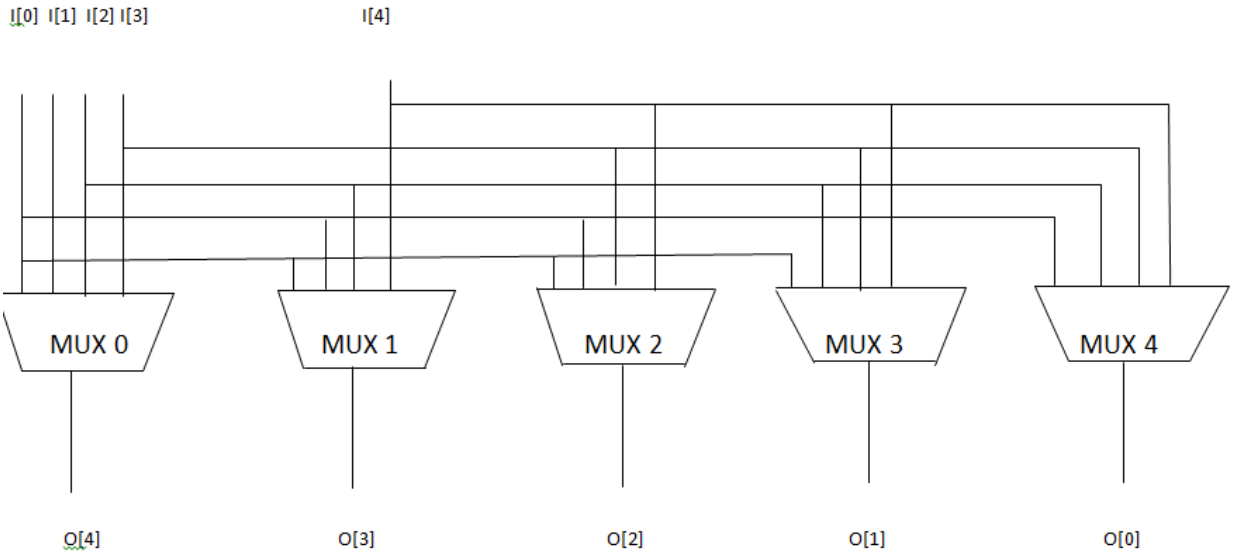


Fig 3.3: 5X5 Crossbar

Here five outputs O[0], O[1], O[2], O[3], O[4] and five inputs I[0], I[1], I[2], I[3], I[4] are present. Crossbar also performs routing algorithm in which multiple packet at a time can not access output port.

Chapter 4

Design Of Pipelined Router

In pipelined NoC router architecture, crossbar switch is devided into two parts which are allocator and arbiter.

4.1 Allocator

Each input port consists of input buffer and allocator. Here allocator determines the direction in which incoming packets are transmitted to. Mainly allocator allocates the link at input port to each incoming message. It has registers, routing computation logic, a control unit and de multiplexer. Structure of allocator is shown below

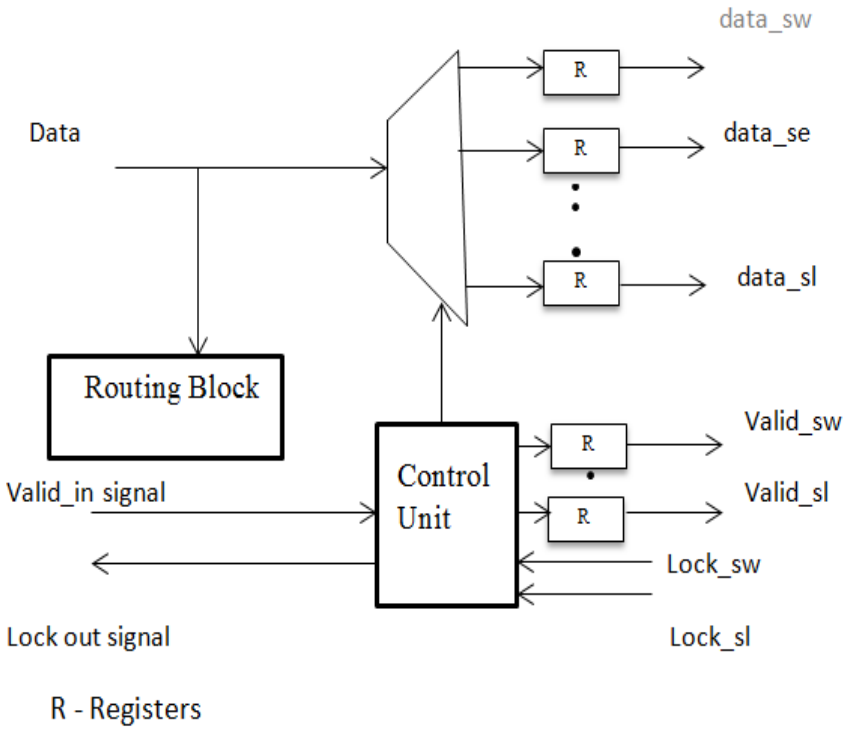


Fig 4.1: Block diagram of allocator

When a packet reachs at input port then first its stored into input buffer serially then a valid signal is given to allocator to inform that data is valid. After that routing computation block will decide the direction (out of north direction, south direction, east direction, west direction and local) of the packet in which it will be transmitted. Control unit is used to check for the available links of input port of a particular node. If link is available then then data will be transferred to the corresponding transmission link else packet will be at the same level where it was earlier.

Buffering is used to store message when they can not be transmitted or scheduled. It can also reduce latency but has impact on area overhead. We have used buffer of 16 bits to avoid head of line blocking at input and output. Head of line blocking happens when already a packet is blocked and thus it causes blocking of other packets coming later which further could be forwarded. When we use output buffer then it causes removal of blocked packet at input buffer so that other packets coming later can be transmitted or routed.

4.1.1 X-Y routing

We have used X-Y routing algorithm because there are many advantages of this routing, some are: it is deadlock free as cyclic dependency of data is absent. It also allows less area requirement for network and it is simple design. It also prevents live lock because packets sent by same source node and reached at the same destination will follow the same path. It also prevents packet reordering. But its disadvantage is that it does not provide solution for traffic or congestion. In X-Y routing position of each node is assigned with a coordinate (X, Y). We have used a mesh of 4X4 mesh for which coordinate of 2 bits is required.

When packet arrives at router, then its destination is informed by header of packet, if X coordinate of destination is greater than the source X coordinate then data will be routed towards east and if it is less than destination Y coordinate then it is forwarded towards west direction and if both are equal then there will not be any displacement as no its place is accurate.

Then it checks for Y coordinate, if Y coordinate of destination is greater than the source then data will be transferred towards north direction and if destination Y is less than source Y coordinate then data will be transferred to south direction and if both are equal then on displacement is required and data will reach the destination node.

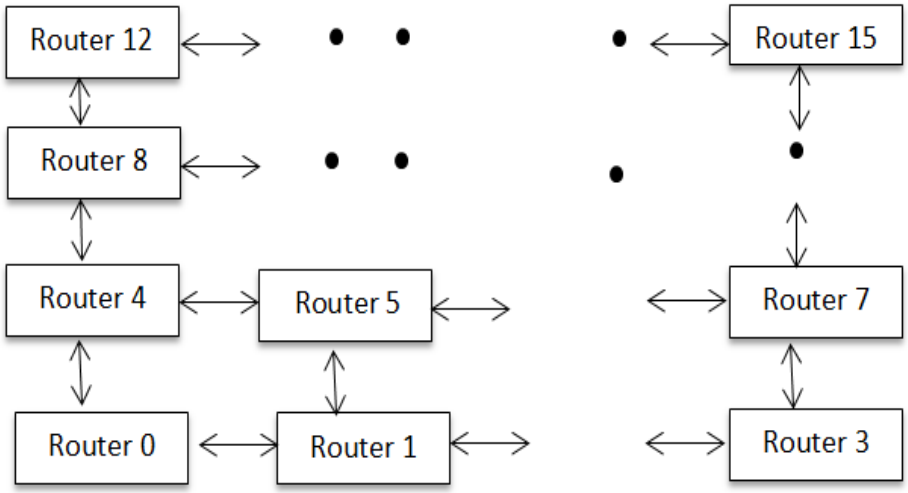


Fig 4.2: X-Y routing

4.1.2 Packet format

Packets are divided into flits which are also known as flow control digits. Flits are the smallest digits which can travel in the network through node to node.

Packet is divided into 3 types which are as follows head flit, body flit and tail flit. Head flit consists of destination address of 4 bits, and 5 bits for transmission directions (east direction, west direction, north direction, south direction and local) of source node, which are required for the connection establishment. Body flit contains all data required to transfer at destination.

We have taken body flit of 8 bits. Remaining bits are tail flit. Other resources can utilize the tail flit. We can use more bits for header but it causes increasing size of flit, which in turn will increase width of buffers.

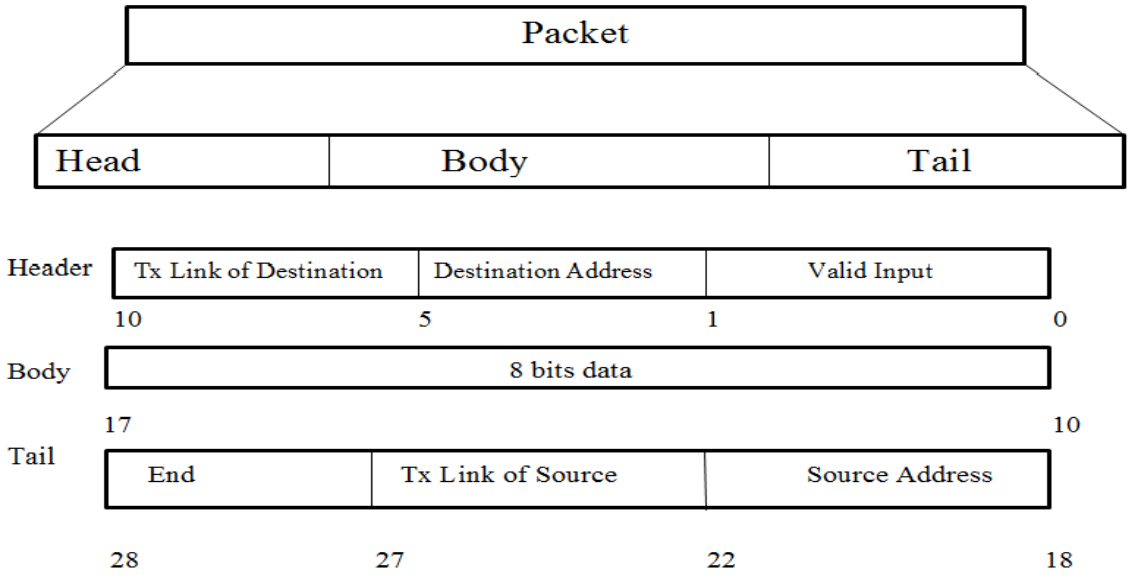


Fig 4.3: Packet format

We have considered sequence of transmission link of both source and destination as east, west, north, south and local from right to left in packet bits.

4.2 Arbiter

Each output port has its arbiter. Arbiter decides for the output port which can be used by the packets coming from four input ports. It has multiplexer, selection module, control unit and registers. The structure of arbiter is shown as below

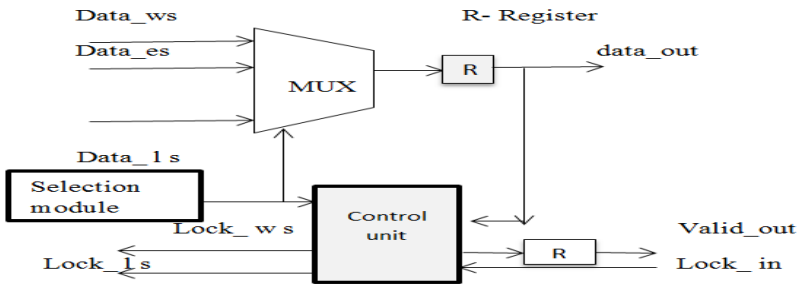


Fig 4.4: Block diagram of arbiter

Here control unit is used to provide the status of next router to which data is transmitted through output port of current router. If the downstream router is busy then it will not allow data to transmit as the transmission link of output port becomes busy and data remains where it was. Selection module is used to indicate that data in the corresponding available transmission link is now ready to be transmitted through the output port to the next downstream router. Selection module can also be used in the situation when data from multiple input ports wants to access the same output port then it will decide which input port will first access the output port.

4.3 Router implementation

When data arrives at input port then first it is stored into buffer then a valid signal is asserted to inform the allocator that valid data has arrived. Then routing computation will decide he direction to which data is transferred. Then control unit will check for the corresponding lock signal, if it is zero then data will be transferred to corresponding transmission link of input port by selecting the required link with the help of demux and control unit. Select lines are provided by control unit based on its lock input signal. Then it assigns lock out and valid signal as zero. If it is one, that means requested transmission link is unavailable then the packet will remain at the same place where it was earlier. When valid signal for particular link is assigned to the selection module of arbiter, then it indicates data is ready to be transmitted through the output port to the next router. If input lock signal for its control unit is one it indicates that next downstream router is busy, so it is unable to receive the packet. Here control unit assigns all the lock out signals as one. Otherwise it will assign input lock signal as zero, and packet in one of the transmission links will be transferred through the output port. Structure of complete router is given below

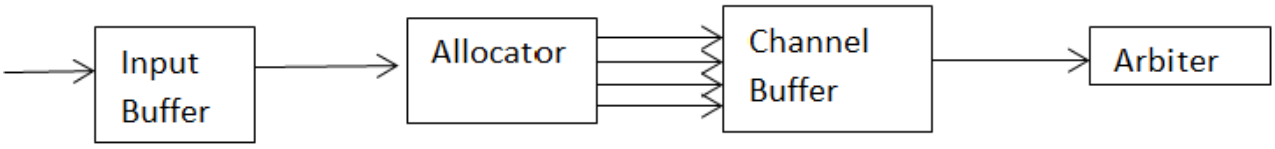


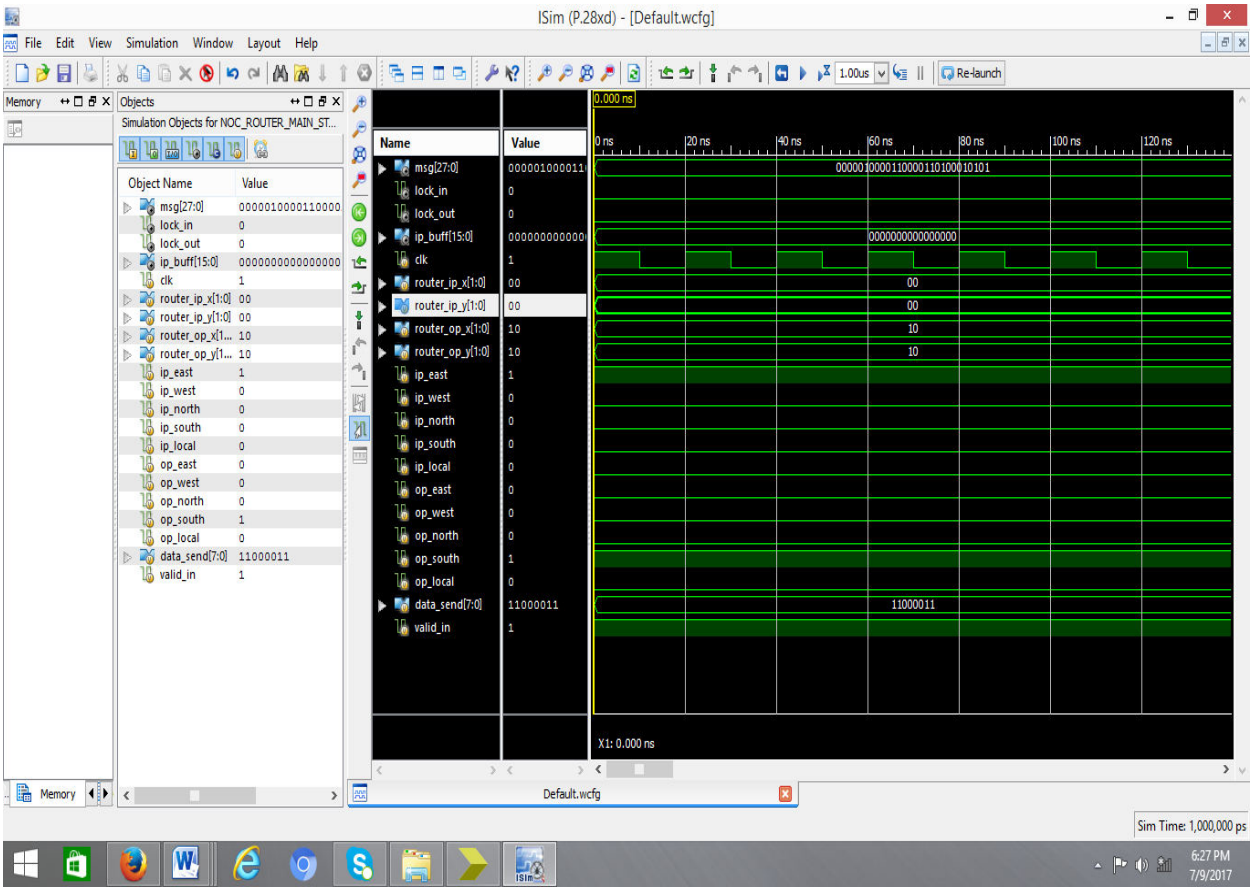
Fig 4.5: Block diagram of router

Chapter 5

Simulation results

In previous chapter we have discussed implementation of pipelined router architecture using xy routing. Here we are going to discuss the simulation results and which are obtained using xy routing and modified xy routing. We have taken an example 4X4 routing, in which we are sending data from east port of source (0, 0) to south port of destination node (2, 2). Data which we are sending is 11000011.

Step 1: here packet of 28 bits is prepared. It has taken one clock cycle of 20ns.



Step 2: Now we will run buffer and allocator module which are combined. Here we get data in 21 clock cycles that is 420ns. And valid signal for destination south port is asserted after 430ns. Parallelly by this module select lines for demux come in 430 ns.

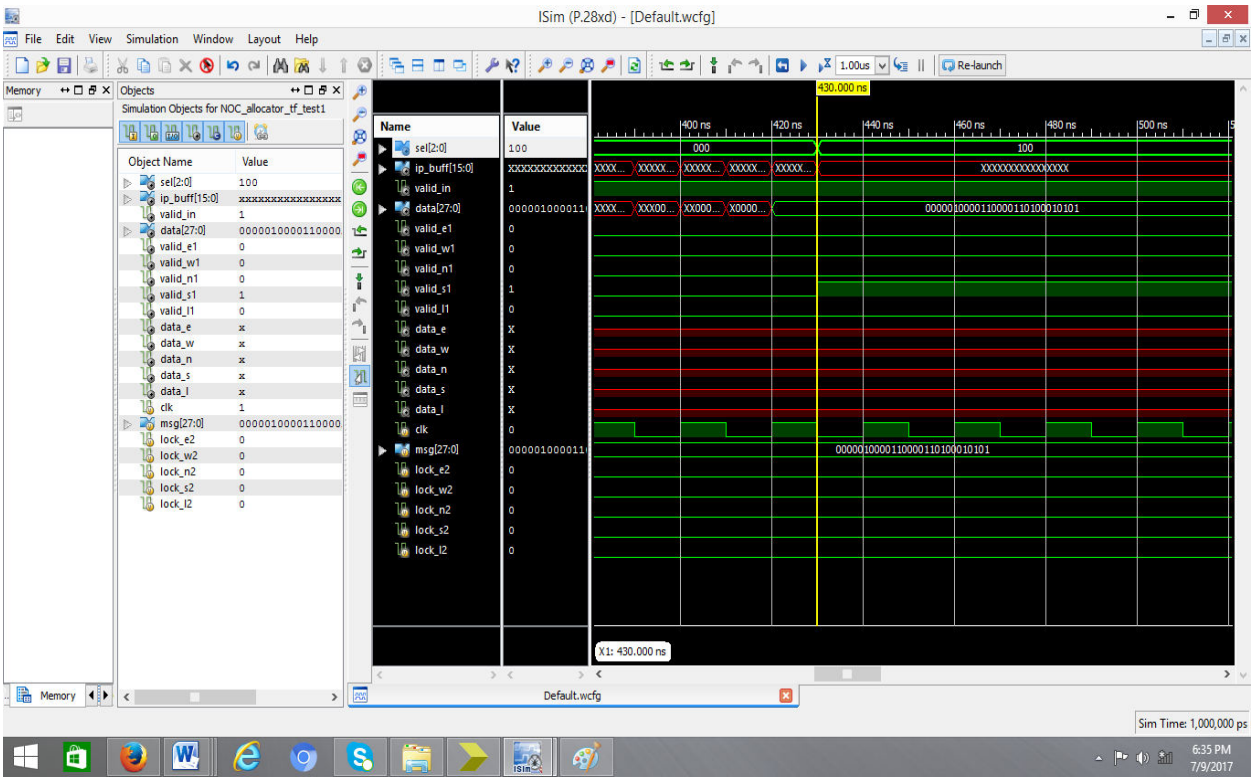


Fig 5.2: Simulation results for allocator

Step 3: Demux will give data to one of five available registors. In this example data has arrived at data_s.

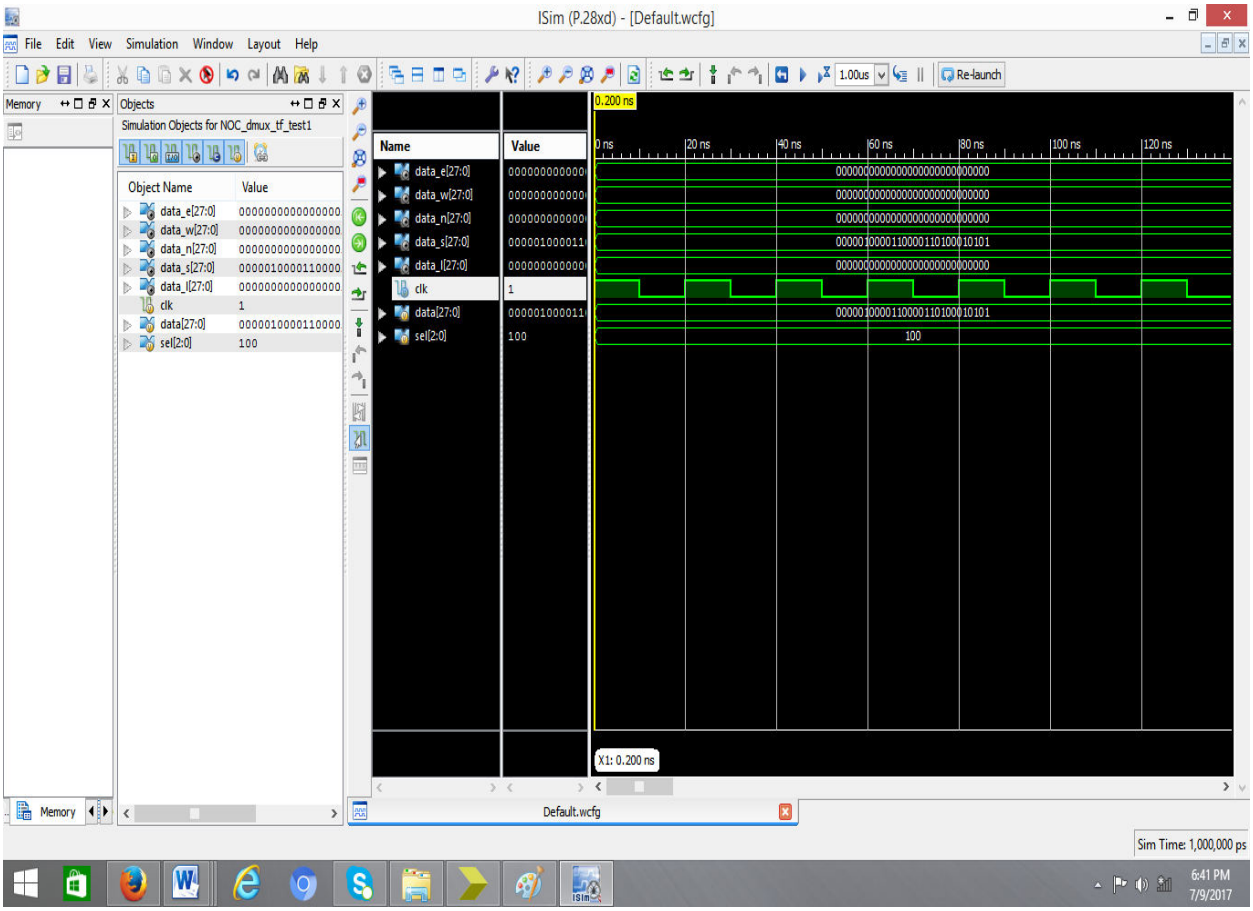


Fig 5.3 :Simulation result for demux

Step 4 : After simulating above module, we will simulate channel buffer module, which is acting as virtual connection between all nodes. This module is transferring packet from allocator section to arbiter section.

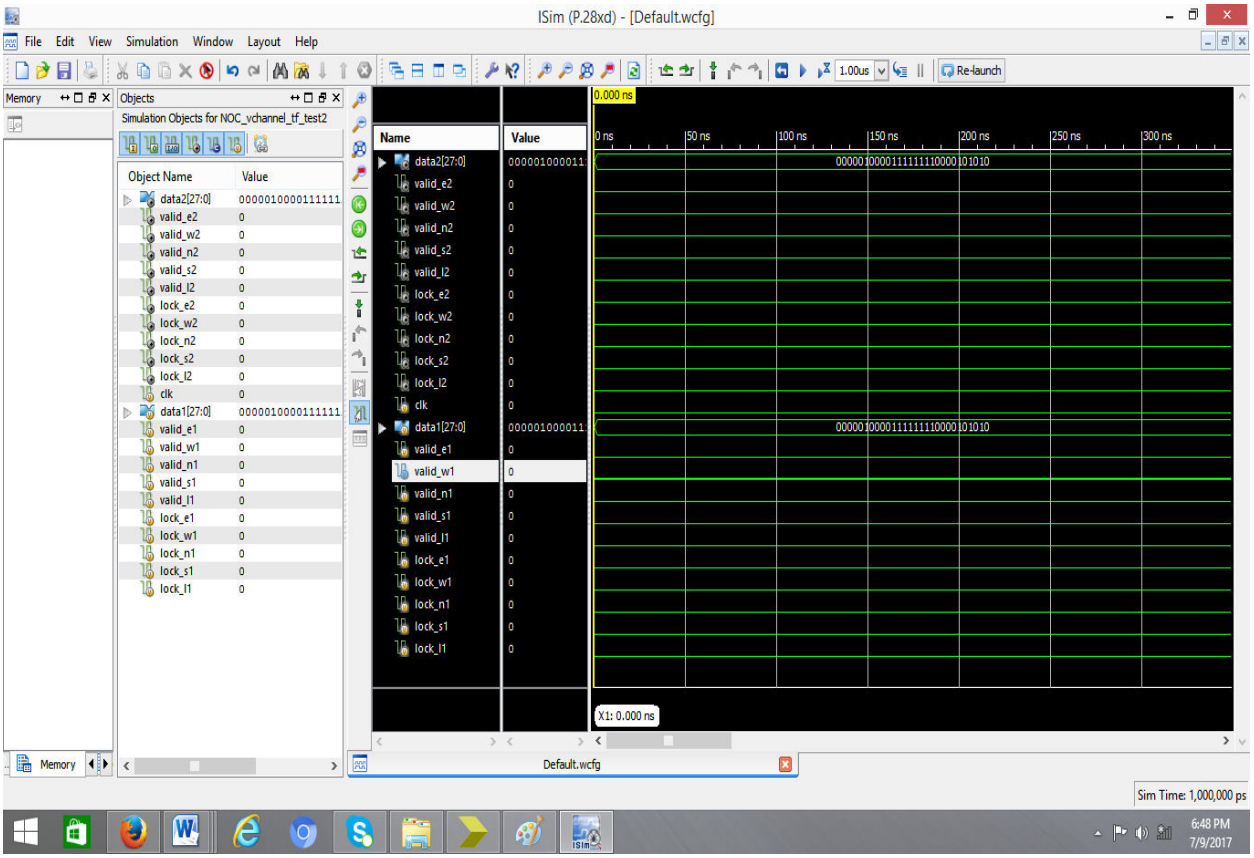
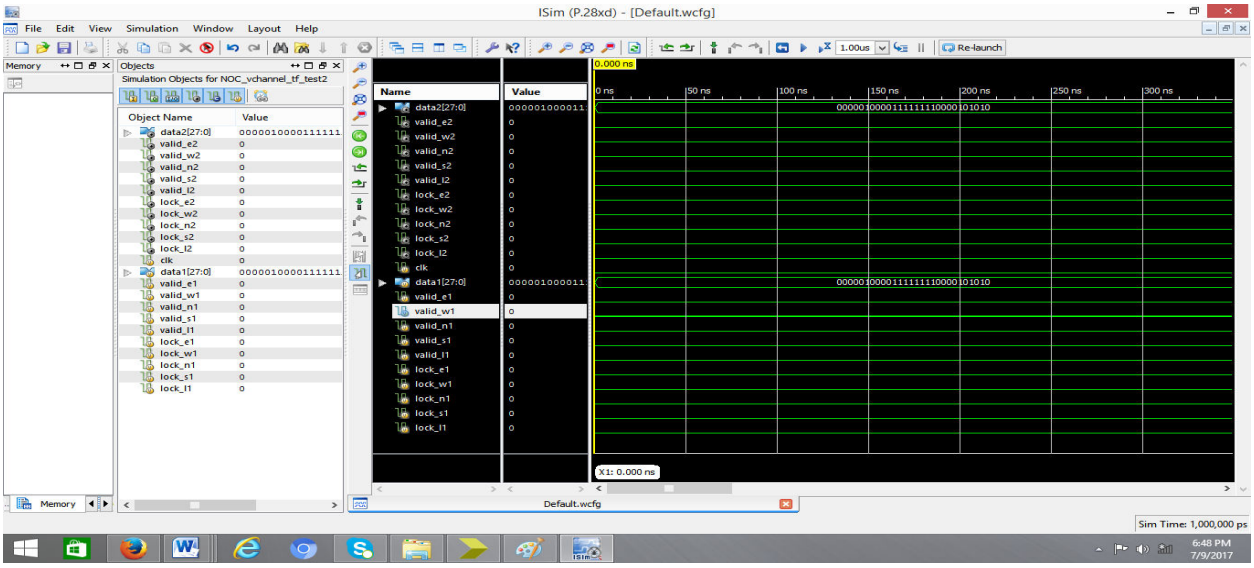


Fig 5.4: Simulation waveform for channel buffer

Step 5: After simulating above module, we will simulate XY routing module, By which data will be available at node number 10 in 40 ns that is in 2 clock cycles. Data will reach at destination node number 10 by traversing nodes numbered as 0, 1, 2 then 6.



Now we in this example we considered node number 2 as blocked, then data will traverse through path 0, 1, 5, 5 and then 10 in same clock cycles as previous as data is traversing same number of nodes. So the modified XY routing result is as shown below

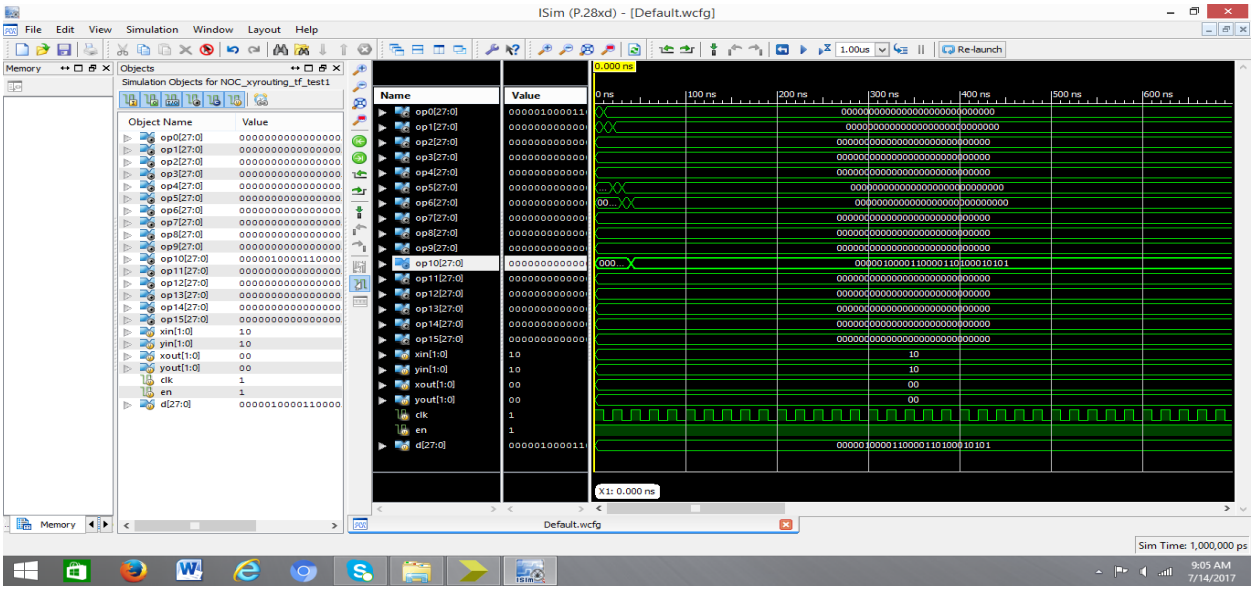


Fig 5.5: Simulation waveform for XY routing module and modified XY routing result

Step 6: Now we will simulate arbiter module which is grouped together with mux so it will provide selection lines for mux based on lock signal and valid signal. In reaching message here, It is taking 370 ns time.

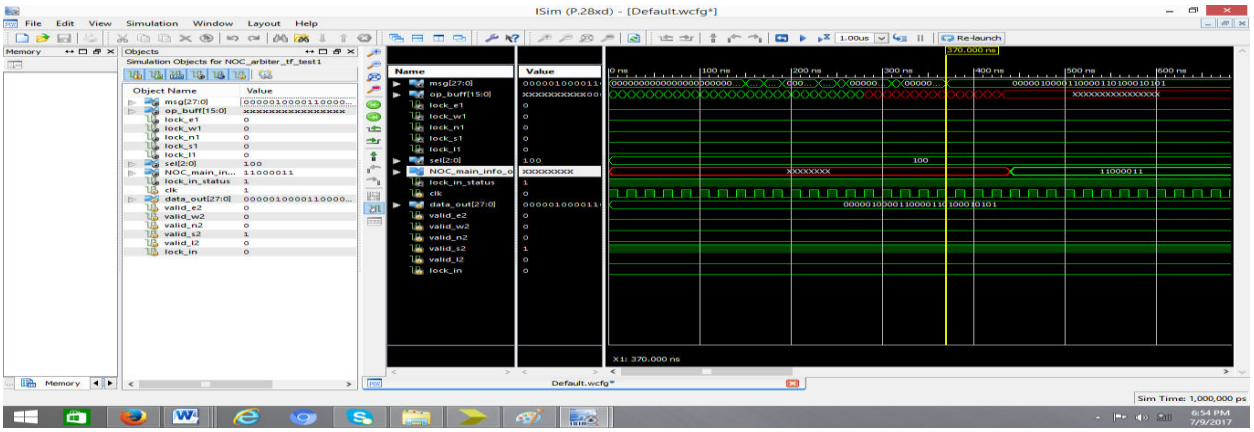


Fig 5.6: Simulation result of arbiter

Step 7: Based on the select line given by allocator, this module will transfer data to register of destination.

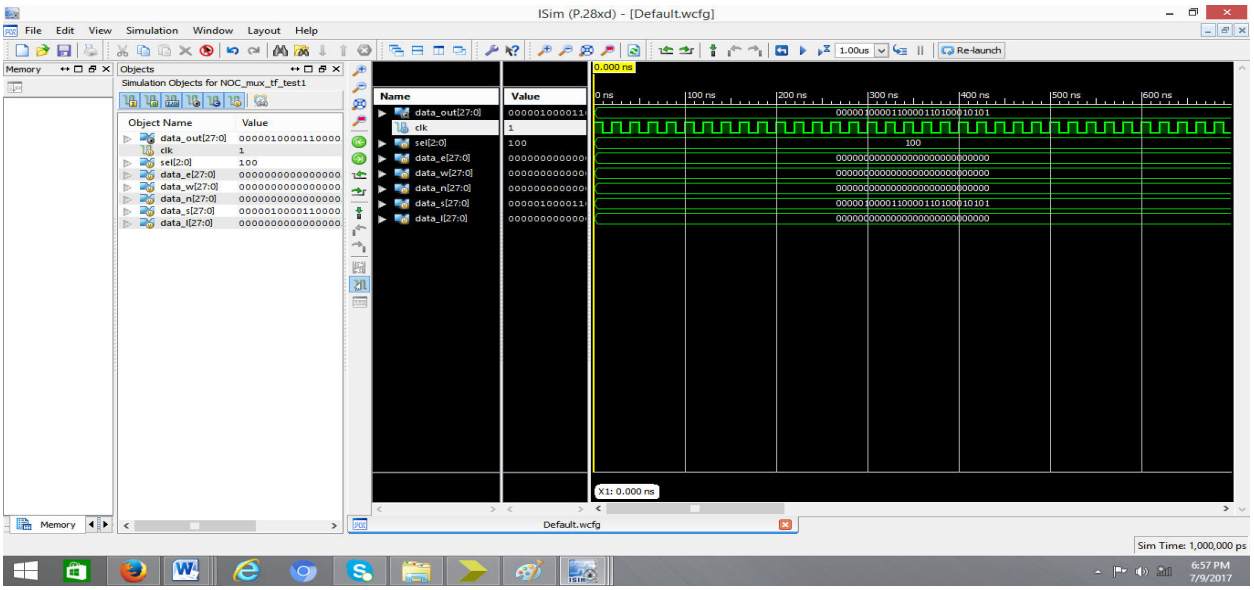


Fig 5.7: waveform showing data at buffer of destination

Chapter 6

Conclusion and Future Work

From the above shown simulation results, we can conclude that message from one router node to destination router node is successfully sent. When we consider router with one blockage node that is if data can notntransfer through that node, then we have implemented a modified algorithm which will consume more clock cycles as compared to the standard XYrouting.

Future work includes the extension of this algorith in which we can make router using a modified XY algorithm which may allow more than one nodes having blockage. So that efficiency of NoC may be increased. The implementation of FPGA may also be done. As XY routing can not provide control over congestion, So Other advanced routing algorithm can be used to implement the same architecture.

References

1. 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*, Jun. 2006.

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

3. A. Jantsch and H. Tenhunen. *Networks on Chip,* Kluwer Academic Publishers, 2003.

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

5. Springer, "Micro architecture of Network On Chip"

6. M. Stensgaard and J. Sparso, "ReNoC: A Network-on-Chip architecture with reconfigurable topology," in *NoCS'08*, April 2008, pp. 55–64.

7. M. K. Papamichael and J. C. Hoe, "Connect: Re-examining conventional wisdom for designing nocs in the context of FPGAs," in *FPGA'12*, 2012,
pp. 37–46.

8. Y. Huan and A. DeHon, "FPGA optimized packet-switched NoC using split and merge primitives," in *FPT'12*, Dec 2012, pp. 47–52.

9. J. Kwa and T. Aamodt, "Small virtual channel routers on FPGAs through block RAM sharing," in *FPT'12*, Dec 2012, pp. 71–79.

10. M. Abdelfattah and V. Betz, "Design tradeoffs for hard and soft FPGAbased Networks-on-Chip," in *FPT'12*, Dec 2012, pp. 95–103.

11. Y. Zhang, X. Hu, and D. Z. Chen, "Task scheduling and voltage selection for energy minimization," in *Proceedings of the 39th Design Automation Conference*, pp. 183–188, June 2002.

12. G. Varatkar and R. Marculescu, "Communication-aware task scheduling and voltage selection for total systems energy minimization," in *Proceedings of the IEEE/ACMInternational Conference on Computer Aided Design, (ICCAD '03)*, pp. 510–517, November 2003.

13. J. Hu and R. Marculescu, "Energy-aware communication and task scheduling for network-on-chip architectures under realtime constraints," in *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition, (DATE '04)*, pp. 234–
239, February 2004.

14. 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.