Network on Chip (NoC) Implementation for 3-D Network Topological Structure in HDL Environment

Dr. Arpit Jain

Network on Chip (NoC) Implementation for 3-D Network Topological Structure in HDL Environment by Dr. Arpit Jain

Published by Pink Petals Publication Pvt Ltd.
Naya Moradabad, B-87-Sector-6, Moradabad (244001), India.

Price: 500 | $ 20
First Edition - 2024

ISBN: 9-788195-380442
ISBN 81-953804-4-1

Cover image: https://www.istockphoto.com/

This book is available as a free download from https://pinkpetalspublications.com

Printer: Reprographic Services, New Delhi, India.

Dedicated to Mother, Father and Wife
Motivating Source of Life

Declaration

I do hereby declare that the thesis titled "**Network on Chip (NoC) Implementation for 3D Network Topological Structure in HDL Environment**" submitted to Teerthanker Mahaveer University in partial fulfillment of the requirement for the award of the degree of Doctor of Philosophy in Computer science and Engineering is a record of original work done by me during the period of my study under the supervision and guidance of **Dr. Sanjeev Sharma** (Director, JPIET, Meerut) and **Dr. Alok Gahlot** (Assistant Professor, College of Engineering, TMU, Moradabad).

This thesis has not formed the basis for the award of any Degree/ Diploma/ Associateship/ Fellowship or similar other title to any candidate of any University.

ACKNOWLEDGEMENTS

It's with great happiness that I would like to acknowledge a great deal of people that get helped me extremely through the entire difficult, challenging, but a rewarding and interesting path towards some sort of Ph. D. Without having their help and support, none of this work could have been possible.

First and primary, I wish to state my appreciation to my research supervisor, **Dr. Sanjeev Sharma** and **Dr. Alok Gahlot** pertaining to his supervision, inspire, and continually assist throughout my academic years with the University of Teerthanker Mahaveer University, Moradabad.

I would express my deep respect toward **Prof. Rakesh Kumar Mudgal**, Hon'ble Vice Chancellor, Teerthanker Mahaveer University for his motivations and support during my research.

I am incredibly thankful to **Prof. Ajay Ruhela**, Director, College of Engineering for helping me in my difficult time.

I want to genuinely thank to be able to **Prof. Bidyut Chatterjee** pertaining to helpful discussions and advices.

I want to express my appreciation to **Prof. Rakesh Kumar Dwivedi**, **Prof. S. K. Jain** and **Prof. Adesh Kumar** who took time to review our work and their enthusiasm, innovative suggestions and deep insight into technical matter were instrumental in the designing of this research work.

I am thankful to **Dr. Vaishali Dhingra,** Joint Registrar, **Dr. Navneet Kumar,** Research Coordinator, College of Engineering and CRC members for agreeing to my research work.

Last not least, although I am forever indebted to my father **Mr. J. B. Jain**, My mother **Ms. Manju Jain** and also the rest of my children **Vaani Jain** and **Vristi Jain**. My wife **Garima Jain** is a huge great source of inspiration in my experience. None of this will have been achieved without her really like, support and steady encouragement.

My parents' prayers have always accompanied me. Their love keeps me ambulatory. I am grateful to all of them.

Dr. Arpit Jian

Contents

ABSTRACT

The solution for the multiprocessor system architecture is Application specific Network on Chip (NoC) architectures which are emerging as a leading technology. Modeling and simulation of multilevel network structure and synthesis for custom NoC can be beneficial in addressing several requirements such as bandwidth, inter process communication, multitasking application use, deadlock avoidance, router structures and port bandwidth. Network on chip (NoC) architecture is an approach to develop large and complex systems on a single chip. NoC is the network version of the MPSoC. A NoC can be structured and arranged by its topology with the completer organization of the routers and cores and the approaches used to understand the technique for routing, arbitration, buffering, flow control, and switching. The data flow control is related to the data traffic or intensity inside the routers and in the channels. Routing is a techniques or method that defines the optimized path between a data or message to take place from the transmitter to the target end or receiver. In this work, 2D and 3D mesh, torus, tree and ring topological structure has been implemented in Very High Speed Integrated Circuit Hardware Description Language (VHDL). The research work focuses on the modeling, simulation and synthesis of mesh and ring topological network. The cluster size of the network is considered as (2 x 2) , (4 x 4), (8 x 8), (16 x 16), (32 x 32), (64 x64), (128 x 128) and (256 x 256) for 2D NoC design and (2 x 2 x 2) , (4 x 4 x 4), (8 x 8 x 8), (16 x 16 x 16), (32 x 32 x 32), (64 x 64 x64), (128 x 128 x 128) and (256 x 256 x 256) for 3D NoC design. The work is carried out in Xilinx 14.2 Software and modules are functionally simulated in latest modelsim 10.0 student edition software. The chip design are tested well on Virtex 5 FPGA and validated in the same hardware by the experimental work. The router is used to forward the data packets in communication network. The data packets are transferred form one source router to another

destination router in the internetwork. The communication among the nodes in all topology is done based using 2D and 3D routers. 2D router accepts the data from 5 ports such as east_input_output, west_input_output, north_input_output, south_input_output, and local_input_output. 3D router accepts the data from 7 ports such as east_input_output, west_input_output, north_input_output, south_input_output, Up_input_output, down_input_output and local_input_output. The data communication in all the NoC structures are simulated and verified for 256 bits. The router design, FPGA implementation of 2D and 3D mesh is scalable in comparison to other topology. The regular design and structure feature of the mesh NoC is the greatest attraction of all designer because there may be another possible link in case of the failure of the specific link for internode communication. Nodes are placed on the equal distance and address by XYZ routing. The hardware utilization and related parameters such as slices, flip-flops, LUTs and Input and output blocks IOBs of 2D mesh, 2D torus. 2D ring and 2D fat tree topology is increasing with the increase in network configuration or the cluster size of the network (N= 2, 4, 8, 16, 32, 64, 128, 256). In the same way, all the parameters for time values are also simulated such as min time and max time before and after clk, min period, and frequency of operation. The timing values are estimating the performance of the design. The simulated and verified results analysis is done for each topology for cluster size (N= 2, 4, 8, 16, 32, 64, 128 and 256). It is concluded that the mesh NoC has the optimal design based solution in terms of hardware resources utilization, timing result, and memory. It is also analyzed that the space or area required for the designing of mesh topology is less. One more reason is also that the topology grows almost in the linear nature with the increment in the number of the nodes in XY and XYZ for 2D and 3D. The advantage of the mesh topology over the other is due to its structure linearity and physical design. The chip modeling and design of mesh and ring topological structure is a boon for the VLSI industry because it integrates the concept of multiprocessor in a single

chip. The Register Transfer Level implementation, extraction of hardware and memory parameters and comparative study will help the designer to understand the feasibility of interfacing the future system on chip (SoC) and network on chip (NoC) based solutions.

INTRODUCTION

The chapter explains the introduction to Network on chip (NoC), Problem statement of the research work, objectives, Need and motivation to carry the research work. The research objectives contain the all supporting parameters for simulation and synthesis approach. The motivation and Scope of the research is also discussed in the end of the chapter.

1.1 **Introduction of Network on chip (NoC)**

Integrated circuits design and their manufacturing is completely depends on the integration of different sub modules which are the pre design block of the intellectual property (IP) and cores at single chip. The property of reprocessing is always resided in any ICs design. Manufacturing and semiconductors companies are working on the new challenges in the field of network on chip design and their throughput. The reuse of already developed sub modules or functional blocks is a new idea to design the circuits having high performance in shorter time period having larger gate counts. The developed design based on the discussed formalities is called as System on Chip (SoC). Traditionally the approach of System on Board (SoB) was used in which each block of the function is created and manufactured individually after that they mounted in a discrete board while now a days SoC used which has a single chip, in which all the cores are synthesized together. Later on the entire functional blocks are synthesized and manufactured in different units, and can be mounted in a discrete board. SoB is based on the already developed blocks. Moreover, the parts of SoC that can be reused are called virtual modules which are only used as the functional logics instead of fabricated ICs. It expresses a main comparison among the core based systems and traditional design methods.

The SoC core in present time have the integration of processors, networks, memories, analog interfaces, cryptography circuits,

input output devices etc. In the market, there are many companies of manufacturing IP. The functionality of the IPs is increasing day by day. The technologies are added in the products although the product time to market and design time reduced. The core based designs are extensively used in embedded based and related applications. The examples of such systems are portable medical disease equipment, Cell phones, automotive controllers and robots are suited example for the applications. The design and development time of the products depends on the technological expertise of the design, verification, signal processing, and security by encryption and decryption, RF design and analog interfaces. It is very difficult to predict that the technology has only the single house challenges. The chip size reduction, minimize power consumption, higher throughput, minimum delay, less area, reduced memory, fast response are the general requirements of these application. Moreover, it is the requirement that the all functions modules should be integrated in the system such as ADC, DAC, memory, microprocessors, mixed signal blocks etc. The business model is following the optimized design cycle to meet the requirements of application engineer, follow the RF or core solutions and focus on the basic system aspects.

It has been noticed in the manufacturing industries that the design time to develop the complex systems has been reduced because of the introduction of core based design. In the SoC based system the designers are able to address the new problems. Now the industries are focusing on design matrices such as time to market, design productivity time and they are also able to meet these challenges. But some parameters such as reliability, power dissipation and yields etc are the key issues in the current systems based on the core based on design or SoC. The testing of the ICs is one the area in which 50 % cost of the overall IC design and manufacture IC cycle is invested. In the early time it was 50 % to meet the requirements of the SoC. Since that time, the research was done to provide the efficient SoC testing and optimized architecture which can provide the mature and best plan for the testing of the complex systems.

The complex SoC designs require the system integration with parallel and pipelined approach, modular design approach, scalable architectures and testing in each phase of the design and production cycle. Then it will possible to understand the adequacy of these complex systems, solutions that can provide the use of NoC in interconnected platform.

Fault tolerance and reliability are the important keys for the future and current requirements of the SoC. It is not possible to predict NoC without SoC. The reliability of the circuits is defined by the test used and identifying manufacturing defects. The reliability of the faults is ensured by fault tolerance that is appearing in the general operation of system. First thing is that the quality and reliability must be considered of the NoC itself. The research in NoC is in the field of verifying the correct implementation, operation and manufacturing aspects of NoC. The NoC Characteristic are precluded the straightforward based on the fault tolerance and interconnections test. The first important characteristics of the NoC are that it has more wires to connect or density of wire is large in comparison to the bus architecture. Another characteristics is relating to manufacturing of the NoC. Some defects can be identified based on high density structure and the nanometer processes and high density structures especially sensitive crosstalk, some defects and large range which are common in most of the communication technologies and their architecture chip. The communication in such architecture is not possible only by wires but the multiple blocks of wires are required. In the addition of this, these wires are not accessible easily and directly as happening in bus based architectures. The nodes in the NoC are accessed using routers with the help of Network interface (NI), core or channels. So, the testing of all these parts relay on the need of each other and it makes the process much complex.

1.2 **Problem Statement**

The problem statement of the research work is given as

"Network on Chip (NoC) Implementation for 3D Network Topological Structure in HDL Environment"

1.3 **Objectives**

The objectives of research are defined to complete in two phases, First phase is the system design and second phase is simulation and verification. The objectives of the research work are given as

- Design and chip implementation of 2D fat tree, mesh, torus, and ring topological NoC
- Design and chip implementation of 3D mesh, torus, ring and fat tree topological NoC
- Functional simulation and analysis of 2D and 3D fat tree, mesh, torus, and ring topological NoC
- Verification and validation of optimal design on FPGA.

1.4 Need and Motivation

The current SoC system design and development depends on the most of the factors such as time to market and design time. The current semiconductor and computer networking companies are looking towards the fast and reliable design and solution in the field of computer nodes communication and technology using the single chip. The chip performance is estimated by many parameters such as delay, frequency of operation, power consumption, chip areas and real time system. Dozens of the functionality of the system depend on the power requirements and frequency. The power is completely related to the hardware and memory resources utilized by the system itself. Network on chip is the network version of the chip based SoC working in multiprocessor environment. When the multiple nodes want to communicate in a real time environment, there is the required of scalable

and feasible architecture which can be reprogrammed and used instead of the failure of anyone of the node. When multiple nodes are communicating in the real time, they are arranged in the specific topology. The topology is categorized as direct or indirect. The examples of direct topology are ring, tours and mesh. Mesh is the highly used topology in the NoC Communication. The 2D mesh torus topology follows the XY routing and 3D mesh and torus topology follows XYZ routing. The current manufacturing companies are looking for the dependability and reliable solution for NoC. Another important feature is that weather the network size is expandable in terms of nodes. The NoC chip time to market time depends on the design time of the NoC. If the designed time is taken a lot to launch the particular NoC in the market, then there is no use of the research and meet market time.

There are many wireless technologies such as Zigbee, Bluetooth, Wifi etc. All the wireless technologies are controlled by the wireless chip and used for node to node communication. All the wireless technology supports the specific topology. For an example the Zigbee wireless technology supports the mesh, star and cluster tree topology. When these wireless modules are placed in wireless sensor networks such as in disaster area, specific field, agriculture fields. It is required to communicate with multiple nodes. If this communication is done with a controllable chip integrated with high speed FPGA. It is one of the best solutions of the existing system working on microcontrollers based on monitoring. As discussed the wireless modules are working in specific topology. If we are able to develop the chip to control such environment, it will be a great research. Another aspect is that the existing research on chip is done on 2D topology for small scale of the network, not for a large scale networks in which we can control multiple nodes. To energies so many number of the nodes, power is required. We can also analyzed the system based on power and hardware used so we can estimate that which topology is best suited for specific applications.

Traditional systems are based on the bus communication

which has limited number of wires to the nodes and dedicated link in case of the failure the communication among nodes is not possible. If we will choose the NoC based solutions nodes have alternate path to communicate because the NoC can configured in multistage and multilayer environment. NoC are working in chip core as IP cores which have many wires

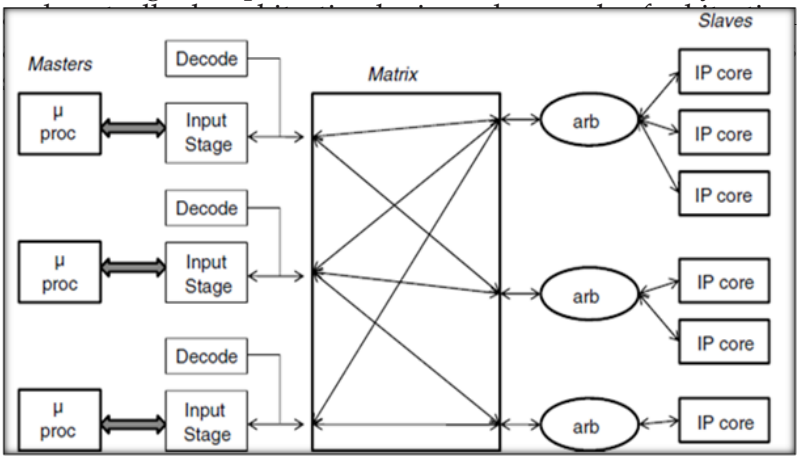


Figure 1: Matrix architecture and multiprocessor environment

Traditional systems are based on the bus based communication which has limited number of wires to the nodes and dedicated link in case of the failure the communication among nodes is not possible. If we will choose the NoC based solutions nodes have alternate path to communicate because the NoC can configured in multistage and multilayer environment. NoC are working in chip core as IP cores which have many wires and controller by arbitration logic, and example of arbitration and optimized matrix in multiprocessor environment as shown in [Figure 1].

The data flow control is relating to the data traffic or intensity inside the routers and in the channels. Routing is a technique or method that defines the optimized path between a data or message to take place from the transmitter to the

target end or receiver. The arbitration mechanism assigns the scheduling or priority of tasks or set the rules when multiple devices want to communicate with master device at the same time or multiple messages are requested by the same node. Switching is the techniques that defines how an incoming traffic is accepted by a router and send to the output port of router. In the last, buffering is the approach used to store messages or data and process while the output channels are busy. Hence large scale NoC design depends on the routing, flow control, switching and buffering.

1.4 Structure of thesis

The structure of thesis can be understood with the help of following discussions.

- **Chapter -1** is based on the introduction to NoC, Problem statement, objectives, need and motivational work and Scope of the research work.
- **Chapter -2** is based on the detailed literature survey done on the different research work based on the problem define in chapter -1 and what are the finding with the review of each paper, is continued in Chapter with identified research gaps.
- **Chapter-3** is based on the functionality of different NoC topologies. The chapter explains the basic concepts of network topology and different topological structures used in network communication.
- **Chapter 4** focuses on the designing of 2D and 3D mesh, ring, tors and fat tree topological configuration cluster size. The chapter also discusses the addressing and routing schemes for all NoC architectures
- **Chapter 5** describers the simulation and synthesis tools environment to development the design and research methodology. VHDL programming language environment is used to develop the chip.

- **Chapter 6** is based on the results and discussion part. It includes the comparisons between design and simulated parameters extracted from Xilinx software, a detailed discussion is carried out to compare the results of 2D and 3D mesh, torus, ring and fat tree to predict the optimal design.

- **Chapter 7** is based on the Conclusion based on the results and future scope of the research is also discussed in chapter 7 and the thesis is ended by with the help of the references and Appendix supporting to carrying out the research work.

REVIEW of LITERATURE

The chapter detailed the literature survey of the different researchers in the field of NoC, the findings of the survey and literature gaps are identified.

2.1 Literature Survey

The details of the some research papers are presented here.

Aurel A. Lazar in 1997 describe the comprehension on the exposed programmable networking model which is based on a fresh provision manner for the progressive telecommunication facilities that overcome on limitation over the current system of networks. This paper investigating the model will help us to clear the some major of the quarries antagonizing the telecommunication facilities industry. This paper focuses the program of switch implementation of the telecommunication networks. This paper focuses on the some important parameters like scalability, performance, Quality of Services, performance and applied matters. In this the paper exploits & tells us the benefits offered by Intellectual Properties and Asynchronous Transfer Machine technologies. It is a start of the era in programmable telecommunicating network and their reconfigurable structures.

Hiroaki Morino. et al. in 2002 describe about increasing the access use of the internet as well as the requirement of the speed up to tera byte of accessing the internet in the near future. Author shows a new paradigm of routing multilevel network for shortest path. The main advantage of this approach to make the simplicity of hardware and in this concept the switches has no buffer. A circuit of 8x8 nodes is proposed on file programmable gate array tool through which sixty-four by sixty-four circuit switch element is implemented in the one VLSI chips, and ten tera bits per second a switch

is realized by 2 phase interconnection additional the analysis work are often increased to implement a similar design on FPGA and validate the results when verify the segment communication. The comparison of the operative style and their performance parameters are often a boon for VLSI business. The performance parameters of the operative style embody minimum and most time, most output frequency, and hardware parameters admire resource utilization in terms of memory, logic gates, logic cells and no. of slices etc. in the last few years the speed of accessing the data is highly increased from kbps to mbps and now it has become mbps to tbps. Author has also discussed about the concept of single phase crossbar switch. My analysis work can target the implementation of the 3D mesh operative and different potential topological operative and compare the performance of all the operative and ring operative that underneath inter process communicating of VLSI chips. There is no buffer memory inside simple deflection routing crossbar switch as seen in multistage networks. Thus hardware simplicity is the most significant advantage of multistage network. Result of Xilinx simulation shows that this method can reduce hardware requirement in multistage networks when compare to conventional networks. Conventional close loop shuffle out switch by 10% resulting in rate of losing the packet is $5x10^{-7}$, sixty percent presented load in 64x64 switch. Switch elements of two types of circuit are floor planed on File Programmable Gate Array. Finally, the paper proposed the model for high capacity switches and the result of the simulator shows the reduction in the amount of hardware by the approximately ten percent.

Partha Pratim Pande et. al 2005 Authors suggested that the architectures based on the NoC topology are characterized by various trade-offs with regard to their specifications, performance, functionality and structure. They carried the work to compare and contrast the different architectures of the NoC to estimate the hardware parameters such as

latency, performance and power dissipation, and silicon area overhead. Some architecture are based on sustaining very high data rates at the cost of more energy dissipation and considerable overhead on silicon area, while others can suggests the lower data rate and lesser energy dissipation levels. Their main contribution lies in the illustration and establishment of a consistent and evaluation technique based on the set of quantifiable and readily parameters for NoCs.

Luciano Bononi et. al 2006 in this paper author analyse the several NoC topologies structure on modelling and simulation based. A uniform load is applied while the comparison is occurring among specially ring, mesh and spidergon topologies through the simulator. It has been observed that for the parallel processing the result is in the favour of the spidergon topology. Spidergon topology shoes the better performance and good scalability as compared to the others.

Muhammad Ali et. al 2006 as the growing complexity on the SoC a new paradigm NoC is introduced. Being introduced the new paradigm it is difficult to choose the appropriate simulator to check the efficiency of the NoC. Author used the NS2 for simulating the performance of the NoC. Authors describe the functionality of the NS2 tool.

Anderson Hansson. et. al in 2007 describe about the increasing the complexity on SOC (system on chip) with growing the intellectual properties integrated. A purely deadlock free concept is introduced that is UMARS known as 'unified mapping, routing and slot allocation'. An experiment has been applied on the MPEG decoder to judge the efficiency of the proposed algorithm, which result the reduction in NoC area by thirty-three percent and reducing in the power consumption by thirty-five percent. The author describe the introduction and related problem in section 2 and 3 respectively, and formalized in part number 4, section five consist the UMARS approach and finally section six consist the result which has been applied on the MPEG decoder. The main aim of this

paper to reduce two main things which are used to make the network totally deadlock free and completion of request constraints.

HyungGye et. al in 2007discussed the problem about the highly consistent on the system on chip due to increasing core. Author asked about the approached used traditionally that are bus system approach and point to point approach. However, the above approaches have low scalability, great complication, and rate and plan struggle. While the bus architecture can connect hundreds of core efficiently in terms of cost and complexity reduction. In this paper author perform a complete implementation on an application of multimedia that is MPEG-2 of P 2 P network, NoC and bus based topological network. The result shows that the consumption of the energy of NoC is less than the bus based and P to P.

TeijoLehtonen et. al. in 2007 emphasizes on the designing of the fault tolerance in NoC architecture. They developed several links that can tackle errors like transient, intermitted and permanent errors. Methods like hamming coding technique is used to realize the presence of transient error, interleaving method is used for error detection and Automatic Repeat Query method is used for recovery of the system. Sharing of information in the link is depend upon the signaling of two phase asynchronous while the lines of control have the responsibility to controlling Automatic Request repeat query. The functionality of the network is controlled by the control lines and the data is protected by using triple modular redundancy. Now, the developed design is judged against reference designs and co-operated. From the obtained result, it has been observed that efficiency of NoC decreases when compared to an ARQ design. But the reconfiguration structure is similar for divided communication strategy and extra cable strategy. The concept of Split communication has throughputtwenty-fivepercent, latencythirty-one percent and the latency of extra wire has fifteen percent while the throughput only ten percent.

Vasilis F. Pavlidis. et. al in 2007 describes the chip integration of 2D and 3D NoC topological designs. Several topological structures are introduced in 2-D and 3-D planes and the result has been compared on the several parameters like as the transaction speed and the power consumption. A processing element can be executed on a single physical plane. Thus, every 3D NoC has at least one PE on every physical plane and the total number of processing element on the system can be calculated as p1*p2*p3, and p1, p2, p3 represents the processing elements in the X, Y and Z plane respectively. The adjustment of nodes on 2D and 3D plane is described in this paper. A 3D NoC topology is proposed, which contains one PE. Each and every PE can be integrated in multiple planes. Thus it becomes possible to construct a 3D hybrid NoC. In such a system, both 2D and 3D NoC can stay together on one physical plane of the system. The paper tells us about the NoC topologies with the consideration of zero load model. The different parameters are compared in both the plane either 2-D or 3-D on the basis of their physical constraints. The result occurs on the basis of performance and analytic substance of every network individuallyfor the zero. In 3D-3D NoC system there is no redundancy of time. But 2D-3D arrangement can be opted for a larger network where power and time redundancy are the major barriers. By developing the 3D NoC structure the number of nodes is reduced while the data is transmitted from one node to another. The result shows that performance is improved by forty and thirty sex percent and the power consumption is decrease by sixty-two and fifty-eight percent respectively in 3-D NoC structure while the number of nodes are take one twenty-eight and two fifty-six respectively.

David. et. al in 2008 give his own views at the increasing complexity of multi core system, new paradigm introduced diverse Multi Processor SoC containing the typical intellectual properties, which transferthe data within the network at a very high speed. As the scalability of the hierarchy of buses is not

so good to embed the thousands of multi core processor. NoCs have been introduced to overcome the problem of complexity and scalability in shared bus architecture. The development of NoCs for multi-processor system on chip is very typical process that includes the different topological architecture to reduce complexity in terms of time and space. The author describes the benefit of NoCs using the synthesis flow of NoC and the analysis report on different implementation of the nanometer scale. Author presents the benefits of NoC in the field of area, consumption of the power and accessibility. Furthermore, there are also several fields are remaining to make the system more feasible design for Nano scale devices.

Mike Santarini in 2009 addressed two types of wired networks one is for computing and the second fortelecommunication. There are separate set of unique protocols for wired networks, there are different bandwidth requirement routing equipment and rate of bandwidth growth. There isa massive increase in the bandwidth requirement of telecom industry. A convergence of sorts took place, during the retooling of last wired networks which is marked by Goron brebner from Xilinx. Telecom and Ethernet both have increased their bandwidthrates. A line card is a combination of network processor unit, CPU, and the higher speed of a file programmable gate array. FPGA facilitates the communication between CPU and NPUs and NPUs are coordinated by processor to read data and route data. In future wired networks will be transferring internet data along with integration of all those IC packages.

Radu Marculescu et al. in 2009 focus on NoC architecture and its applications. Then they focused on 5 points of NoC relating to their research area which are NoC application classification, its transmission structure, transmission paradigm, their investigation, and theirsolutions for evaluation of their performance. Standard N-o-C design and its mapping to the NoC are described in this paper. They address the problems based on system, their micro architecture, and circuits.The internal structure of a node with on-chip router, processing

element and buffers are also described in this paper. In the paper it is also focused about the system level issues and infrastructure relating to communication. Major issue deals with the simulation-based methods which expresses that there is a trade-off between the simulation time and level of implementation. NoC combination movement containing the application prerequisite, NoC architecture, NoC confirmation and testing is described also.

Umit Y. Ogras. et al. in 2010 represents a mathematical model for on-chip routers which is based on the new model evolution of NoC performance and their analysis. This new approach not only gives the accurate result of the performance of the network on chip but also it provides the different NoC designing methods with in the optimization loop. They have shown the co-operated results based on the accuracy and practical consumption through their embraced result. Modeling and analysis technique are chosen for the multimedia traffic applied for the delayed studies. The described routing algorithm is a group of the buffer filled on the basis of first in first out phenomenon with 4 input packets through channels. Impending speed of router is explained through the concept of matrix which is diagonal based and the average gathering of the one packet is discussed through N input channels. The proposed algorithm also provides the feedback of the different characteristics of the network like as the average latency per router, the utilization of the buffer and the latency of the data flow. The proposed model is a work on the clock cycle which issynchronous. The router model should also consist the method to deal with the clock cycles with several domains.

Yang Quansheng. Et al. in 2010 proposed the T-mesh topology. A mesh NoC is the most popular topology which provides the complete solution to avoid the complexity on the system on chip. Still there is a back draw of transmission delay in the regular mesh NoC. The author proposed a regularTmesh topology model which is an improved version of the mesh

topology. The proposed topology consist the four more long link attached to the all terminal. Author suggested the routing algorithm based on the T mesh topology called Txy. By this algorithm the number of nodes while traverse the signal or data from one end to another end will reduce. The Tmesh is based on the algorithm Txy which make the system deadlock free through which data is transfer from one node to another node without any loop. The algorithm is not supporting to the concept of shortest path findings. That's why there exist many differences between the experimental results and theoretical results. Authors compared the results based on mesh and T-mesh from network diameter, throughput and average delay time. The experimental results show the reduction in nodes number and reduction in the average delay to reach the packet from one node to another node. Author applied the experiment for 8*8 nodes in the network and finds that there are reduction in the average delay is 2.92 percent and the reduction in nodes number approximately three percent less than as compared to the mesh NoC network.

Suleyman Tosun in 2011proposed a new paradigm on cluster based mapping from the point of view of integer linear programming. The method used for optimal solution is Integer Linear Programming (ILP) based mappings technique. Although, they suggested that method take more execution time. Authors have suggested ILP formulations method following the clustering technique. Many taste cases were tested following different custom graphs and multimedia benchmarks that were closer to achieve the optimal and closer solution within the range of tolerable period. At last after the experiment, the results indicates that the mapping on the cluster based is much better to map the different task in the distributed manner on chip network through which the accessibility become very easy and help to raise the speed of the transmission. The graph clustering methodology is same as the mesh clustering. The process of divide is repeated on the graph until the nodes of the sub graph cannot be

spitted further. To achieve maximum throughput, the graph clustering should be designed in the manner so that the flow of data among two clusters become minimum.

Ganghee. et al. in2011 explain about the 'coarse Grained Reconfigurable Architecture'. An application of multimedia and 3D graphics is used to analyze the result in this paper. The produced chips and applications drawn on CGRA show the performance and it has been found that the improvement in the performance is increase by one hundred and twenty times.

Jason cong. et al. in 2011 describe that the terms like energy and delay are needed to be more efficient while the data are transfer from one node to another node at the network on chip environment. The custom on chip network target on a given application. This is increased the efficiency as compared to the regular structure on chip. In this paper author proposed a tree topology phenomenon known as atree also written as ATNOC which helps to synthesize the network topology in which the pairs of nodes are connected through the links. From each module a router has maximum power consumption which is picked to minimize the consumption of the power and to decrease the latency. The reason to use the tree topology is that this topology is developed with shortest path, which help to reduce the path latency. A predict latency concept is prepared to predict path latency between the two intellectual properties. A method of temporal merging is used in this paper. In this paper both the models with temporal merging and without temporal merging are compared. The results off the experiments shows that the synthesize topology without using the bus and with using the bus reduce the power latency by forty-seven percent and fifty one percent respectively.

Aamir Zia. et al. in 2011 proposed an algorithm CNOC that is cloc network on chip. Authors developed the physical design and architecture for CNoC and compare it with other several topological structures on the behalf of several parameters

which used to judge the performance of a system. It is needed scalable and high performance communication infrastructure that provides the communication at low power. The paper suggested a closed Network on Chip (CNoC) architecture with the integration of the topological structure such as mesh, fat tree and flattened butterfly. The 3D NoC design was developed for 64 nodes and 512 nodes. It is analyzedthat the power consumption is increased when the size of 3D NoC was varied from 64 to 512. The system 3D- CNOC design takes 15 % less power consumption mesh, fat tree and flattened butterfly topology in comparison to other topology. The partitioning strategies, wire delay is also compared for these topologies. The 3D integration method is used to employ floor planning and partitioning and discusses their effects on wire delay. The 3D mesh structure utilizes minimum number of TSVs (9216) in comparison to the CNOC number of TSVs (10944). For the wire Length (mm) 10.5, 9.0, 7.5, 6.0, 4.5, 3.0, and 1.5, the power dissipation (mW), with respect to the same lengthis 1.74, 1.513, 1.306, 1.025, 0.8, 0.513, and 0.306 respectively, in case of plane wire. Thedelay (ps) is 287, 251, 208, 171, 128, 87.1, and 45.4 for the respective length of wire for plane wire. In the same way, power dissipation value and delay along interconnects is also analyzed. For the wire length (μm) 8, 16, 24, 32, 40, 48, and 56, the power dissipation (mW) is 0.014, 0.017, 0.018, 0.017, 0.021, 0.022, and 0.022. The delay (ps) corresponds to the same length is 0.017, 0.0212, 0.0281, 0.0453, 0.0505, 0.067, and 0.07.

ToshinoriTakabatake et. al 2011author discussed about the utilization of the several NoC topologies from the point of view of communication on the simulator. Authorpresent a new approach that is HCC known as the hierarchical completely connected. Authors compared this topology with the other network topologies and conclude on the behalf of the simulator that the performance of HCC topology is much better than the other NoC topological structure.

Ahmed Abousamra et. al (2011)asked about the concept of CMP (chip multi-processor) in which many task shared the

information from one node to another on the cache reside on the chip, so an effective transmission phenomenon is required among the cores at the chip as in the future hundreds or thousands of nodes will embedded on a single chip. NoC is the approach which provides the efficient communication among the nodes. In this paper the concept of two nodes based optical strategy used on the place of one node to make the result more efficient.

Wen-Chung Tsai. et al. in 2012 discuss the several common architecture and the techniques that deals with transmission performance, system scalability and power consumption in NoC environment. This article provides the information about architecture of the layered protocol. Author proposed the model of BiNoC that is also known as the bi directional NoC model. The number of channels between the two cores is not limited only up to two in the Bi NoC architecture. As the number of channels will introducethe performance will be increased. Every NoC router must have both hardware and software implementation as to support the functionality of these layers. The author also discuss about the Bidirectional channel that can improve the NoC performance remarkably with restructured NoC parameters. Eg: cost and power. Only two fixed unidirectional signal for conventional NoC and two bi directional channels signals for Bi NoC architecture are used while the comparison occur.

Yung-Chang Chang. et al. in 2012 authors have adopted to use 2D mesh and H-star topology to accompany a high performance switching architecture, and also proposed the work on Birkhoff-von Neumann (BvN) switching architecture. The architecture permits the resource bandwidth that depends on a particular traffic pattern. They tried to map a video object plane Authors developed system 'C' and cycle accurate model to probe further target NoC platforms into the dynamic. Author compared the experimental results and revealed that simple 2D-mesh network has traffic awareness algorithm for bandwidth allocation that overtakes H-star with

novelty and better connectivity. A real world application that is video object plane, a multimedia application also known as VOP is used.

Ciprian Radu. et al in 2013addressed on the problem of the mapping in the environment of the NoC. In this paper one operator that is mutual and two crossovers are proposed by the author. The problem of mapping is proposed in many objective ways. The objective of this paper is not only to minimize the consuming power but also make the NoC system thermally balanced. It has been proved by the simulator in this article that developed genetic operator enhance the utilization of the algorithm performance by using the domain knowledge edge on a real application.

Pradip Kumar Sahu .et al. in 2013 says that Application mapping is the new research issue in NoC. The cores of design for a particular application are mapped to the routers in NoC topology. It affects the power requirement and overall system performance. The paper focuses on the different mapping techniques employed in the last decade. The mapping technique is classified in dynamic and static. Static mapping approaches are further been classified as exact methods, transformative, constructive approaches and branch-and-bound. The performance comparison is also presented among static mapping techniques. Some test cases were generated that accepts 64 cores and 28 cores in the network. SUNMAP is a tool that is used applicable to map many cores of particular network and decides the best suitable to support the network architecture. It takes the available topology from its library targeting a particular application and synthesized the same topology. It explores the RTL design to attempt the design and minimizes design area, power dissipation and average communication delay and bandwidth. There is one of the most important compile is called Xpipes Compiler. It instantiates itself network components such as links, routers and interface specific to the topology. The greatest benefit of Xpipes is that it has highly parameterized components

based on specific structure. SMAP is used to map the network based architecture in MATLAB environment. XENoC is used to perform HW/SW design and complete the co-design to form an efficient distributed application MPSoC based NoC design. The MPS is used to targets fast and flexible design for space that performs evaluation of the mapped design used to decide greatest mapping for MPSoC based NoC design.

Amir-Mohammad Rahmani. et al. in 2013 describe that 3D NoC are used to provide lower power consumption in Multiprocessor interconnects, higher performance, improved packing density in comparison to 2D NoC. But there are some challenges which are not ignored such as power densities, peak temperatures, area footprints in vertical interconnects, higher peak temperature. The paper discusses the Bidirectional Bisynchronous Vertical Channels (BBVC) as area efficient architecture as power aware 3D NoC. Warm hole packet switching is used for data transfer and inter channel communication is also verified using VHDL based design. The simulation results revels that are footprint is reduced by 47 % through silicon via (TSV) and 18 % NoC power in their proposed architecture. For minimizing the power consumption, a concept that is dynamic frequency scaling used. The encoder for the 3 x 3 x 3 NoC was modeled and mapped. The clock frequency was decided of 200 MHz to perform interlayer communication for BBVC. The XYZ routing scheme was used. Several frequency at the particular interval of hundred are used for communication in interlayer. In comparison to the serialization methods, the suggested architecture has improved the requirements of power and average time for packet latency.

Haoyuan Ying et al. in 2013 describe that3D IC is becoming the most design issue and the 3D NoC are used to suggest the solutions based on yield, design complexity, and chip area. It can be used to minimize the number of TSVs, but it is a big challenge. The paper proposed routing algorithms which are suitable for 3D NoC environment to avoid the deadlock.

Authors proposed an algorithm for the routing based on system 'C' and simulated on GSNoC Simulator. They are also implemented using several routing algorithms using VHDL and power consumption is determined with the help of SYNOPSIS tool. The global frequency was set as 1GHz and the size of NoC to be developed of 8 x 8 in XY plane 8*8 in YZ plane and 8*8 in ZX plane and 4 x 4 in XY plane, 4*4 in YZ plane and 4*4 ZX plane respectively. All routing are deadlock free and follow adaptive routing. The proposed routing algorithms have achieved better performance in the comparison to the algorithm XYZ. The authors have proposed the methods of core placement and task mapping to get the better performance.

Eman Kamel Gawish. et al. in 2014 focused on the mesh NOC design and implementation of routing algorithm for variability tolerant. Many algorithms are proposed that provides lower probability of failure links in mesh network for variability tolerant perspectives. Different conventional algorithms like as XY, Negative first are discussedin the papers. The work is carried out in a tolerant and cycle accurate simulator NOCT weak. How the tolerant provides process variations to measure the performance of the NoCs. The result is shows that there are reduction in failure ratio of NoC is fifty six percent, while only three percent NOC failure rate is found while using the XY routing for variability tolerant. The results are obtained for a case of 8 x 8 mesh networks carrying random and uniform traffic for buffer size 16 at 45 nm and injection rate 0.1. The rate of NoC failure is increased with the increment in the number of nodes at the mesh NoC as well as there is the reduction in the size of buffer. The process variations are increased with the technology scale and the failure rate of NoC is also increased as well. Apart from it, different traffic size and patterns have different rates for NoC failure. The highest failure traffic rate has tornado, whereas the lowest NoC failure rate has traffic pattern neighbor. In the proposed tolerant variability routing uses static values

probability for link failures. The adaptive routing algorithms, such as even and odd, uses dynamic network conditions values probability for link failures.

Haytham Elmiligi. et al. in 2014 discussed that 3D NoC mesh architectures are the scalable architecture used to meet the requirements of low power consumption for multi core applications in larger scale. The research article discussed the 3D NoC applications methodology used to NoC mapping. The authors have suggested to use Genetic Algorithm (GA) is applied to understand the best mapping utilizes the minimum power consumption. The multicore architecture with 32 symmetric nodes is presented. All the nodes are running independently in homogeneous structure. The 32 nodes can exchange the data with the help of common memory unit. The core of the all the nodes is symmetric. A traditional algorithm known as Dijkstra's is used to determine the minimum path between the nodes. The suggested algorithm supports the minim power for the developed 3D mesh NoC. The proposed solution is beneficial for the design engineers to minimize the consumption of power while data transmitted in packet form with in the physical link.

Feng Weng et al. in 2015 describedthat NoC(Network on Chip) is an efficient but it experiences increasing leakage power. In order to remove the increasing power leakage, low power technique called power gating is being used but disconnection problem is one of the reasons which effect on network performance. Author proposed a way to avoid the disconnection. The asymmetrical bit splits the router into 2 different slices. Author introduced a concept that the slice may be divided into two Parts first part consume the maximum power while second part consume very less power. The slices which consume the wide range of power will cut off to avoid the leakage and slice which consumes very less power will be always active to avoid the power disconnection. By applying this phenomenon the power consumption can be avoid approximately twenty percent and as the result there are forty

four percent less power leakage. As technology scale drop down to deep submicron nanometer domain, the on chip communication is inclining on top. The on-chip consumes lots of power in multi-many-core system so it is important to use lower power technique. The low power gating phenomenon works at the level of router.

Ke Pang. et al in 2015 explore the different shapes of the NoC topology for different algorithm of task mapping. The concept of the task mapping is much effective from the point of view of reducing the consumption of the power and improving the performance of the timing. The flows of the mapping task play the important role to calculate the timing performance and energy usage on the platform of the NoC. This concept may be used for any algorithm regarding the mapping of task for any shape of the mesh topology. In this paper a 3 * 3 NoC shape is used for random benchmark mapping strategy.

Junxiu Liu. et al in 2015 proposed the ways to make optimal path selection called the EDAR algorithms. EDAR algorithms define different weight values. This paper also describe about a path routing algorithm for efficient data transfer using Efficient Dynamic Adaptive Routing (EDAsR) Algorithm. The strategy used in the paper is based on real time NoC traffic and weighted path selection approach using monitoring modules. The main objective is to maintain faulty and congestion conditions with the decisions based on effective routing. The status can be in ideal mode, busy mode, faulty mode or busty. The lowest weighted port is ranked as nearly optimized route to forward the packets. The method is applicable to by-pass the route for congested ports and diverts or tolerates the faulty ports. In the paper several traffic patterns are evaluated with the faulty or non-faulty nodes to access the throughput and latency to make fault free nodes. Under the different traffic load, congestion and injected faults, EDAR provides maximum throughput in comparison to other existing algorithms. In the addition of it, hardware area the overhead of EDAR is demonstrated in their research

to optimize the cost and design a scalable design of NoC for large scale. Multiplexers are used selected the different weights to provide the low area overhead. Adaptive routing structure is implemented using sum of different weights with preferred port, faulty, congested and busy. The output of EDAR is selected in east, west, north, and south direction to establish the connection between input traffic and output node.

Ran Manevich. et al in 2015 describe that as the number of hops will increase with growing the modular number the performance scalability of NoC topological structure will become the limited. It is happened because of the increasing in the number of hope which includes the long path as well as increase the number of routers in the path. The latency of the network is increased with increasing the number of hopes as the number of router is also increased. The hierarchical structure of the topologies may be one of the solutions to minimize the number of hopes as well as routers in the path. Author introduced a concept of active supervision of ordered NoC topologies.

Adesh. et al. in 2015 developed the NoC for ring topological structure. They followed the architecture based on rotator on chip architecture and token ring concept. The design was developed for 65536 nodes and following the concept of FIFO logic. The FIFO logic was used to provide the priority of service in the inter-process communication. The nodes were assigned the different nodes address each node can be serviced with the help of node address. To address 65536 nodes there is the requirement of 16 bit address used as source and destination address. Virtex-5 FPGA was targeted to synthesis the design and validates the results for inter process communication. The design supported the maximum frequency was 535.733MHz frequency.

Tahir Maqsood. et al. in 2015 presents quantitative assessment for mapping algorithms on NoC structure. It can be widely

vary with mesh sizes, task loads and communication system. Author concludes that 'communication aware packing based nearest neighbor' known as CPNN algorithm consumes less power. Although CPNN algorithm have higher communication overhead. Authors proposed an enhance model of CPNN technique that reduce the communication overhead. The discussed phenomenon was successfully developed at lower cost with the comparison to CPNN that is an effective solution of high scale mesh NoCs. Further the concept of patients was introduced with the NoC to carry the modeling and formal verification of designed NoC. It is concluded that it has the low cost, end to end latency, and hop count. The Communication-Aware Migration (CAM) algorithm has lesser power consumption than CPNN. Moreover, the suggested method has achieved 6% energy savings in case of smaller size mesh NoCs.

Majid Rezazadeh. et al. in 2015 describe that indirect topology can be designed and integrated in chip to support many conventional networks in multistage environment. The networks are applicable to minimize the hop count and can be integrated very easily integrated in a topological structure. In the paper authors have contributed to develop Multistage Interconnection Network (MIN) architecture under carbon nanotube technology (CNT) under synthesis traffic approach and track division. It is one of the architecture for future digital traffic technology. CNT is the promising technology for the future generation of the traffic networks, because it is capable to handle high densities of currents for a longer period without degradation in the performance, designed in next generation network chip and fabricated. Can all networks be feasible to provide a latest model as much they are efficient to provide maximum throughput? Authors have used the concept of data mining in the way to develop to develop a unified performance parameter. MIN structure provides the minimum delay, maximum bandwidth for DSP components and IP with memory units. The increasing bandwidth of the

pins can be utilized with the help of high radix routers. In a large scale network, as the numbers of components are increasing, the interconnection network provides the better performance and low cost to design the system. Clos, flattened Clos network, Benes networks, flattened Benes network, Omega network, flattened Omega network, baseline network, flattened Baseline networks are discussed in the paper. The simulation results of the paper show that CNT technology provides 30 % reduction in delay and 8 % deduction in power consumption. The throughput and message latency is increased significantly in comparison to conventional MINs.

Feng Wang. et al. in 2015 describe that in the many core technology, the main parts of the system are network on chips. It provides efficient and scalable architectures, which connects many chip resources. There is a crisis in the leakage power technology, which is further increased to scale down day by day. Power gating technology is used to mitigate the issues of leakage power. The problem of disconnection used in conventional NoC can affect the performance of network. In the research article authors tried to overcome the problem of disconnection using partial power gating. It avoids the performance loss of disconnections. Authors used the bit slice mechanism to distribute the routers into two slices. The data path architecture of the router is sliced into two parts and wider part is switched off using slices gating technique to save the power. All the narrow slices are kept active to avoid the problem of disconnection. Authors synthesized such traffic model to gain considerable amount of power at low load and performs better in comparison to the available router gated mechanism. The developed design was able to save 25.5 % power in comparison to the baseline design. It reduces the latency time by 45.00 % in comparison to the conventional power gated design. Authors claimed that power gating in bit sliced NoC technique provides better performance and power efficiency. The problem of leakage power can be mitigated with the help of power gating mechanism. Authors

have utilized the concept of FSM in modeling the traffic and synthesis their design. Bit slice technique was used to split the design into two parts asymmetrically and further partial power distribution approach was used. To construct a full subnet al narrow slice were used and gating mechanism used to save possible leakage power in wide slice. The slicing mechanism of narrowed on chip network and subnet in full connected form cannot perform the transmission of to normal packets.

Kamel Messaoudi. et al. in (2015) presented the codec H.264/AVC used for high quality video hardware chip. The hardware chip implementation is planned several codec processing elements (PE). The idea to implement codec is to provide SoC based solution so that system can work for multiprocessor system environment to integrate the different IPs. They proposed the 3 x 3 mesh architecture and then integrate the architecture with H.264/AVC in a particular network environment. The hardware architecture of H.264/AVC is implemented with the help of VHDL at RTL level. The results are simulated with the help of modelsim 6.1 software and the synthesis is carried with the help of Xilinx ISE12.2. The processing time of each block of pixel is calculated using simulated results. The overall system is designed using EDK tool on XUPV5 Xilinx support board. Same technique can be applied for inter decoding and inter prediction with a global system. The intra module was subdivided into two modules intra four by four and intra sixteen by sixteen prediction modules. They have used the inter prediction to provide connection in 16 x 16 module. The different modules for the synthesis H.264/AVC IP are 3 x 3 mesh NoC, intra 4 x4 prediction, intra 16 x 16 prediction, DCT & Quantization, inverse Quantization, inverse DCT, deblocking filter, Communication system (NoC + NIC) and IP resource for 3 x 3 NoC. The number of slice registers are 127/69120, 1477/69120, 2650/69120, 1821/69120, 1865/69120, 6702/69120, 2540/69120, and 14443/69120 with respect to each discussed module. The maximum frequency

of the utilization is 172.828 MHz, 278.431 MHz, 282.028 MHz, 594.107 MHz, 394.058 MHz, 188.638 MHz, and 165.781 MHz respectively to each module. The number of slice LUTs are 2154/69120, 1703/69120, 2180/69120, 1603/69120, 1712/69120, 8890/69120, 3226/69120, and 17242/69120 with respect to each module. Fully used bit slices are 5.89 %, 38.02 %, 49.33 %, 59.49 %, 38.70 %, 29.20 %, 0 % and 37.10 % respective to each module. BRAM/ ROM utilization is 0/148, 4/148, 0/148, 1/148, 1/148, 14/148, 6/29 and 29/148 with respect to each module. In the similar manner the number of DSP48E are 0/64, 0/64, 0/64, 16/64, 16/64, 0/64, 0/32 and 32/64 corresponding to modules respectively.

Prasad. et al. in (2015)proposed the Network on Chip (NoC) base d Fast Fourier Transform (FFT). The FFT is following the constable Geometry FFT transform (CG-FFT), parallel data, can process up to 8192 point reconfigurable structures. The large numbers of points are accumulated with the help of twiddle factor of the FFT and CORDIC algorithms. High throughput of the design is achieved using pipelined rotators of CORDIC. The signal flow (SFG) is used to under the different modules of FFT and help in the routing and network design. The latency of the FFT computations is reduced using SFG and optimizes the design of routers and network interfaces. The design was mapped to the mesh NoC and provides latency by 5 in the comparison to existing NoC architecture. The hardware utilization of the processing elements and different components of the network are synthesized on Kinetx-7 FPGA. The maximum frequency of the design of a PE in the suggested architecture was 184.010 MHz. The frequency meets the standards of timing parameters for DAB, 802.11a, DVB-T/H, and UWB. One PE with 8bits data width can occupy 2.595 mm^2area and consumes total power of 82.540 mW. The operating voltage is 1V to achieve these results. In the same way 4 port routers correspond to 16 bits data width was synthesized. The area occupied was 0.097 mm^2 and power was 15.717mW at 500 MHz frequency. The proposed

architecture has been designed with the help of VHDL and variable data rate is used to apply the same concept in several other applications.

Tassadaq Hussain in 2015 describe that the memory controller and the scheduling of the memory access are very much important in a multi core system. The system follows multiple memory units and task that enhances the delay, which affects the overall performance of the system. An efficient high speed scheduler and memory system is needed that is helping data access and its pattern to main processing core. In the research work authors have proposed Heterogeneous Multi-Core Memory Controller (HMMC) which is used to manage memory and scheduler. It can handle computational task and data movement. HMMC is coupled with the heterogeneous system that is used to provide the both application specific accelerators and general purpose cores. The designed and developed HMMC is tested and synthesized on evaluation board of Xilinx XC7VX485T FPGA VC707. HMMC can support regular and irregular access of memory. The experimental evaluation of the board supports the hardware utilization in which Xilinx Micro Blaze multi-accelerator and Xilkernel(RTOS) supports Multi-core System hardware resources. In the comparison to the baseline system, it is found the hardware utilization in HMMC is less by 43 %, dynamic power is less by 35.8 %, and speed improvement is 6.8 % and it can run multiple applications in real time.

2.2 Research Gaps

From the literature survey of the different authors it has been identified that the work has some gaps identified as

- NoC modeling is done for crossbar structures for 4 x 4 mesh NoC, 5 x 5 NoC etc. The research work is lagging with the implementation of the same mesh architecture that can support the functionality of 256 x 256 NoC structured

model or at a larger scale.

- In the same way, the ring NoC is also implemented for the limited number of the nodes and not explored in much way.

- The chip implementation of 2D NoC is done in large scale. There is very limited contribution to 3D NoC Chip and the performance estimation in terms of hardware resources and timing analysis.

- Further the research work can be enhanced to implement the same architecture on higher scale FPGA and validate the results after verify the internode communication.

- The comparison of the NoC design and their performance parameters can be a boon for VLSI industry. The performance parameters of the NoC design include minimum and maximum time, maximum output frequency, and hardware parameters such as resource utilization in terms of memory, logic gates, logic cells and no. of slices etc. My research work will focus on the implementation of the 2D and 3D mesh, torus, ring and tree NoC for large cluster size (N= 2, 4, 8, 16, 32, 128 and 256)

- The authors have not implement the chip of 2D and 3D mesh, 3D torus, 3D ring and 3D tree Topological NoC to support the large scale network (256 x 256 x 256) and perform internode communication

- The comparative analysis is not done yet to estimate that which topology is best for the future computer networks, wireless sensor network and existing wireless topology.

- Nodes communication and priority assignment or arbiter design for the 3D NoC for large scale networks is addressed by very less number of authors. Some discussed but for low size of networks only.

NOC ROUTER AND TOPOLOGIES

The chapter explains the 2D and 3D router functionality and its use in the configuration in NoC design. The chapter also explains the detailed description of Mesh, Torus and Ring topology.

3.1 Router Architecture (2D)

The router is the device used to forward the data packets in communication network. The data packets are transferred form one source router to another destination router in the internetwork. Router connected to two or many data lines inside the network to transfer the packets to destination. When the data packets are arrived on these lines the router read the address of then destination network to deliver the data packets. In the routers can transmit the packets from one network to another network in this way. The common used routers are small office routers and homes which are passing simple IP based routers between the internet and home computers. The example of a router can be DSL or owner's cable that provides the connectivity to the internet through internet service provider (ISP). Some examples of the more safe and sophisticated routers are enterprise routers which are helpful to connect large ISP networks and large business networks as power main routers that can forward the data at very high speed rate in the direction of the optical fiber lines along with internet backbone. The routers are designed based on dedicated routing and these are dedicated hardware device. There is the existence of software based routers also.

NoC provides the architecture and communication model in which multiple nodes can communicate and utilize the resources. Due to this reason, it is possible to design and implement the chip working under independent resource utilization in one block and multiple blocks as the working

node or processing element of the communicating network. The flexibility and scalability in the chip design is cable to provide the feasibility to configure the NoC with different cluster size and workloads.The NoC architecture consists of the multiple resources which are designed and arranged in particular topology. The common and important topology used for the NoC is mesh topology. In mesh topology the local interconnections are existing between the switches and resources. The interconnections are not depending on the communicating network cluster size. The routing of the 2D network can be done in easiest way and gives the overall scalability, large bandwidth and short period clock cycle. The NoC have the switches are resources which are interconnected in such a manner that they can communicate directly among each other using packet data transfer technique. The resource is the computation and storage unit. The NoC switch is used to provide the path or routing to incoming traffic and buffered the same traffic between the other resources. There are many inputs and output channels or interconnections through which the switch can connect to other neighbour's switches. Each interconnection channel has two or more buses in one direction between two switches as point to point connections.

The incoming traffic is handled using wait and go technique for queue and controlled by switch. The size of mesh network depends on the clusters and their associated memory. The fabrication technology can change chip size. The size of chip depends on the resources utilization and number of hardware resources. The applications and system volume demand the larger bandwidth but it is the challenge to the design engineer to provide the chip design and related communication in the allocated or associated bandwidth.

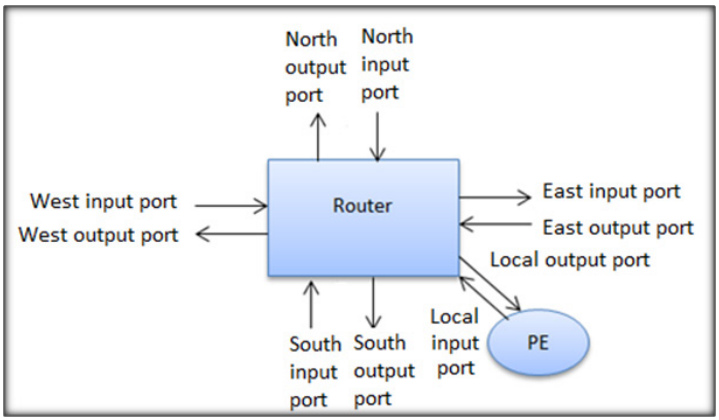
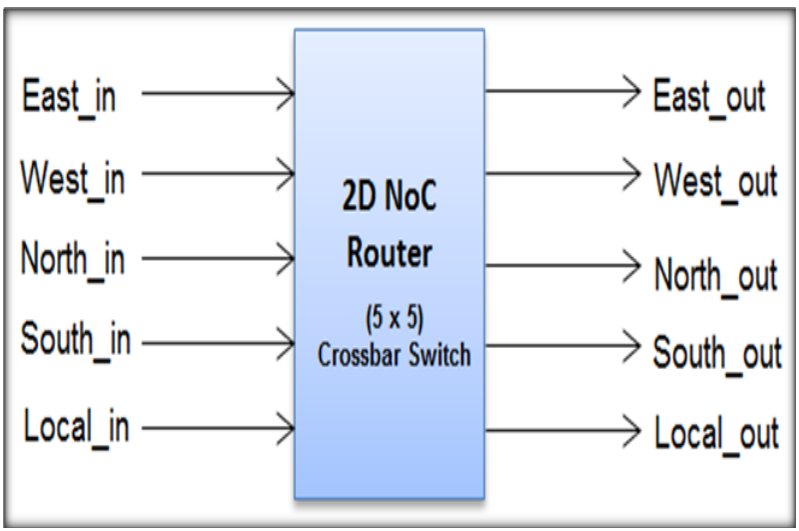


Figure 3.2 Crossbar switch of 2D NoC router



All the processing elements and the routers in NoC architecture are connected with the help of several connecting wires. The processing elements are processing the data under flow control technique with fixed length. The 2D router is shown in [Figure 3.1]. It can accept the data for five distinguish ports as the part of core: east port, west port, north port, south port and local port.

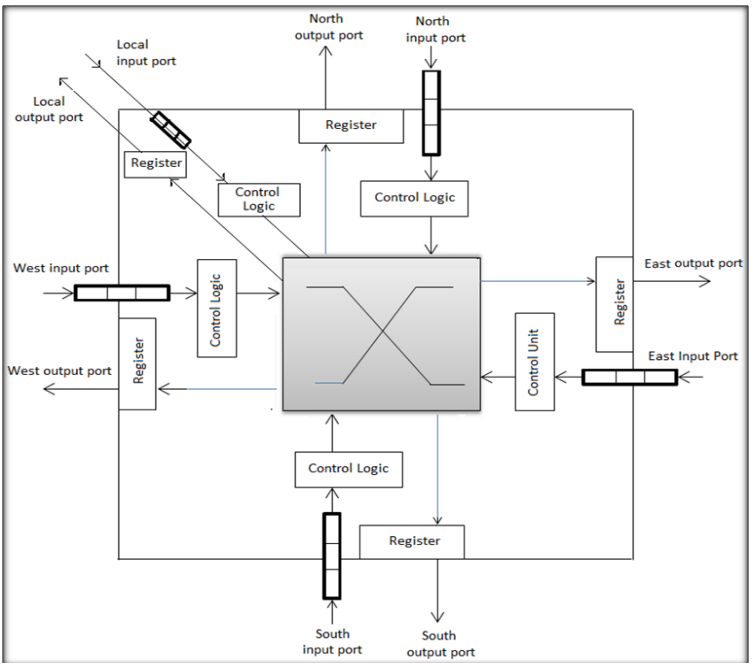


Figure 3.3 Router architecture in 2D configuration

Moreover all the ports has bidirectional nature, can be configured as input and output to send and receive the data. The ports are named as: east input port, north input port,west input port, south input port and local input port as inputs, east input port, north input port,west input port, south input port as outputs. The 2D NoC router (5 x 5) crossbar switch is shown in the fig. 3.2 which depicts the input and output inputs of the 2D NoC. The architecture of 2D router is shown in fig.3.3. The

data is coming from east input port, west input port, north input port, south input port and local input ports in the packet form and connected to the control unit which decide the size of the data packets need to transmit. The data coming from the east output port, west output port, north output port, south output port and local output port is stored in their associated registers. The design and hardware implementation of the router is done based on the policies, routing and deal with packet collision. The router is also consisting of logic blocks which is helpful in the implementation of the flow control, routing, policies and details the full strategy for the data transfer in the NoC.

3.2 NoC Router with 3D

The crossbar switch (7 x 7) for 3D NoC is shown in fig. 3.4. The 3D router accepts the seven inputs and seven outs as input and output directional ports.

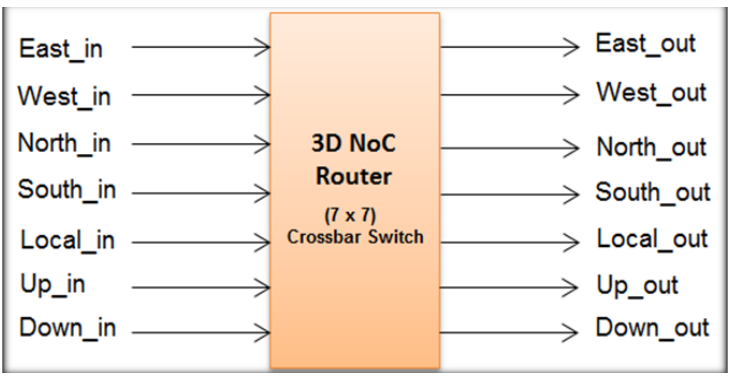


Figure 3.4 Crossbar Switches (7 x 7) for 3D NoC

The data is coming from down input ports, south input port, east input port, north input port, local input port, Up input port, west input port in the packet form and connected to the control unit which decide the size of the data packets need to transmit. The data coming from the east output port, west output port, north output port, south output port, local

output port Up input port, Down input ports is stored in their associated registers. The architecture of 3D NoC Router is shown in fig. 3.5 that depicts the functionality of the NoC design.

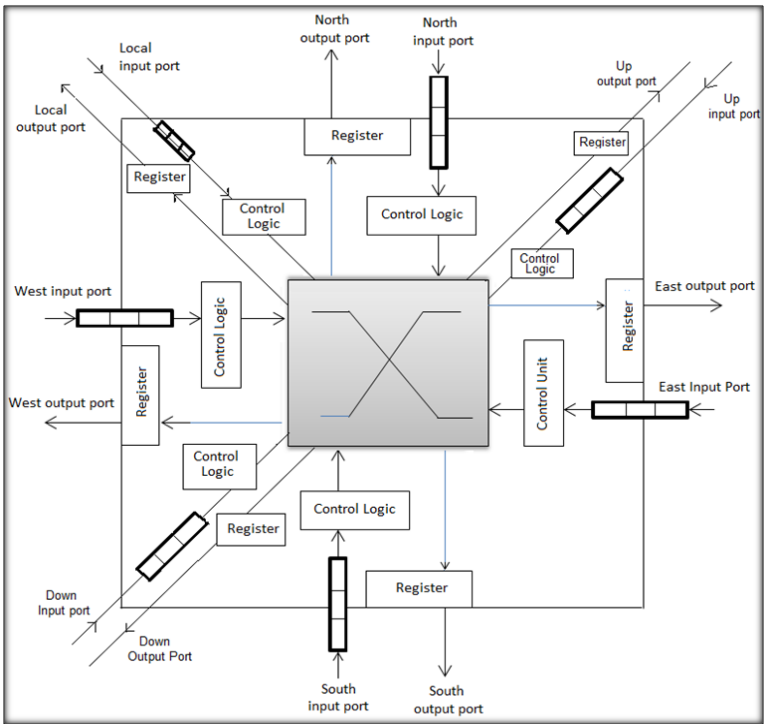


Figure 3.5 The Architecture of 3D router

3.2.1 Flow Control

A flow control is the method for packet movement along the NoC because it is involved at both level NoC chip level and local router level. It is possible to do flow control and judge the deadlock free routing for specific measures to avoid certain paths within the NoC. The optimization in the NoC depends on the channel requirements and bandwidth and it can guarantees the requirements and need of flow control. The selection technique of the routing algorithm

reduces the critical path and traffic congestion is minimized by implementing virtual connections. The quantity of the communication infrastructure and performance is called the Quality of Service (QoS).

The control mechanism is NoC can be classified as centralized control or distributed control. The routing decisions are taken globally and applicable to all the nodes associated in NoC, followed with a strategy that guarantees that traffic contention is not affecting the NoC performance. The approach avoids the bus requirements that all nodes are sharing in bus in a common time. Time Division Multiplexing (TDM) approach is used in which each packet is concern to frame. The NoC uses the distributed approach in which each router can take the decision locally. Virtual channels are very important for the flow control in NoC. These channels are helpful in multiplexing a single physical channel over different or separate logical channels associated in single and independent buffer queues. The actual use of the VC is to implement the NoC to enhance the performance to avoid deadlocks, increase traffic handling capacity, and optimize wire usage. The situation of the deadlock may occur when multiple request sate coming to service node and network resources are fully busy and the nodes are waiting to each other to be that the connection will be free or nodes are waiting to release the connection and proceeding with the communication entity. It is also possible when the two communicating paths are blocked in cyclic manner. If the status of the resources is changing time to time, live lock may occur but no guarantees of communication may be successful.

3.2.2 Routing

Routing in NoC is very important to suggest the optimal path. The routing algorithm, decides the output port for the forward packet when it arrives form the source router to the destination router. The destination port is decided based in the routing information carried by the packet

header. There are many routing techniques used to address in NoC, each one having several tradeoff between cost and performance. The same path is used by the data packet in case of deterministicrouting, when communicating between two particular nodes. The general deterministic routing schemes used are XY routing and source routing. Source core are used to specify the route to destination in source routing. In case of the XY routing technique, the row and column approach for routing is used in which the packet is moving towards rows first and then towards column or vice versa is also possible. In another routing technique, alternative paths are possible to communicate among different nodes, in case of original link or path is not available or route is congested. This routing is referred as adaptive routing. This routing is used in modular approach based design for large scale networks in which multiple chances may occur of failure the original links and traffic may be delayed or blocked due to the discontinuation of the original link.

The link load is evaluated using dynamic evaluation technique and follows the strategy based on dynamic load balancing. The other examples related to adaptive routing methods are Negative First (NF) algorithm and West First (WF) algorithm.

The static routing is also one of the important routing that provides the path between the different cores and depends on the time required in the completion of the application, but in case of dynamic routing the routing path is depending on the run time required in the completion of the communication. The data packet can have a single target to deliver the data, called unicast routing whereas one data can be routed to multiple nodes simultaneously with the help of multicast routing technique. It is similar to bus communication or the concept used in master –slave node communication. In the same way, a broadcast communication has the destination for all the nodes but narrowcast communication is started by a master node and associated with single slave. The routing methodis also said as minimal routingor non-minimal routing.

In case of minimal routing, it is suggested that the shortest path is always decided. In this routing technique a boundary box exist virtually and validates that the decreasing paths for source node to the destination node are valid. On other side, the source to destination distance is increasing based on the non-minimal algorithm. Routing methods and algorithms are very much helpful to prevent the conditions of live locks and deadlocks. The turn model is also one of the routing concept that is also prohibited by certain turns may haveincrease the risk of deadlock in the network cycle. The network locations are restricted with the help of even odd concept followed in turn model by some types of turns are possible. Another routing algorithm used in networks is hot potato routing. In this technique the data packet is forwarded immediately in the direction of the path which has minimum delay or shortest route for lowest delay. If the packet is not considered by the network, the same packet is returned back to the source router. That's why this routing is also called deflective routing. The packets are not in the buffer and each packet is having multiple set of outputs preferred by the packet itself and used in forward direction against each possibility.

3.2.3 Arbitration

In the NoC communication it is possible that the multiple requests are arriving to the service node. The arbitration logic is used when the routing algorithm decides the output port for packet data transfer and multiple packets are arrived at the source router simultaneously and requesting to provide the service. There are multiple options to implement the logical arbiter. It can becentralized with one per port or distributed one per router. The execution of the operations is taken based on the priority assignment. It may be based on dynamic priority and static priority among the different ports. Router switching matrix is associated with the arbiter and central arbiter can optimize the utilization of router switching matrix. Such arbiter can enhance the latency whereas the distributed technique can reduce the latency. Arbitration

model is also decided whether the network is following the loss communication model or delayed model. There are the chances of the delaying the packet in the delay model but the packet will not drop out. It is possible that the packet may be dropped in case of loss model. In case of heavy traffic or congestion the packet may be dropped in loss model. In such situations retransmission logic should be implemented. The main function of the crossbar switch is to provide arbitration that provides the solution for conflicting requests occurring for the same output. The speed of the operation directly related to the scheduler and delayed by the arbiter. The designing of the faster arbiter is the challenging and the integration in NoC is very important. The block diagram of the arbiter logic is shown in fig. 3.6. The arbiter has the multiple inputs coming in the form of packets and assigned to the FIFO, logic which assigns the priority. The Requests and grant signals are associated with the crossbar scheduler and directly to arbiter that handles the requests from all the directions in 2D and 3D NoC. Part from that the virtual channel allocation and routing mechanism also the integral part of the arbiter.

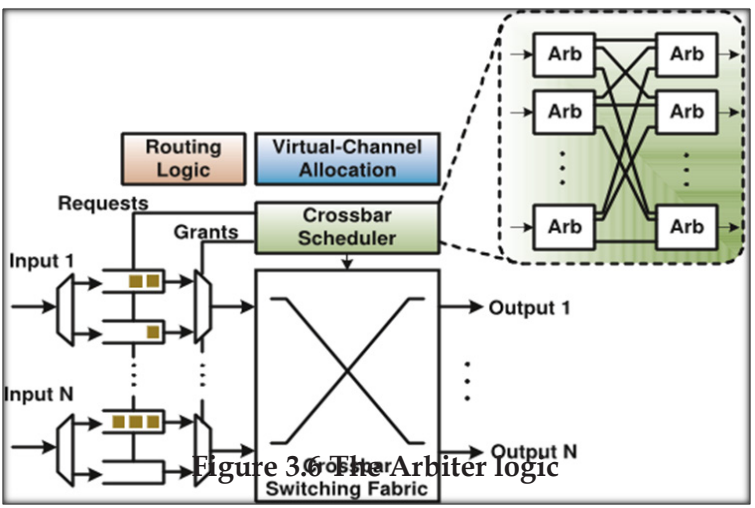


Figure 3.6 The Arbiter logic

3.2.4 Switching

The switching technique decides the way to transmit data from source node to the target node. The entire path is decided already in circuit switching method. The path is included as channel path, routers or whole, established by the header and reserved for sending the complete packet. The payload is not transferred till time the completer path has been reserved. It can enhance the latency. But if the path is decided, the approach can provide some guaranteed throughput. On the other hand, in the packet switching all the flits of data packets are transferred as the header to make the connection among the routers. In this technique the designer has the freedom to decide different forward and buffering strategies than can impact on the overall NoC by implementing the operation concurrently or in parallel pipelined mode by sending the flits. The connection is established to the next routers in pipelined mode. The nodes store the full packet before staring transmission to the next node in defined path. It is the part of store and forward technique in NoC transmission. It should be ensured that the buffer size for each node must be enough to store the full packet

On the other side, in wormhole strategy the nodes have the freedom to take routing decisions and sending the packets to the destination as the header arrives. In the subsequent way, the flits accept the header as it arrives. It reduces the router latency but many links may be blocked due to packet stalling. So there is the involvement of risk factor. Similar to wormhole technique, there is another mechanism called virtual-cut-through in which data is filtered first to the next node in the path before forward and node is waiting for the full packet acceptance by the next node and confirms the complete reception of data. Thus in case of stalling the packets, only current node is affected and other links are safe.

3.2.5 Buffering

The buffering is the methodology used to store the router information in case of traffic congestion and the packet is not reaching directly to the network. The buffer size, buffering strategy and location has very important impact on the traffic intensity. So, the NoC performance is affected by buffering. The router area is also dependent on buffer size. Some routers can have one buffer shared by all the ports and one can have multiple buffers associated with each input and output ports. The biggest advantage of the first concept is that it provides optimized area but the control is very much complex and extra time is required to deal with buffer and overflow. Each port is having its own buffer and FIFO logic is used to implement although other techniques are also available to implement the logic. The techniques may be less efficient because different input ports may require the data storage in single structure.

3.3 NoC Performance Parameters

NoC performance can be evaluated using three parameters.

Bandwidth

Through put Latency

Bandwidth: the bandwidth is called the maximum data rate required to send the messages in network. The unit of the bandwidth is bps. It has the complete packet which includes the bits of header, payload bits and tail bits.

Throughput: throughput is defined as the maximum volume of the incoming traffic considered by the network. It is considered as the maximum amount of information delivered to the load in per unit time. The throughput is measure the message per clock cycle or message per second. One network can have normal throughput which is independent from the size of the message of the network distributed by the size of message and by the length of the network as a result when normal throughput is considered per node or per clock cycle or bit per second.

Latency*:* latency is the time required in elapsing between the start of the transmission of the data packet or message and complete receiving at the destination node. Latency is measured with respect to the time unit and mostly following the comparison between different design choices. Latency is also described in terms of simulator clock pulses or clock cycles. In normal way latency is not considered for single packet but average latency is considered to judge the performance of complete network, on the other side when all the messages are presenting larger latency then the average latency become important. That is why the standard deviation is very important to measure against the network.

TOPOLOGY DESIGN AND INTERCOMMUNICATION

The chapter focuses on the designing of 2D and 3D topological configuration with different cluster size. The chapter also discusses the designing of 2D and 3D mesh, torus and ring topological structure and their behavior in intercommunication.

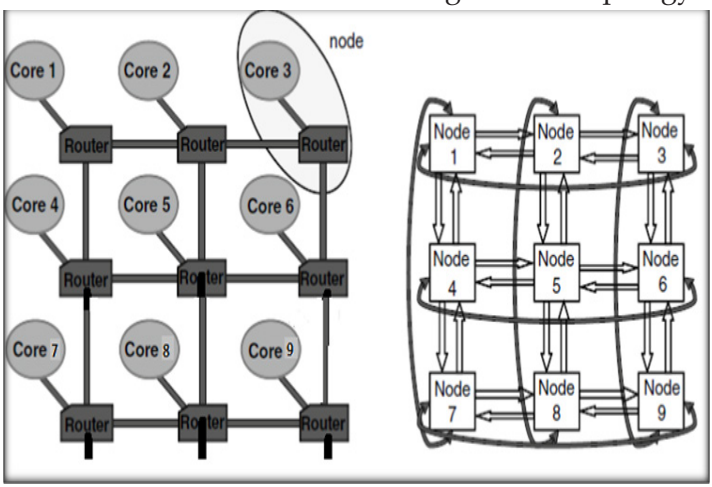
4.1 Introduction to Network Topology

Many applications need the architectures which are based on bus topological structures and bus based architectures may be used to prevent the performance of these systems, as there is the increment in Systems-on-Chips (SoCs) based IP modules. The systems which are generally use bus based communication, are not able to meet the requirement of bandwidth, power consumption and latency. Network on Chip (NoC) is the solution for such communication based system, which is a bottleneck for an embedded switching network to interconnect the different IP modules in SoCs. In comparison to the bus based communication system, the bandwidth and design space is larger to maintain the arbitration mechanism and routing algorithms and their implementation strategies with different communication infrastructure. Moreover, NoC are very much helpful for fault tolerance and enable to SoC design engineers to search the suitable solutions for several system constraints and characteristics. The NoC is characterized by different structure and routers connection. The way of connected the different routers and their organization is called the topology and can be represented on the graphical forms. Topology graph G (N, C) in which

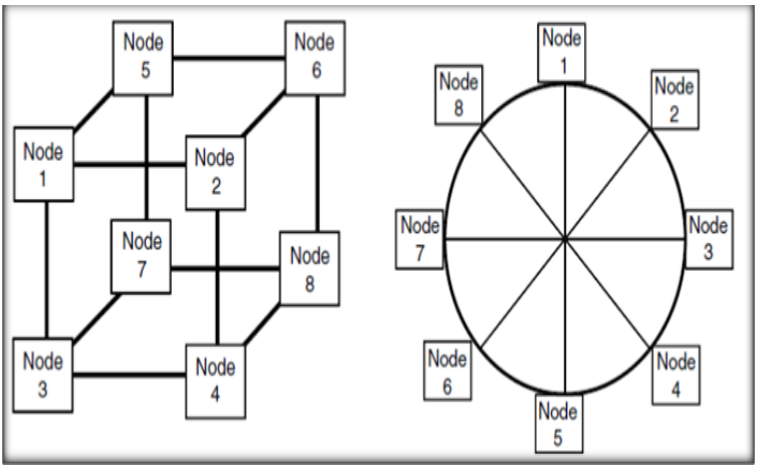
N -> set of the routers

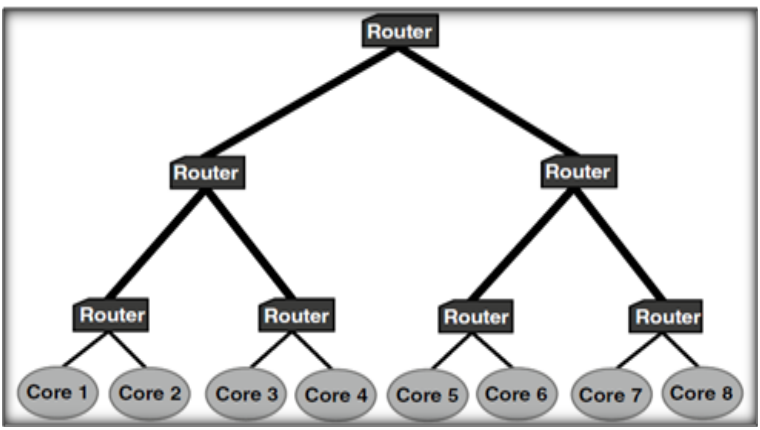
C -> set of the communication channels

The routers can be connected in the direct topology and indirect topology. In the direct topology all the routers are associated with the processor and their combination can be used as a single processing element in the system referred as the nodes in the communication network. In the direct topology the nodes are directly connected to the neighboring nodes with fixed length and messages transferred among the nodes directly with the help of one or more immediate nodes. The communication is taking place according to the different routing algorithms among the routers and routers are directly involved in the communication throughout the topology.



(a) 2D mesh (b) 2D torus





(e) Fat tree

Figure 4.1 Topological Structures

Most of the structured are based on the orthogonal arrangement with in the routers and nodes are scattered in N- Dimension and data packets moves in a dimension at a particular time. Such types of arrangement are trade-off

between performance and cost and give the scalable and programmable architecture. Most of the popular topologies are N- Dimension mesh, torus and hypercube. The 2D mesh topology is configured in XY structured where x represents the row and y represents the column. It is assumes that all the link have the same length to imposes the regular structure used in physical design for simplification. It is also easy to predict that the space or area required for the designing of mesh topology. One more reason is also that the topology grows almost in the linear nature with the increment in the number of the nodes in XY direction. The advantage of the mesh topology over the other is due to its structure linearity and physical design. It has also some disadvantage as router used in the mesh topology can lead the consisted region in the NoC. Due to this problem design has to take lot of care that the design and application should be mapped to avoid the traffic and concession especially in the center region of the mesh.

Fig. 4.1(a) to (e) shows the direct and indirect topology. The fat tree is the indirect topology as depicted in fig 4.1(e). In indirect topology all the routers are not connected in direct manner as in mesh, torus or ring. The source and target nodes are connected in indirect way such as in multistage networks, crossbar switch and tree structures. The tree topology follows the parent- child relation to connect the source to target node.

4.2 Mesh Topology

In computer network, a crosspoint switchis also referred as crossbar switch, or a switching matrix that provides multiple inputs to multiple outputs in the form of matrix. 2D NoC router follows the cross point switch. The mesh NoC is the structured form if m x n size routers. The data is processed by the nodes. Each node is consist of its own processing element. One node can communicate to another with the help of router. The m x n size of the NoC presents that the nodes can are identified by X direction and Y direction with the size of

m x n. For an example, let us consider 4 x 4 mesh architecture as shown in fig. 4.2. To address 16 routers ($2^n = 16$, n = 4 bit) addressing is required in which 2 bit address in taken care on X axis and 2 bit address on Y axis. The selection of the routers is done based on the cross point addressing as listed in table 4.1. It is called the XY routing of the 2D mesh network in which X axis presents the row address and Y axis presents the column address. To understand the detailed behavior of the 2D mesh NoC, another example of 64 nodes ($2^n = 64$, then n = 6 bit) is also depicted in fig. 4.3 which is identified by 3 bit address in X axis direction as row address and 3 bit address in Y axis direction in column address. The node addressing for nodes is listed in table 4.3. In the mesh NoC it is considered that all the nodes are placed on equal distance and configured or arranged in regular structure.

Table 4.1 2D Mesh routing using XY

Source_address		**Destination**
X	Y	Router Selection
00	00	Acknowledgment to Router R0
00	01	Acknowledgment to Router R1
00	10	Acknowledgment to Router R2
00	11	Acknowledgment to Router R3
01	00	Acknowledgment to Router R4
01	01	Acknowledgment to Router R5
01	10	Acknowledgment to Router R6
01	11	Acknowledgment to Router R7
10	00	Acknowledgment to Router R8
10	01	Acknowledgment to Router R9
10	10	Acknowledgment to Router R10
10	11	Acknowledgment to Router R11
11	00	Acknowledgment to Router R12
11	01	Acknowledgment to Router R13
11	10	Acknowledgment to Router R14
11	11	Acknowledgment to Router R15

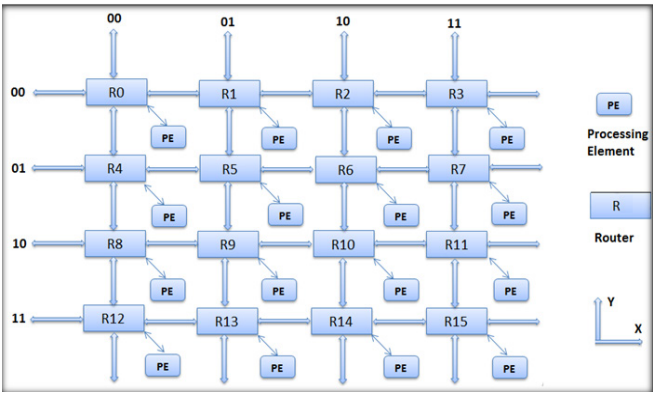


Figure 4.2 Mesh Configuration (4 x 4)

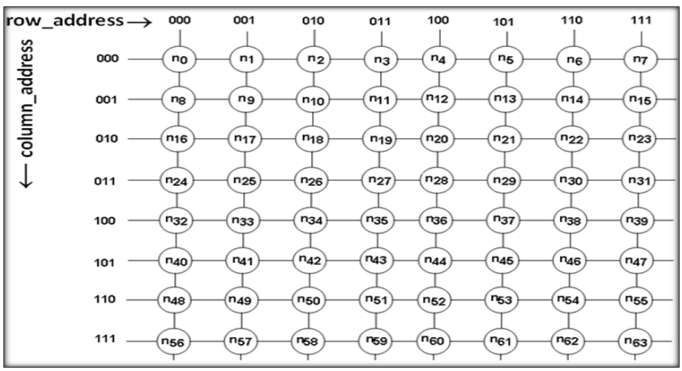


Figure 4.3 Mesh Configuration (8 x 8)

Table 4.2 Router identification in 2D_mesh network

X_ Address	Y_ address	Destination Router
000	000	Router 0
000	001	Router 1
000	010	Router 2
000	011	Router 3
000	100	Router 4
000	101	Router 5
000	110	Router 6
000	111	Router 7
001	000	Router 8
001	001	Router 9
001	010	Router 10
001	011	Router 11
001	100	Router 12
001	101	Router 13
001	110	Router 14
001	111	Router 15
010	000	Router 16
010	001	Router 17
010	010	Router 18
010	011	Router 19
010	100	Router 20
010	101	Router 21
010	110	Router 22
010	111	Router 23
011	000	Router 24
011	001	Router 25
011	010	Router 26
011	011	Router 27
011	100	Router 28
011	101	Router 29
011	110	Router 30
011	111	Router 31
100	000	Router 32
100	001	Router 33
100	010	Router 34
100	011	Router 35
100	100	Router 36
100	101	Router 37
100	110	Router 38
100	111	Router 39
101	000	Router 40

101	001	Router 41
101	010	Router 42
101	011	Router 43
101	100	Router 44
101	101	Router 45
101	110	Router 46
101	111	Router 47
110	000	Router 48
110	001	Router 49
110	010	Router 50
110	011	Router 51
110	100	Router 52
110	101	Router 53
110	110	Router 54
110	111	Router 55
111	000	Router 56
111	001	Router 57
111	010	Router 58
111	011	Router 59
111	100	Router 60
111	101	Router 61
111	110	Router 62
111	111	Router 63

In the general way, a matrix switch configured structure is formed with the help of a grid of crossing metal bars, and can be broadened in matrix form as crossbar switch. It is one of the original switching architectures, memory switch together with a rotating switchand a crossover switch Using crossbar switches up to 'N' CPU's can be connected. In case of same topological from, more CPU's can be connected and the network becomes too complicated and very expensive. Although the cluster size of n*k switches is needed for the crossbar switches connects 'n' no. of CPU's to 'k' memories of CPU's. We can understand the addressing and node selection scheme with the help of the functional table listed below. The selection of the source and destination subscriber depends on the X axis address and Y axis address. The value of XY is not fixed it is variable depending on the number of input and output nodes. For the considered case, it has been taken 3 bitas row wise address and 3 bit columns wise to recognizes

the node in 2D configuration.

The 3D mesh topology is also configured in the same manner as 2D mesh topology. In 2D mesh the routing is done based on XYZ. The diagram for Mesh (3 x 3 x 3) NoC in 3D is shown in [Figure 4.4]. In the design 27 routers can communicate. The packet information is depicted in [Figure 4.5]

End bit (1 bit): The status of the end bit is depicted about the ending of the transmission and it indicates that the data is received at the receiving end.

Layer Identification (3 bit): Multilayer environment is also supported by the 2D and 3D NoC. In layer identification, the addresses of the layers are identified. In our case it is assumed 3 bit, it means it can support 8 layers environment.

Sxyz (3, 3, 3) Source Router (9 bit): It denotes the address of source routers need to communicate based on XYZ routing. It is address as X (3 bit), Y (3 bit) and Z (3 bit) direction.

Dxyz (3, 3, 3) Destination Router (9 bit): It denotes the address of target routers to end the commutation as destination routers based on XYZ routing. It is addressed as X (3 bit), Y (3 bit) and Z (3 bit) direction.

Data (n bit): It indicates the size of the data. It may be of 'n' bit in our case it assumed of 0 to 255 or 256 bit data.

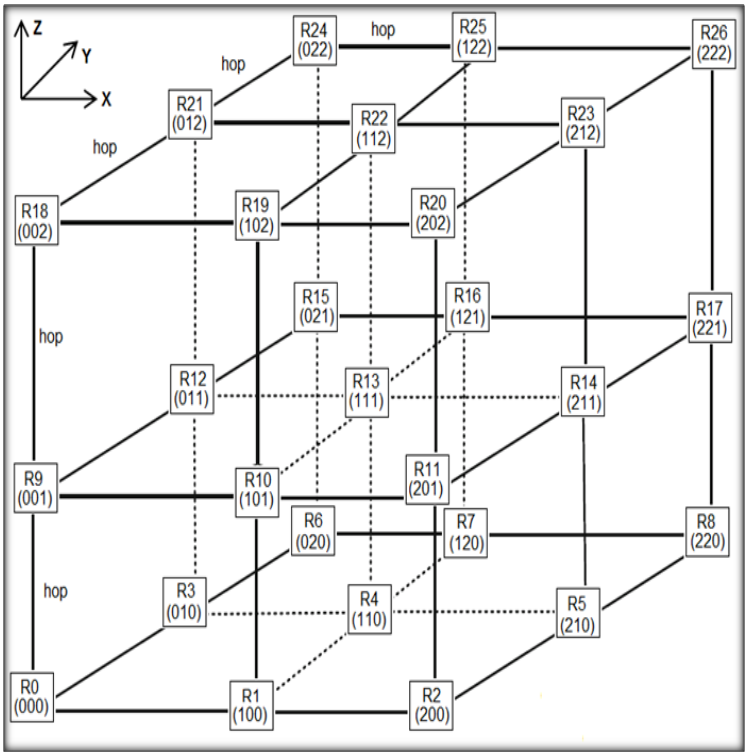


Figure 4.4 Mesh (3 x 3 x 3) NoC in 3D

End bit	Identification _ Layer	Router Source Sxyz (3, 3, 3)	Router_ Destination Dxyz (3, 3, 3)	Data (n bit)
(1 bit)	(2 bit)	(9 bit)	(9 bit)	(256 bit)

Figure 4.5 Data packet format

Table 4.3 Mesh 3D (3 x 3 x 3) NoC under XYZ routing

X_address	Y_address	Z_address	Router Selection	XYZ Routing
000	000	000	Acknowledgment to Router R0	(000)
001	000	000	Acknowledgment to Router R1	(100)
010	000	000	Acknowledgment to Router R2	(200)
000	001	000	Acknowledgment to Router R3	(010)
001	001	000	Acknowledgment to Router R4	(110)
010	001	000	Acknowledgment to Router R5	(210)
000	010	000	Acknowledgment to Router R6	(020)
001	010	000	Acknowledgment to Router R7	(120)
010	010	000	Acknowledgment to Router R8	(220)
000	000	001	Acknowledgment to Router R9	(001)
001	000	001	Acknowledgment to Router R10	(101)
010	000	001	Acknowledgment to Router R11	(201)
000	001	001	Acknowledgment to Router R12	(011)
001	001	001	Acknowledgment to Router R13	(111)
010	001	001	Acknowledgment to Router R14	(211)
000	010	001	Acknowledgment to Router R15	(021)
001	010	001	Acknowledgment to Router R16	(121)
010	010	001	Acknowledgment to Router R17	(221)
000	000	010	Acknowledgment to Router R18	(002)
001	000	010	Acknowledgment to Router R18	(102)
010	000	010	Acknowledgment to Router R20	(202)
000	001	010	Acknowledgment to Router R21	(012)
001	001	010	Acknowledgment to Router R22	(112)
010	001	010	Acknowledgment to Router R23	(212)
000	010	010	Acknowledgment to Router R24	(022)
001	010	010	Acknowledgment to Router R25	(122)
010	010	010	Acknowledgment to Router R26	(222)

4.3 Torus Topology

The torus topology is formed by the revolution of a circle on axis to the circle as a coplanar. In 2D planner topology the degree of freedom is 4and the nodes can the configured in 2D rectangular lattice shape with m x n size means m is the number of rows and n is the number of columns with four neighbours as connecting nodes. The rectangular array can be rolled out and the opposite side nodes can be visualized by rolling the nodes array structure. The 2D torus (4 x 4) is shown in fig. 4.6. The nodes can communicate in +X, +Y, -X and –Y direction. In case of the 3D torus nodes the network is configured in a shape that can shape a rectangular prism with 6 degree of freedom or directions +X, +Y,+Z –X,-Y and –Z directions. The example of 3D torus is depicted in fig. 4.7. The addressing of 2D torus and 3D torus is based on XY and XYZ routing discussed in mesh topology.

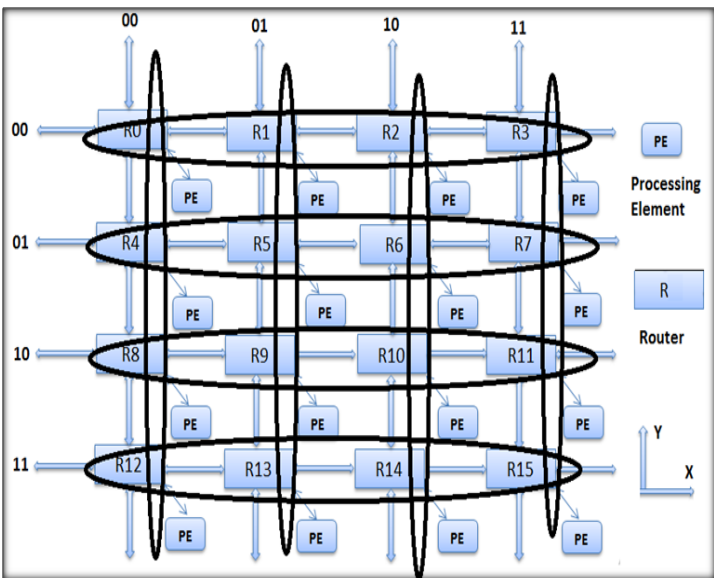


Figure 4.6 Torus (4 x 4) Topology in 2D shape

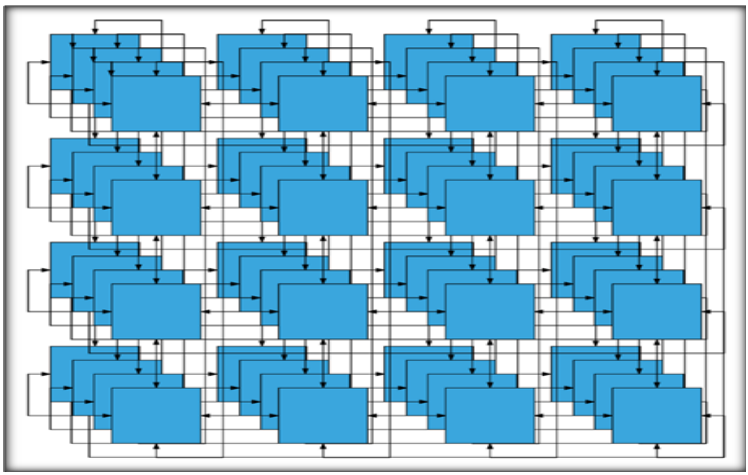


Figure 4.7 Torus (4 x 4 x 4) Topology in 3D shape

4.4 Ring Topology

Ring Topology is the well-known topology based on direct connections. The example of the ring NoC is Octagon which is the simple structure in which 8 nodes communicate to each other with the help of 12 interconnecting links. The Octagon is shown in fig. 4.8. The links are helping in the two ways communication of the structured NoC arranges in a ring shape. It is following the easy and simple algorithm to choose shortest path of routing. A switch is used to connect the nodes and establishes the communication in multidimensional shape. Fig. 4.9 shows the examples of 64 nodes are shaped in the ring form. To address 64 nodes in ring form the addressing of 6 bit is required. It is started from "000000" as M0 for node 0 and ended to "111111" M63 as node 63. Table 4.4 lists the addressing and behavior of router selection in the ring topological structure form node 0 to node 63 and corresponding routers as R0 to R63. All the nodes in the ring can communicate to each other. The node data packet has the data format in which source node and destination nodes address are kept of 6 bit each and 256 bit is the size of data.

The inter-process communication is done by the ring based NoC and corresponding architecture. Nodes are understood with the help of source address and destination address. For an example let node 1 want to communicate with node 15 then source address will be "000001" and destination node address will be "00001111". If a node wants to communicate with any one, it has the probability to communicate any of the target nodes as shown in fig. 4.10

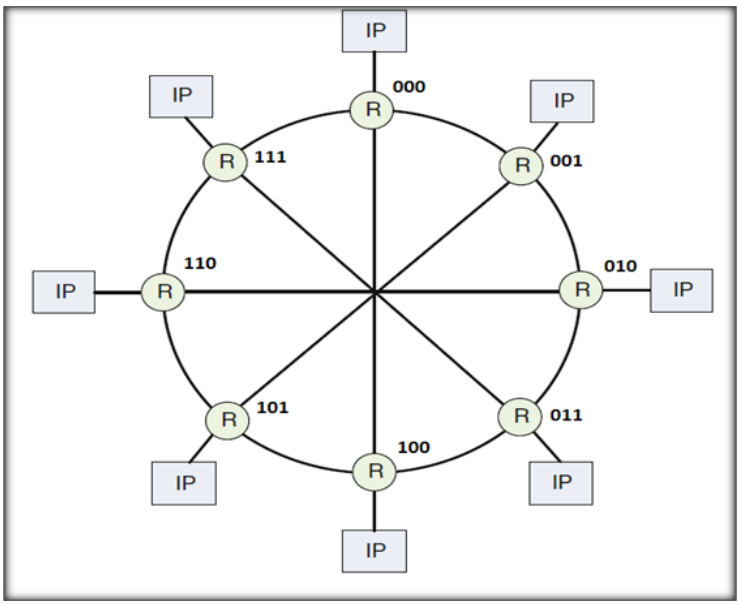


Figure 4.8 Ring NoC structured by 8 nodes

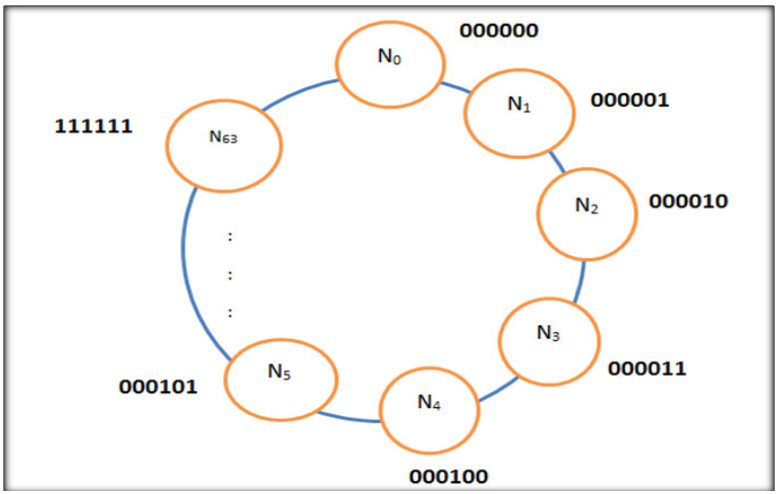


Figure 4.9 Ring NoC structured by 64 nodes

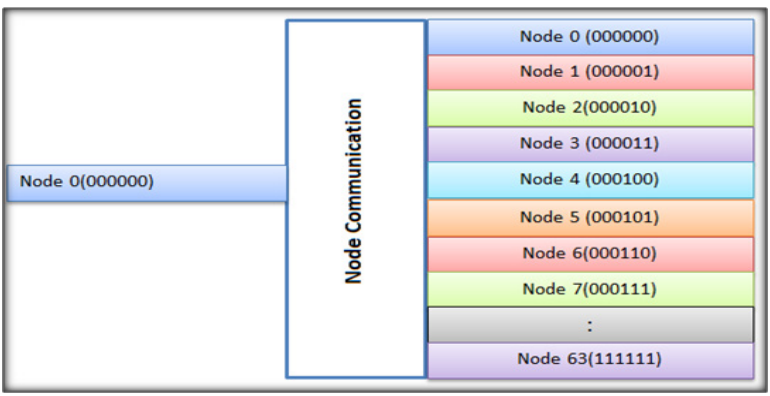


Figure 4.10 One node "0000" intercommunication to other nodes

Table 4.4 Node Identification scheme in ring NoC

Source_node(6 bit)	Destination_ node(6 bit)	Router Selection
000000 (Router 0)	000000:111111	Router 0 :Router 63
000001 (Router 1)	000000:111111	Router 0:Router 63
000010 (Router 2)	000000:111111	Router 0:Router 63
000011 (Router 3)	000000:111111	Router 0:Router 63
000100 (Router 4)	000000:111111	Router 0:Router 63
000101 (Router 5)	000000:111111	Router 0:Router 63
000110 (Router 6)	000000:111111	Router 0:Router 63
000111 (Router 7)	000000:111111	Router 0:Router 63
001000 (Router 8)	000000:111111	Router 0:Router 63
001001 (Router 9)	000000:111111	Router 0:Router 63
001010 (Router 10)	000000:111111	Router 0:Router 63
:	:	:
111111 (Router 63)	000000 :111111	Router 0:Router 63

The packet data is transmitted form the source router to target router. Fig. 4.11 shows the packet information having 6 bit defined for source router and 6 bit defined for target router. When the multiple requests are arriving to one of the destination, the priority based on FIFO logic is given to set the target nodes. The data of 'n' bit transfer is possible in ring NoC but in our case it is considered of 256 bit.

Source Router Address (6 bit)	Destination Router Address (6 bit)	Data value (n bit) (256 bit)

Figure 4.11Communication data format

4.5 Fat Tree Topology

Fat trees are the indirect topology network by Charles E. Lesierson suggested in 1985 in which the routers are connected in top to bottom in the form of tree. The tree structure has the advancement in its design that the links form the top to bottom and bottom to top are equal. The structure is based on the parent (root) and child, sub Childs relationship as siblings. It is the reason that the links are becoming "fatter" towards the top side of the tree. The switch in the root direction is having much number of links in comparison to the switch in the bottom side. The fat tree structure having 16 nodes in bottom is shown in fig 4.12.In the stage first, the router has two branches '1' and '0'. Two subunits (00, 01) are attached to the branch '0' and two sub units (10, 11)to branch '1'. In this four routers in stage two are addressed as"00", "01", "10" and "11".

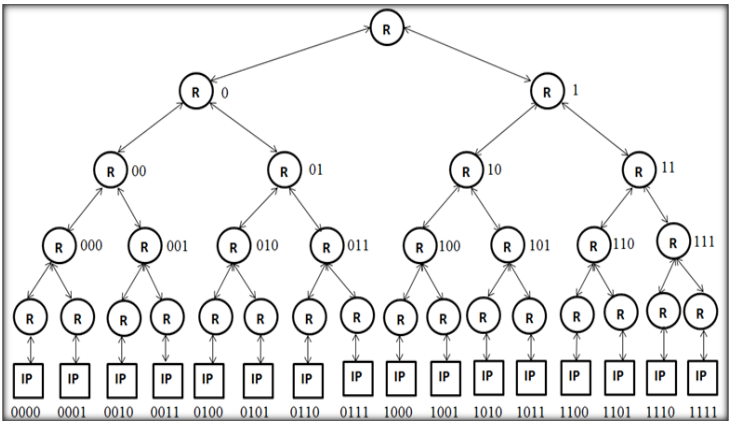


Figure 4.12 Structured fat tree NoC for 16 nodes

In the same way each router associated with third stage routers which are "000", "001", "010", "011", "100", "101", "110" and "111". In the last stage the sub routers are associated with their IPs. The communication for the 16 nodes form parent to child nodes or routers is possible in indirect way in the same tree the nodes are identified as "0000" node0, "0001" node1, "0010" node2 and so on.

Methodology and Implementation

The chapter explains the research methodology and software tools description from simulation as well as synthesis. It also details the complete environment of functional simulation and logic verification

5.1 Software Tools

The software used in the design of NoC and the chip implementation used is Project Navigator Xilinx ISE 14.2 and Modelsim 10.0 software

5.1.1 Xilinx ISE Project Navigator 14.2

Xilinx is one of the leading companies in the field of SCI and FPGA design. It is the biggest semiconductor company to cover the front end solutions in the chip design, verification and synthesis. In the Xilinx software the programmers are developing the chip using latest HDL languages such as Verilog HDL, VHDL, and ABEL etc. After the design there are the options to see the RTL, inter schematic view and view synthesis report. The developed chip is configured using input pins, output pins and input/output pins. Xilinx has the ISIM simulator to see the waveform using inbuilt waveform simulator which provides the functional check of the developed chip. It also has the Chip scope for FPGA signal analysis, Static timing analysis feature, verification and logical synthesis environment. Different test benches and test cases are simulated in the software environment and FPGA guarantees the chip for mask production in the market. The tool provides the full information of logic design, synthesis, simulation, and verificationand timing analysis. The hardware and pre-synthesis parameters obtained directly from the tool which details the hardware parameters usage, memory requirements and timing values required in the design of chip.

5.1.2 ModelsimSoftware 10.1 Version

Modelsim Software is the software given by the Mentor Graphics Company. It is a multi-language HDL simulation software works on Verilog HDL, VHDL and System 'C'. It has the inbuilt 'C' debugger. It is preferred one of the best tool for GUI and Xilinx software interface. The chip design, functional simulation and timing analysis are done using the software. It also can be integrated with MATLAB or Simulink environment. The software gives the following advantages

Benefits of Modelsim EE

- Modelsim software gives low cost chip design solution using HDL
- Providing interactive debug using Intuitive GUI in effective time
- It simplifies the research data and manages the project management integrated in software and hardware
- It has outstanding technical support to give the solutions in HDL and easy to use
- Easy to use with outstanding technical support
- Popular ASIC libraries are available and sign-off support for all defined libraries
- The complete platform for hardware and software debugging
- Simplifies the functional simulation and gives testing environment for all the possible test cases, can be used to check the functionality of the designed chip.

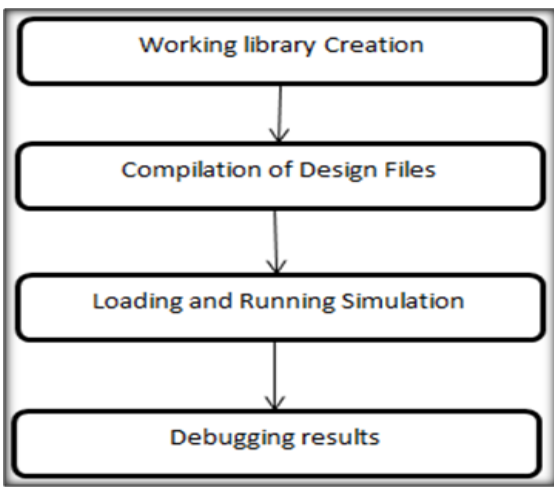


Figure 5.1 Modelsim design process

The chip design flow and simulation block diagram using Modelsim software is shown in fig. **5.1**

- Creation of working library:All the chip design based solutions in Modelsim software require the creation of the library. There is the default library in the Modelsim by the name 'work' which includes all the logical values and possible library functions required to do simulation by the compiler in the working library. The default destination of all the chip design is the library in Modelsim software.

- Designed file compilation:The designed which is developed using any HDL, and stored in the working library is complied. The binary created by the user is suited to work on all the platforms. The designed file may be one file or grouped as the top design having sub modules or structured in bottom to top level.

- Running and loading Simulation: The designer is loading the top module of the developed chip into the simulator after completing the compilation of the design. The loading involves the entity of chip and architecture. The design can be done in dataflow, behavior and structural modeling. The modeling of the design is choosen by the designer. Some design structured using structure style of modeling, gives better results. The design is developed in different modeling can give different timing and performance results. In the running process, it is considered that the simulation period is zero and run operations are entered by the designer to perform the functional simulation

- Results debugging:The developed chip may contain some errors. So, debugging environment is required to track the errors in program window. The errors are listed by the modelsim with respect to line and code debugging environment help the program to check the code and take correct action in design. There are redefined scripts can be used in the software to follow the short methods.

The verified results are seen in the form of waveform and timing diagrams.

5.2 Methodology

Multiple steps are required in the NoC design.The methodology has the several steps used to design the chips for topological architectures.

- Design Specification: There are two approaches in the NoC design one is bottom up design another is top-down approach. In the bottom up technique the design is developed for small modules and structured in a top design using structural style of modeling. In top –bottom up approach the full system is considered and designed in a way that it will meet the behavior of the system.

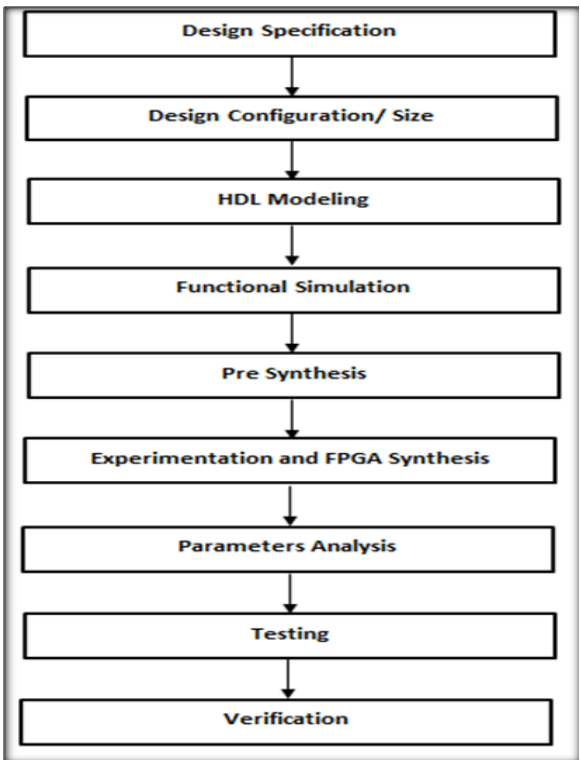


Figure 5.2 NoC design methodology

- Design Configuration:In the NoC design of ring, torus, mesh and fat tree topology is of the cluster size. The topology design can vary with the cluster size and topology behavior in 2D and 3D. The design specifications are decided by the designer based on company/ project requirement.

- HDL Modelling:The chip design is done using any of the HDL language. The industry favorite languages are Verilog HDL and VHDL. In our design of 2D and 3D topological NoC, VHDL based design is used for the problem statement of our research because the NoC design can be modeled in dataflow modeling, behaviour modeling or structured modeling using this language.

- Functional Simulation: The functional simulation depends on reset circuitry, clock input and test cases. The designed chips and modules are checked by RTL view, internal schematic diagram and test cases. Test cases are decided by the designer based on the functionality of the designed chip.

- Pre Synthesis:In the parenthesis the NoC performance is analyzed based on the hardware and timing parameters. The hardware parameters are slices, flip-flops, LUTs and memory requirements in the chip. The timing parameters are relating to combination delay, minimum time and maximum time of clock etc. The timing parameters are relating to combination delay, minimum time and maximum time of clock etc. If any chip design is consuming the memory more than 100% utilization, then chip redesigning is required.

- FPGA Synthesis and Experimentation:Based on the view synthesis report as the hardware and device utilization summary the designed modules are synthesized on FPGA. The FPGA is interfaced with the computer in which the code was developed for specific NoC. The test inputs are given using switches of the board and verified

using LED, LCD or monitor using VGA. FPGA has the feature of inbuilt ADC and DAC conversion. So, FPGA synthesis and experimentation is required to test the designed chip in real time application. In the NoC design experimentation is done on Vertex 5, high speed FPGA.

- Parameters Analysis:The NoC design parameters analysis is must and required to check step by step weather the hardware and other related parameters are consuming unbalanced number of resources. In NoC design the parameters analysis is done based on the cluster size and configuration.

- Testing:FPGA accepts the inputs in the form of LUTs. The synthesized results are checked and tested successfully on FPGA. In the NoC design, main testing is done for the input signal processed by the FPGA or not. FPGA input and output signal can be checked with the help of Chip scope inbuilt in the Xilinx ISE Software.The signals can be checked on CRO or DSO because the FPGA can be interfaced with DSO because it has inbuilt ADC and DAC that provides this feasible solution. The testing is also done in software and data is checked in the input signals and processed properly. The assignment of the source router address, destination router addresses and data size are the essential part in each NoC.

- Verification: Standard VHDL Programming has all the features required to check and verify the stimuli, randomization and functional coverage. It is essential in NoC because it directly relating to clear cluster size of the network. The nodes can discontinue their link, routing. There should be alternate route, path, and links for full available NoC communication. The data communication the destination nodes is checked by 2D and 3D mesh, ring, torus and fat tree topology. The verification of NoC is done using timing parameters and different possible test cases of the developed chip called design under test (DUT).

RESULTS

The chapter explains the results relating to the design and simulation in Xilinx software environment and functional verification on Modelsim 10.0 software. The results are presented for 2D NoC router; 3D NoC router, 2D and 3D mesh NoC, 2D and 3D Torus NoC, and 2D and 3D ring NoC.

6.1 RTL and Waveform Simulation for 2D NoC Router

The RTL shows the all inputs and outputs of the router. It is the description of all the input and output logic used to design the chip. The RTL view of the developed 2D router chip is shown in fig. 6.1 and its internal schematic is depicted in fig.6.2. Internal schematic presents the internal structure having different logic gates and internal hardware required in the design. The functional verification is carried for 256 bit data transfer and simulation waveforms are shown in fig. 6.3 and fig. 6.4 Table 6.1 is used to discuss the use, size and detail of the pins used.

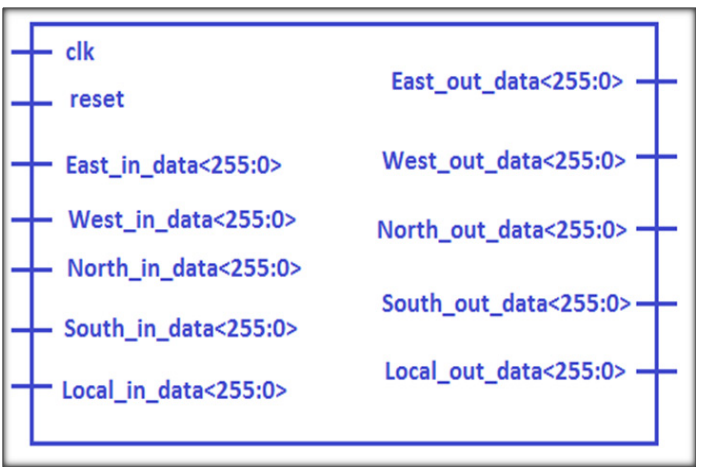


Figure 6.1 RTL of Router in 2D

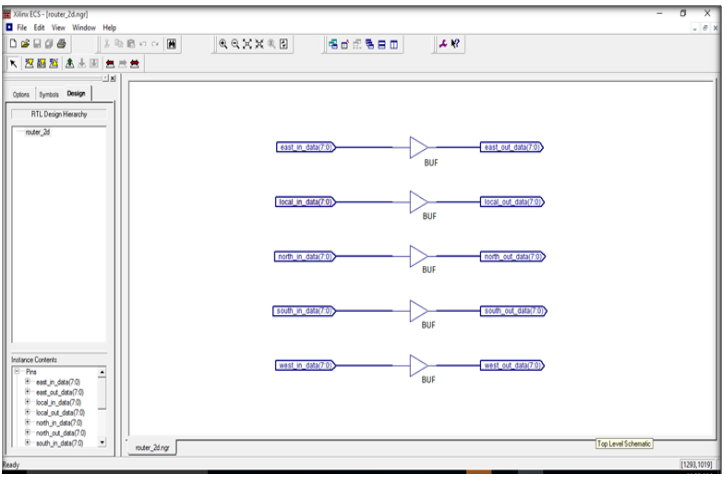


Figure 6.2 Internal schematic of the 2D router

Table 6.1 Pin explanation for the 2D router chip

Pin	**Size**	**Functional Description**
Reset	1 bit	It is the input pin of std_logic, used to reset all the data and contents of all the ports and registers in the design.

Clk	1 bit	It is 1 bit input and default associated with design used to give the rising edge (+ve) edge in the simulation results and works on 50 % duty cycle.
East_in_data<255:0>	256 bit	It is the input data packet coming in the router from east input port (256 bit) and controlled through the control unit associated with network.
West_in_data<255:0>	256 bit	It is the input data packet coming in the router from west input port (256 bit) and controlled through the control unit associated with network.

North_in_data<255:0>	256 bit	It is the input data packet coming in the router from north input port (256 bit) and controlled through the control unit associated with network.

South_in_data<255:0>	256 bit	It is the input data packet coming in the router from south input port (256 bit) and controlled through the control unit associated with network.
Local_in_data<255:0>	256 bit	It is the input data packet coming in the router from local input port (256 bit) and controlled through the control unit associated with network.
East_out_data<255:0>	256 bit	It is the output port having 256 bit data on east out data and stored in the corresponding register.

West_out_data<255:0>	256 bit	It is the output port having 256 bit data on west out data and stored in the corresponding register.
North_out_data<255:0>	256 bit	It is the output port having 256 bit data on north out data and stored in the corresponding register.
South_out_data<255:0>	256 bit	It is the output port having 256 bit data on south out data and stored in the corresponding register.
Local_out_data<255:0>	256 bit	It is the output port having 256 bit data on local out data and stored in the corresponding register.

The simulation is verified in Modelsim 10.0 software. Fig. 6.3 and Fig. 6.4 present Modelsim waveform result for 2D router for 256 bit data transfer in hexadecimal and ASCII. The data is coming from the port east_in_data<255:0> and local_in_ data<255:0> and output is obtained at east_out_data<255:0> and local_out_data<255:0>.

Test-1: Reset = '1' and run the simulation environment. All the ports will get zero value. Reset = '0' and give the rising edge of clk, and East_in_data<255:0> ,Local_in_data<255:0> = "54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72" (heaxdecimal) or "TMU@ TMU@ComputerTMU@TMU@Computer" in ASCII. The

same data is obtained at East_out_data and local_out_data. The data is given as "0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010"

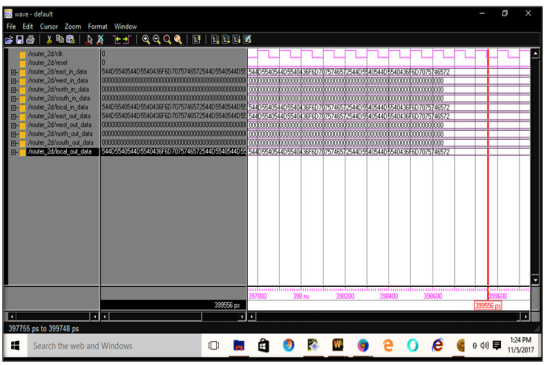


Figure 6.3 Modelsim waveform result for 2D router data transfer, 256 bit (hexadecimal)

6.2 is used to discuss the use, size and

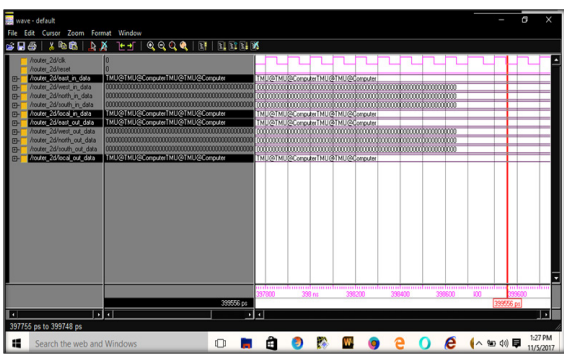


Figure 6.4 Modelsim waveform result for 2D router data transfer, 256 bit (ASCII)

6.2 RTL and Waveform Simulation for 3D NoC Router

The RTL view of the developed 3D router chip is shown in fig. 6.5 and its internal schematic is depicted in fig.6.6. Internal schematic presents the internal structure having different logic gates and internal hardware required in the design. Table detail of the pins used. The functional verification is carried for 256 bit data transfer and simulation waveforms are shown in fig. 6.7 and fig. 6.8. Modelsim waveform in Fig. 6.7 and Fig. 6.8 presents the result for 3D router for 256 bit data transfer in hexadecimal and ASCII. The data is coming from the port local_in_data<255:0>, Up_in_data<255:0> and down_in_ data<255:0> and output is obtained at local_out_data<255:0>, Up_out_data<255:0> and down_out_data<255:0>. The flow chart of the 2D and 3D router is shown in fig. 6.9.

Test-1: Reset = '1' and run the simulation environment. All the ports will get zero value. Reset = '0' and, given+ the rising edge of clk, and East_in_data<255:0>, Local_in_data<255:0> = "54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72" (heaxdecimal) or "TMU@ TMU@ComputerTMU@TMU@Computer" in ASCII. The same data is obtained at East_out_data and local_out_data. The data is given as "0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010"

Table 6.2 Pin explanation for the 3D router chip

Pin	Size	Functional Description
Reset	1 bit	It is the input pin of std_logic, used to reset all the data and contents of all the ports and registers in the design.

Clk	1 bit	It is 1 bit input and default associated with design used to give the rinsing edge (+ve) edge in the simulation results and works on 50 % duty cycle.
East_in_data<255:0>	256 bit	It is the input data packet coming in the router from east input port (256 bit) and controlled through the control unit associated with network.
West_in_data<255:0>	256 bit	It is the input data packet coming in the router from west input port (256 bit) and controlled through the control unit associated with network.
North_in_data<255:0>	256 bit	It is the input data packet coming in the router from north input port (256 bit) and controlled through the control unit associated with network.
South_in_data<255:0>	256 bit	It is the input data packet coming in the router from south input port (256 bit) and controlled through the control unit associated with network.
Local_in_data<255:0>	256 bit	It is the input data packet coming in the router from local input port (256 bit) and controlled through the control unit associated with network.

Up_in_data<255:0>	256 bit	It is the input data packet coming in the router from upward input port (256 bit) and controlled through the control unit associated with network.
Down_in_data<255:0>	256 bit	It is the input data packet coming in the router from downward input port (256 bit) and controlled through the control unit associated with network.
East_out_data<255:0>	256 bit	It is the output port having 256 bit data on east out data and stored in the corresponding register.
West_out_data<255:0>	256 bit	It is the output port having 256 bit data on west out data and stored in the corresponding register.
North_out_data<255:0>	256 bit	It is the output port having 256 bit data on north out data and stored in the corresponding register.
South_out_data<255:0>	256 bit	It is the output port having 256 bit data on south out data and stored in the corresponding register.
Local_out_data<255:0>	256 bit	It is the output port having 256 bit data on local out data and stored in the corresponding register.

Up_out_data<255:0>	256 bit	It is the output port having 256 bit data on upward out data and stored in the corresponding register.
Down_out_data<255:0>	256 bit	It is the output port having 256 bit data on downward out data and stored in the corresponding register.

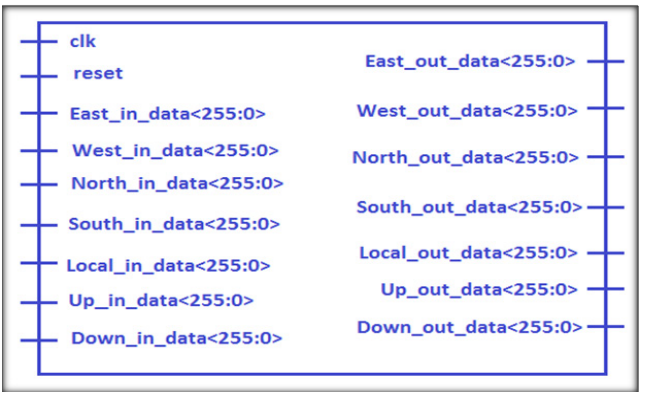


Figure 6.5 RTL of Router in 3D

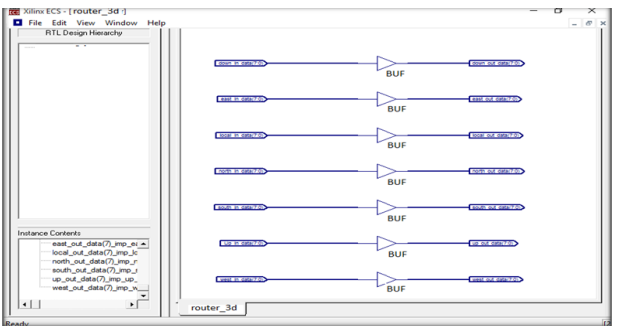


Figure 6.6 Internal schematic of the 3D router

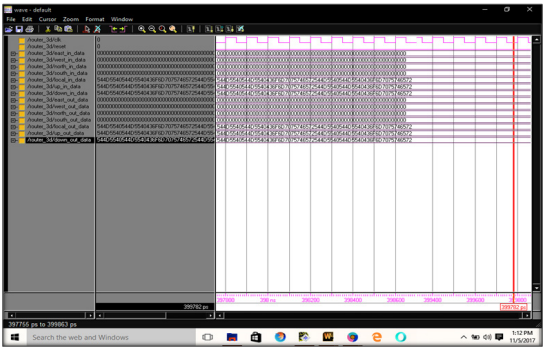


Figure 6.7 Modelsim waveform result for 3D router data transfer, 256 bit (Hexadecimal)

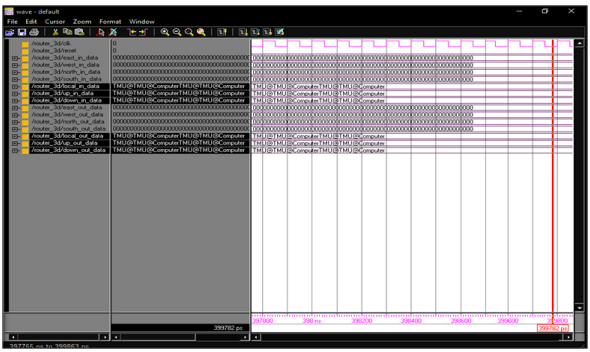


Figure 6.8 Modelsim waveform result for 3D router data transfer, 256 bit (ASCII)

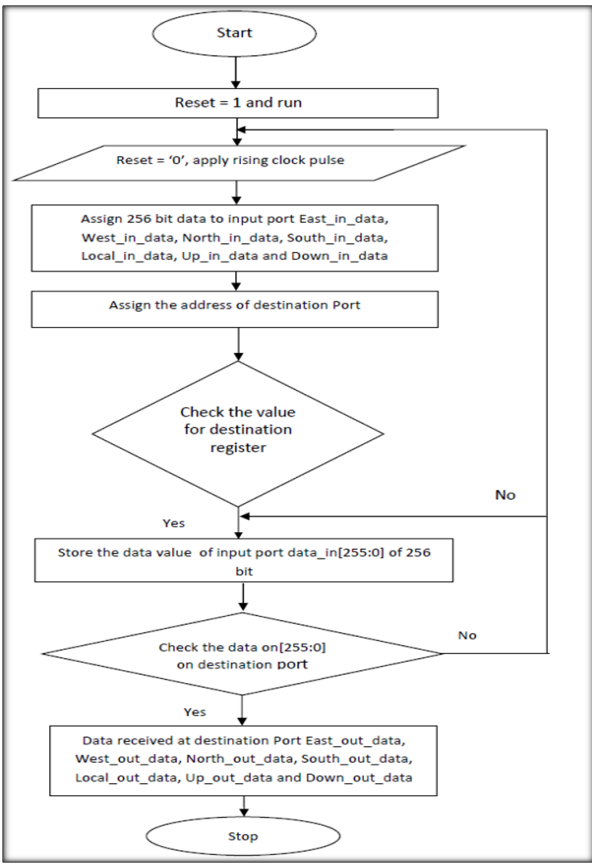


Figure 6.9 Flow Chart for 2D and 3D NoC Router

6.3 RTL and Wave form Simulation for 2D Mesh, Torus, Ring and Fat tree Topology

The RTL view of the developed 2D mesh, torus, ring and fat tree topology is shown in fig. 6.10 and its internal schematic is depicted in fig.6.11. Internal schematic presents the internal structure having different logic gates and internal hardware required in the design. Table 6.3 is used topology. The functional verification is carried for 256 bit data transfer and

simulation waveforms. Fig. 6.12 (a) shows the waveform and simulation off mesh NoC (4 x 4) for 8 bit data communication, Fig. 6.12 (b) 16 bit data communication, Fig. 6.12 (c) 32 bit data communication, Fig. 6.12 (d) 64 bit for 8 bit data communication, Fig. 6.12 (e) 128 bit data communication binary Fig. 6.12 (f) 128 bit data communication in hexadecimal. Fig. 6.13 and Fig. 6.14 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for mesh topology. Fig. 6.15 and Fig. 6.16 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for torus topology. Fig. 6.17 and Fig. 6.18 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for ring topology. Fig. 6.19 and Fig. 6.20 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for fat tree topology. The flow chart for the functional behavior is shown in fig. 6.21.to discuss the use, size and detail of the pins used for 2D mesh, torus, ring and fat tree

Table 6.3 Pin explanation for the 2D topology (Mesh, Torus, Ring and Fat tree)

Pin	**Size**	**Functional Description**
Reset	1 bit	It is the input pin of std_logic, used to reset all the data and contents of all the ports and registers in the design.

Clk	1 bit	It is 1 bit input and default associated with design used to give the rinsing edge (+ve) edge in the simulation results and works on 50 % duty cycle.
Data_in<255:0>	256 bit	It is the input data packet (256 bit) by the source router and controlled through the control unit associated with network.
Data_out<255:0>	256 bit	It is the output data packet (256 bit) by the source router and controlled through the control unit associated with network.
In_node_address<3:0>	4 bit	It is the input of source node used to define the input address of the source node need to communicate
Out_node_address<3:0>	4 bit	It is the input for destination node used to define the output address of the target node need to communicate

Row_adddres	2 bit	It is the address of the source and target nodes corresponding to the row for mesh and torus topology

Column_address	2 bit	It is the address of the source and target nodes corresponding to the column for mesh and torus topology
Write_enable	1 bit	It is the input control signal used to write the contents from source node to destination node. If the write_enable = '1' and read_enable = '0', the contents are written from the control unit to register
Read_enable	1 bit	It is the input control signal used to read the contents from source node to destination node. If the read_enable = '1' and write_enable = '0', the contents are written from the control unit to register
M0<255:0>	256 bit	It is the input/output node carrying 256 bit data
M1<255:0>	256 bit	It is the input/output node carrying 256 bit data
M2<255:0>	256 bit	It is the input/output node carrying 256 bit data

M3<255:0>	256 bit	It is the input/output node carrying 256 bit data
M4<255:0>	256 bit	It is the input/output node carrying 256 bit data
M5<255:0>	256 bit	It is the input/output node carrying 256 bit data
M6<255:0>	256 bit	It is the input/output node carrying 256 bit data
M7<255:0>	256 bit	It is the input/output node carrying 256 bit data
M9<255:0>	256 bit	It is the input/output node carrying 256 bit data
M10<255:0>	256 bit	It is the input/output node carrying 256 bit data
M11<255:0>	256 bit	It is the input/output node carrying 256 bit data
M12<255:0>	256 bit	It is the input/output node carrying 256 bit data
M13<255:0>	256 bit	It is the input/output node carrying 256 bit data
M14<255:0>	256 bit	It is the input/output node carrying 256 bit data
M15<255:0>	256 bit	It is the input/output node carrying 256 bit data

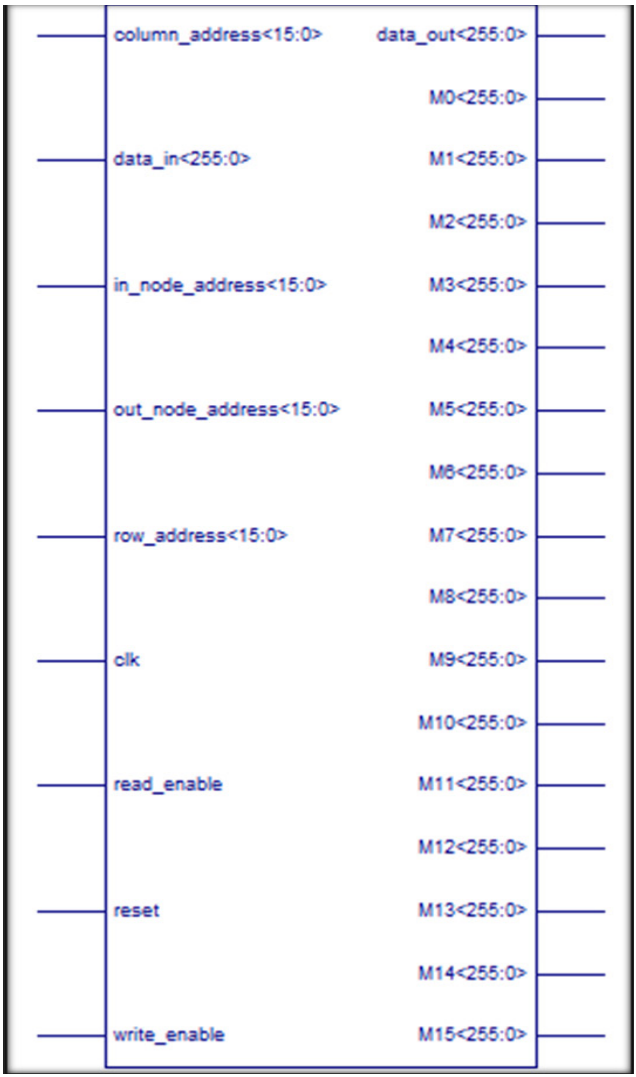


Fig. 6.10 RTL of 2D Mesh, Torus, Ring and Fat tree Topology

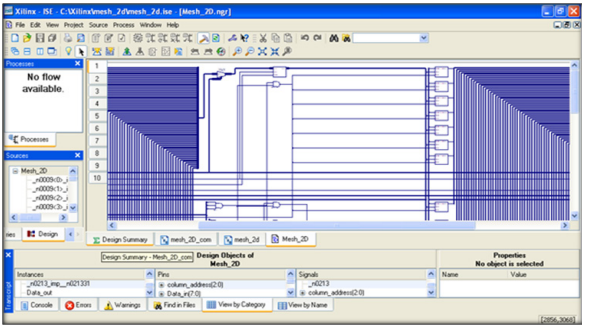
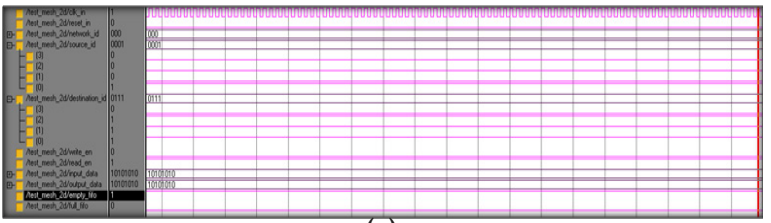
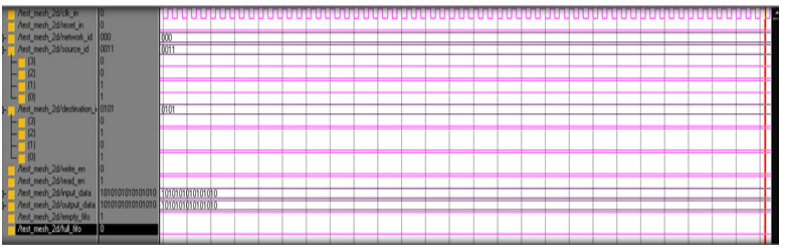


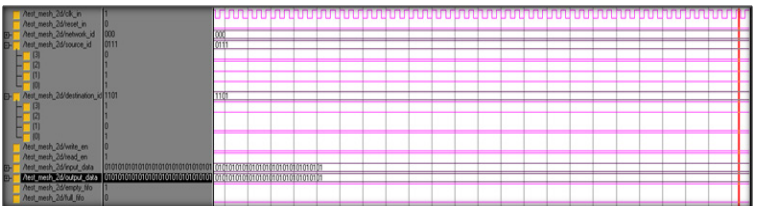
Figure 6.11 RTL of 2D Mesh, Torus, Ring and Fat tree Topology



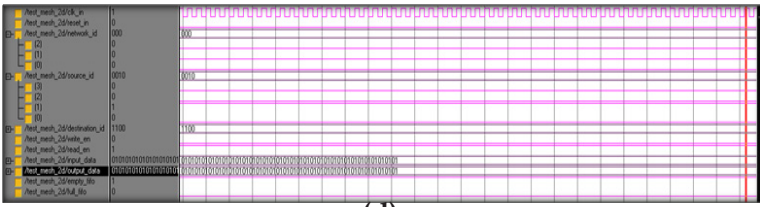
(a)



(b)

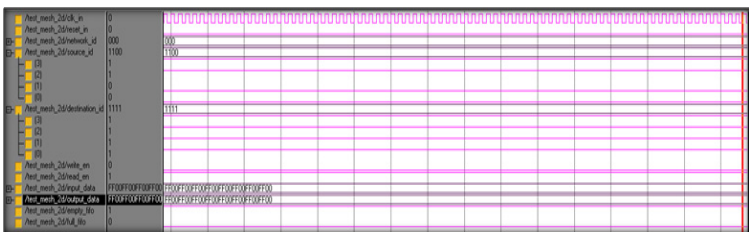


(c)



(d)

(e)



(f)

Figure 6.12 (a) Waveform and simulation of mesh NoC (4 x 4) for 8 bit data communication (b) 16 bit data communication c) 32 bit data communication (d) 64 bit data communication (e) 128 bit data communication (f) 128 bit data communication in hexadecimal.

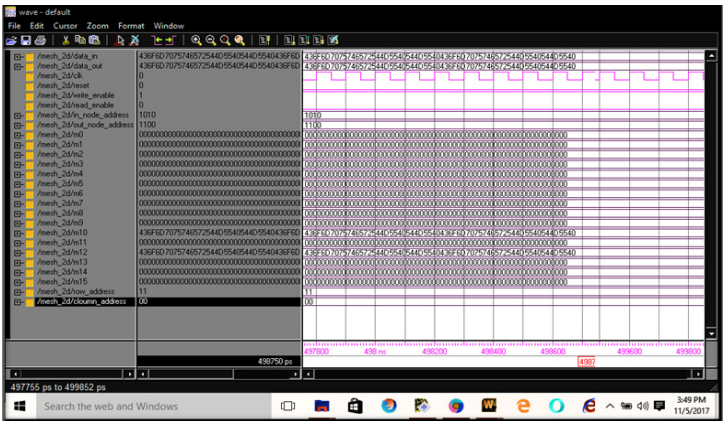


Figure6.13 Modelsim result simulation of 256 bit data in hexadecimal for 2D mesh topology

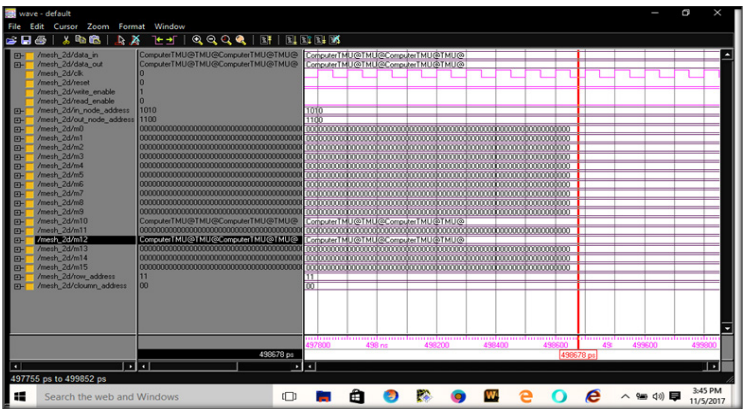


Figure 6.14 Modelsim result simulation of 256 bit data in ASCII for 2D mesh topology

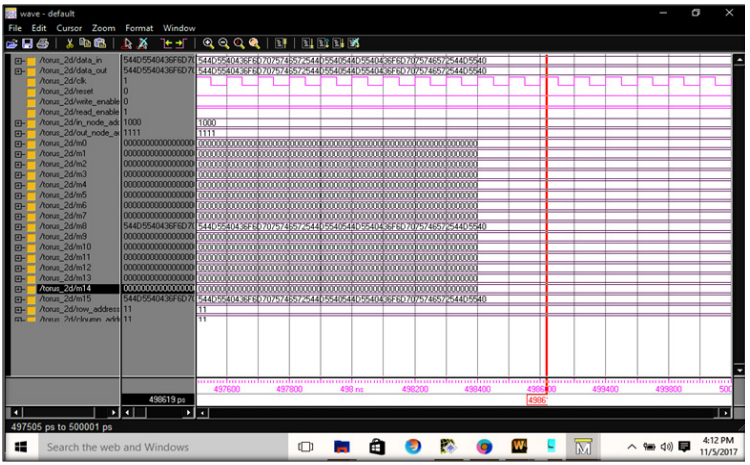


Figure 6.15 Modelsim result simulation of 256 bit data in hexadecimal for 2D torus topology

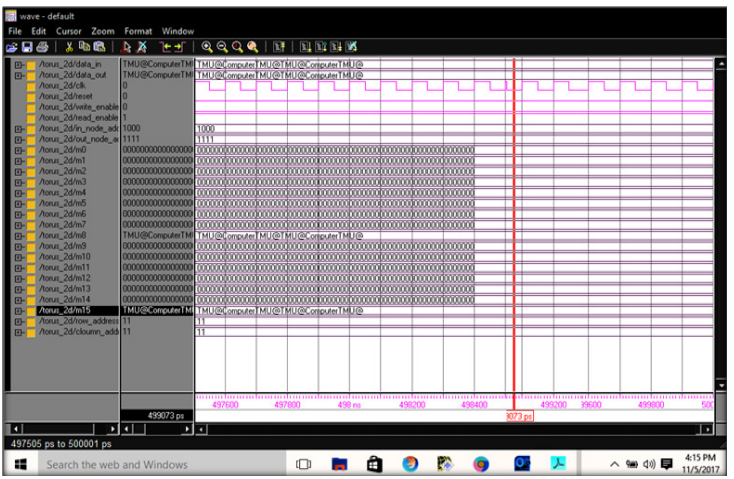


Figure 6.16 Modelsim result simulation of 256 bit data in ASCII for 2D torus topology

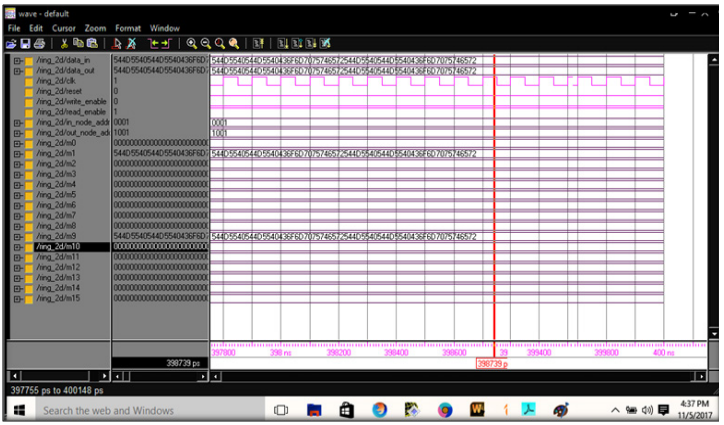


Figure 6.17 Modelsim result simulation of 256 bit data in hexadecimal for 2D ring topology

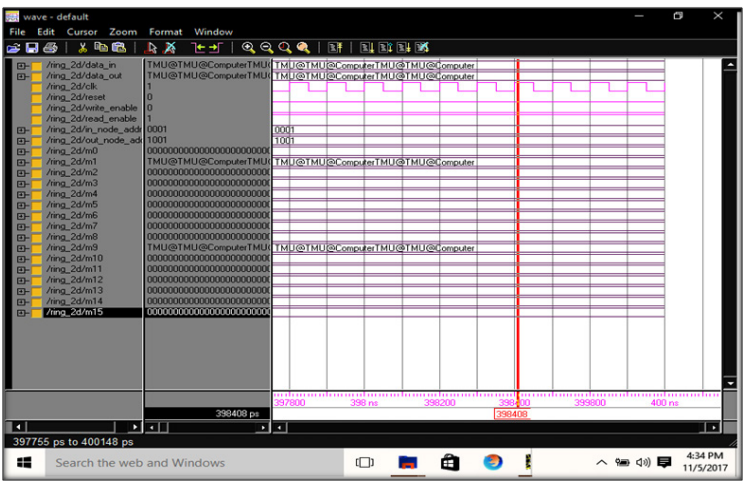


Figure 6.18 Modelsim result simulation of 256 bit data in ASCII for 2D ring topology

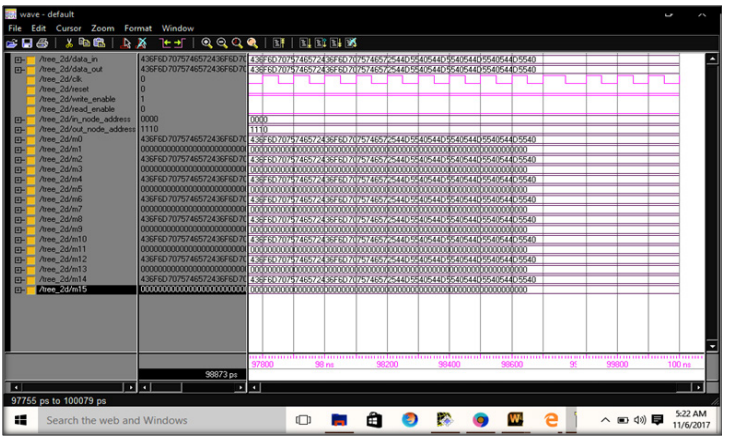


Figure 6.19 Modelsim result simulation of 256 bit data in hexadecimal for 2D fat tree topology

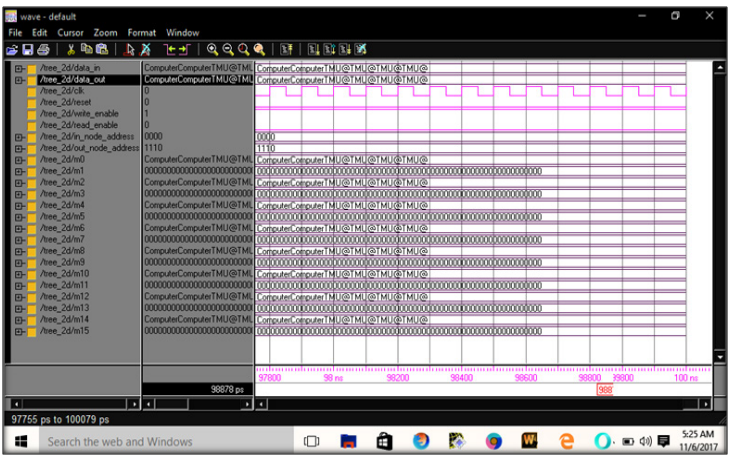


Figure 6.20 Modelsim result simulation of 256 bit data in ASCII for 2D fat free topology

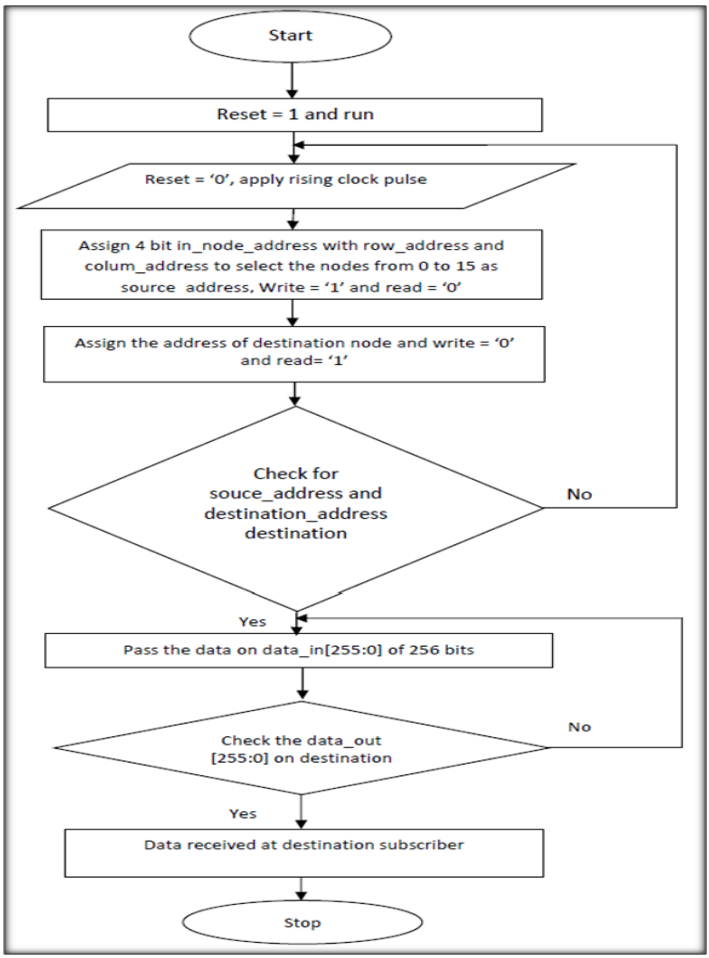


Figure 6.21 Flow chart for 2D Topology (Mesh, Torus, Ring and Fat Tree)

The Simulation of the 2D NoC is completed for the following test cases:

Test -1 (Mesh): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', Source_id = "0001", Destination_id = "0111", Read_in = '0',

input_data = "10101010", in hexadecimal (AA). The data is written in destination_id. Now, Write_en = '0', Read_in = '1', data is transferred to destination node.

Test -2 (Mesh): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_ node_address = "1010" out_node_address = "1100", row_ address = "11" and column_addess = "00" based on output node, data_in = "43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40" in hexadecimal or **ComputerTMU@TMU@ComputerTMU@ TMU@ in ASCII.** The same data is from source node M10 <255:0>. When Write_en = '0', Read_in = '1', the destination node M10 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -3 (Torus): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_ node_address = "1000" out_node_address = "1111", row_ address = "11" and column_addess = "11" based on output node, data_in = "54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40" in hexadecimal or **TMU@ComputerTMU@TMUComputerTMU@ in ASCII.** The same data is from source node M8 <255:0>. When Write_ en = '0', Read_in = '1', the destination node M15 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -4 (Ring): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_ node_address = "0001" out_node_address = "1001", based on output node, data_in = "54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72" in hexadecimal or **TMU@TMU@ComputerTMU@TMU@ Computer in ASCII.** The same data is from source node M1 <255:0>. When Write_en = '0', Read_in = '1', the destination node M9 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010" in binary.

Test -5 (Fat Tree): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_ node_address = "0000" out_node_address = "0000", "0010", "0100", "0110", "1000", "1010", "1100" and "1110", based on output node, data_in = "43 6F 6D 70 75 74 65 72 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 54 4D 55 40 54 4D 55 40" in hexadecimal or **ComputerComputer TMU@TMU@TMU@ TMU@ in ASCII.** The same data is from source node M0 <255:0>. When Write_en = '0', Read_in = '1', the destination node M0 <255:0>, M2 <255:0>, M4 <255:0> , M6 <255:0> , M8 <255:0> , M10 <255:0>, M12 <255:0> and M14 <255:0> . Data_ out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

6.4 RTL and Wave form Simulation for 3D Mesh, Torus,

Ring and Fat tree Topology

The RTL view of the developed 3D mesh, torus, ring and fat tree topology is shown in fig. 6.22 and its internal schematic is depicted in fig.6.23. Internal schematic presents the internal structure having different logic gates and internal hardware required in the design. Table 6.4 is used to discuss the use, size and detail of the pins used for 3D mesh, torus, ring and fat tree topology. The functional verification is carried for 256 bit data transfer and simulation waveforms

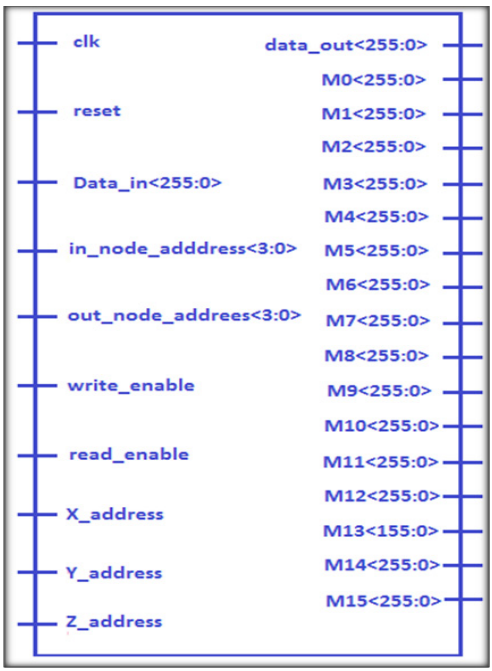


Figure 6.22 RTL view of 3D Topology

Fig. 6.24 and Fig. 6.25 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for 3D mesh topology. Fig. 6.26 and Fig. 6.27 and present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for 3D torus topology. Fig. 6.28 and Fig. 6.29 present the Modelsim result simulation of 256 bit data in hexadecimal

and ASCII for 3D ring topology. Fig. 6.30 and Fig. 6.31 present the Modelsim result simulation of 256 bit data in hexadecimal and ASCII for 3D fat tree topology. The flow chart for the functional behavior is shown in fig. 6.32.

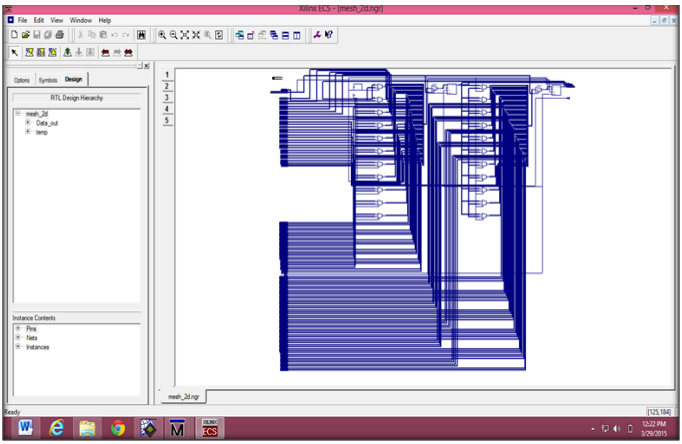


Figure 6.23 Internal schematic of 3D Topology

Table 6.4 Pin explanation for the 3D topology (Mesh, Torus, Ring and Fat tree)

Pin	Size	Functional Description
Reset	1 bit	It is the input pin of std_logic, used to reset all the data and contents of all the ports and registers in the design.
Clk	1 bit	It is 1 bit input and default associated with design used to give the rinsing edge (+ve) edge in the simulation results and works on 50 % duty cycle.

Data_in<255:0>	256 bit	It is the input data packet (256 bit) by the source router and controlled by the control unit of the network.
Data_out<255:0>	256 bit	It is the output data packet (256 bit) by the source router and controlled by the control unit of the network.
In_node_address<8:0>	9 bit	It is the input of source node used to define the input address of the source node need to communicate
Out_node_address<8:0>	9 bit	It is the input for destination node used to define the output address of the target node need to communicate
X_adddres	3 bit	It is the address of the source and target nodes corresponding to X axis
Y_address	3 bit	It is the address of the source and target nodes corresponding to Y axis
Z_address	3 bit	It is the address of the source and target nodes corresponding to Z axis

Write_enable	1 bit	It is the input control signal used to write the contents from source node to destination node. If the write_enable = '1' and read_enable = '0', the contents are written from the control unit to register
Read_enable	1 bit	It is the input control signal used to read the contents from source node to destination node. If the read_enable = '1' and write_enable = '0', the contents are written from the control unit to register
R0<255:0>	256 bit	It is the input/output node carrying 256 bit data
R1<255:0>	256 bit	It is the input/output node carrying 256 bit data
R2<255:0>	256 bit	It is the input/output node carrying 256 bit data
R3<255:0>	256 bit	It is the input/output node carrying 256 bit data
R4<255:0>	256 bit	It is the input/output node carrying 256 bit data

R5<255:0>	256 bit	It is the input/output node carrying 256 bit data
R6<255:0>	256 bit	It is the input/output node carrying 256 bit data
R7<255:0>	256 bit	It is the input/output node carrying 256 bit data
R9<255:0>	256 bit	It is the input/output node carrying 256 bit data
R10<255:0>	256 bit	It is the input/output node carrying 256 bit data
R11<255:0>	256 bit	It is the input/output node carrying 256 bit data
R12<255:0>	256 bit	It is the input/output node carrying 256 bit data
M13<255:0>	256 bit	It is the input/output node carrying 256 bit data
R14<255:0>	256 bit	It is the input/output node carrying 256 bit data
R15<255:0>	256 bit	It is the input/output node carrying 256 bit data
⋮	⋮	⋮
R26<255:0>	256 bit	It is the input/output node carrying 256 bit data

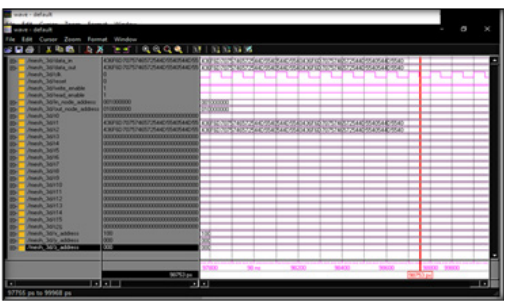


Figure 6.24 Modelsim result simulation of 256 bit data in hexadecimal for 3D mesh topology

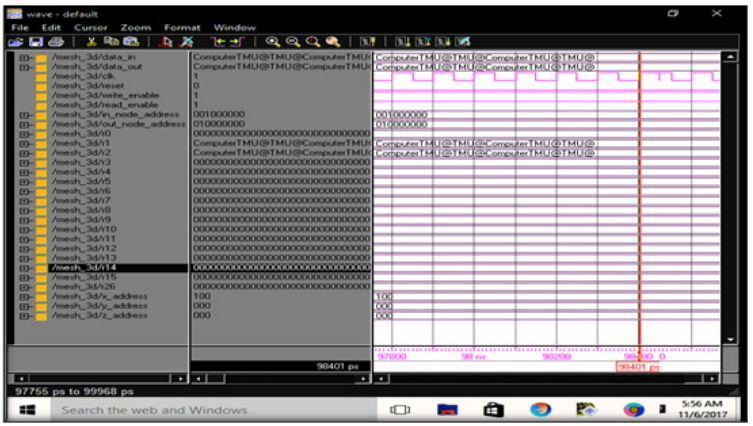


Figure 6.25 Modelsim result simulation of 256 bit data in ASCII for 3D mesh topology

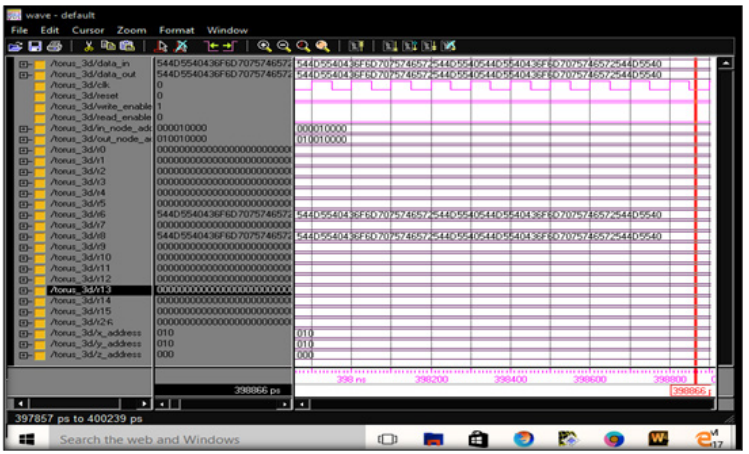


Figure 6.26 Modelsim result simulation of 256 bit data in hexadecimal for 3D Torus topology

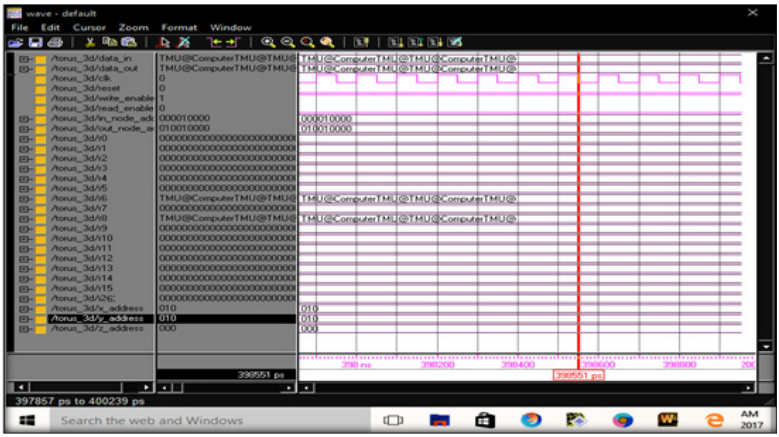


Figure 6.27 Modelsim result simulation of 256 bit data in ASCII for 3D torus topology

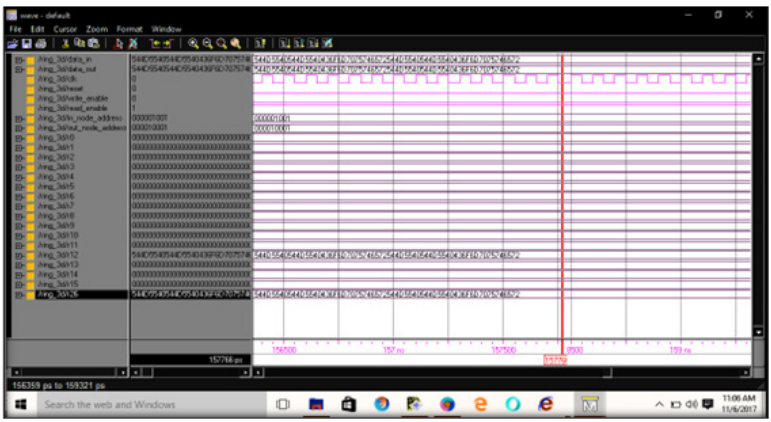


Figure 6.28 Modelsim result simulation of 256 bit data in hexadecimal for 3D ring topology

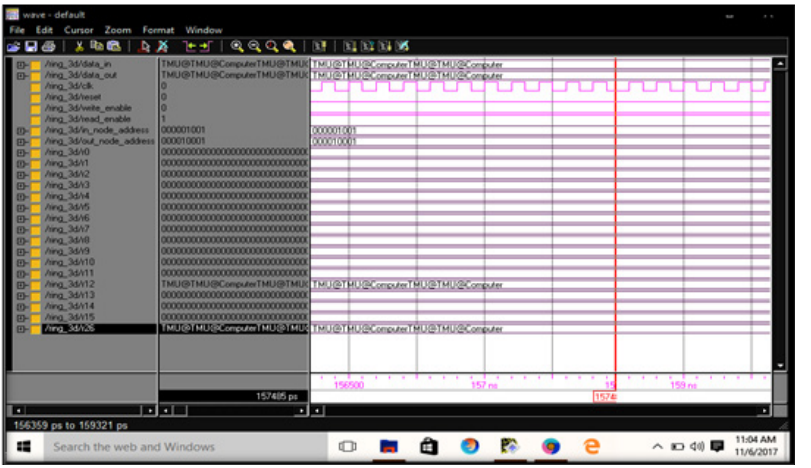


Figure 6.29 Modelsim result simulation of 256 bit data in ASCII for 3D ring topology

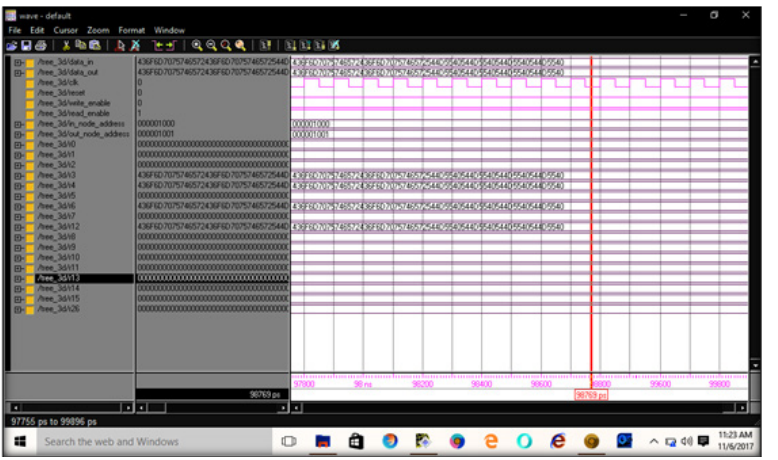


Figure 6.30 Modelsim result simulation of 256 bit data in hexadecimal for 3D tree topology

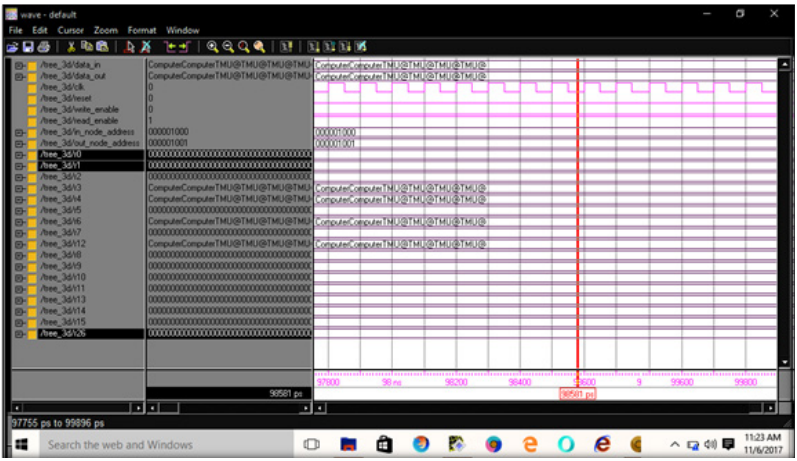


Figure 6.31 Modelsim result simulation of 256 bit data in ASCII for 3D tree topology

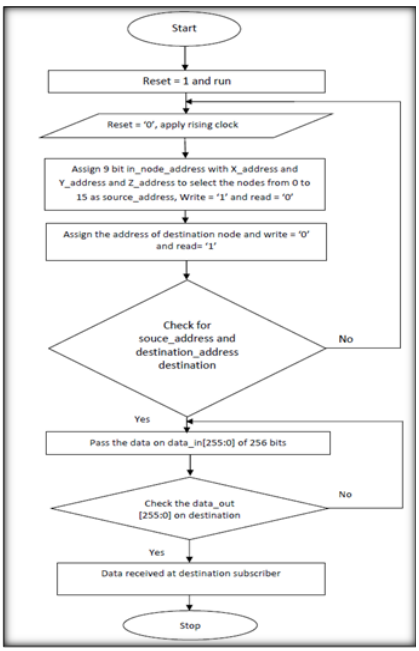


Figure 6.32 Flowchart of 3D NoC topology function

Test -1 (Mesh): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_node_address = "001000000" out_node_address = "010000000", X_address = "010" Y_addess = "000" and Z_ address = "000" based on output node, data_in = "43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40" in hexadecimal or **ComputerTMU@ TMU@ComputerTMU@TMU@ in ASCII.** The same data is from source router R1 <255:0>. When Write_en = '0', Read_in = '1', the destination node R2 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -2 (Torus): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_node_address = "010000000" out_node_address = "010010000", X_address = "010" Y_addess = "010" and Z_ address = "000" based on output node, data_in = "54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40" in hexadecimal or **TMU@ComputerTMU@ TMUComputerTMU@ in ASCII.** The same data is from source router R6 <255:0>. When Write_en = '0', Read_in = '1', the destination node R8 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -3 (Ring): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_node_ address = "000001001" out_node_address = "000010001", based on output node, data_in = "54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72" in hexadecimal or **TMU@TMU@ComputerTMU@ TMU@Computer @ in ASCII.** The same data is from source router R12 <255:0>. When Write_en = '0', Read_in = '1', the destination node R15 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010" in binary.

Test -4 (Fat Tree): First of all reset = '1' and run. It will make all the router or nodes data to zero. Then reset = '0' and give direct positive clock signal. Assign the Write_en = '1', in_node_address = "000001000" out_node_address = "001001000", "000010000", "000001001 based on output node, data_in = "43 6F 6D 70 75 74 65 72 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 54 4D 55 40 54 4D 55 40" in hexadecimal or **ComputerComputer TMU@TMU@TMU@ TMU@ in ASCII.** The same data is from source node M3 <255:0>. When Write_en = '0', Read_in = '1', the destination node M4 <255:0>, M6 <255:0> , M12 <255:0> getting the data. Data_out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

6.5 Hardware Detail on FPGA Synthesis

All the chips are synthesized on Virtex 5 FPGA. The hardware utilization depends on the most of the parameters. The parameters relating to the hardware chip are Number of Slice Flip-flops, Number of Slices,Number of GCLK, Number of 4 input LUTs and Number of Slice Flip-flops. The hardware parameter summary for 2D and 3D NoC router is listed in table 6.5.

Table 6.5 Hardware parameters summary for 2D and 3d NoC Router

Hardware Parameter	2D Router	3D Router
No. of Slices	130 /12480	131/12480
No. of Slice Flipflops	145/12480	148/12480
No. of 4 input LUTs	8/512	9/512
No. of Bounded IOBs	130/172	131/172
No. of GCLK	1	1

The description of the required values of FPGA hardware for 2D mesh, 2D torus, 2D, ring and 2D fat tree topology for cluster size (N= 2, 4, 8, 16, 32, 64, 128, 256) is listed in table 6.6, table 6.7, table 6.8, and table 6.9 respectively. The corresponding graphs for Hardware utilization with cluster size in 2D mesh NoC, 2D torus NoC, 2D ring NoC and 2D fat tree NoC are listed in Fig. 6.33, Fig. 6.34, Fig. 6.35, and Fig. 6.36 respectively.

Table 6.6 Hardware parameters summary for 2D mesh NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	12	34	6	8	1
N=4	20	39	10	16	1
N=8	32	52	24	30	1
N=16	40	60	30	56	1
N=32	46	72	42	84	1
N=64	50	78	52	112	1
N=128	61	86	64	136	1
N=256	82	104	80	148	1

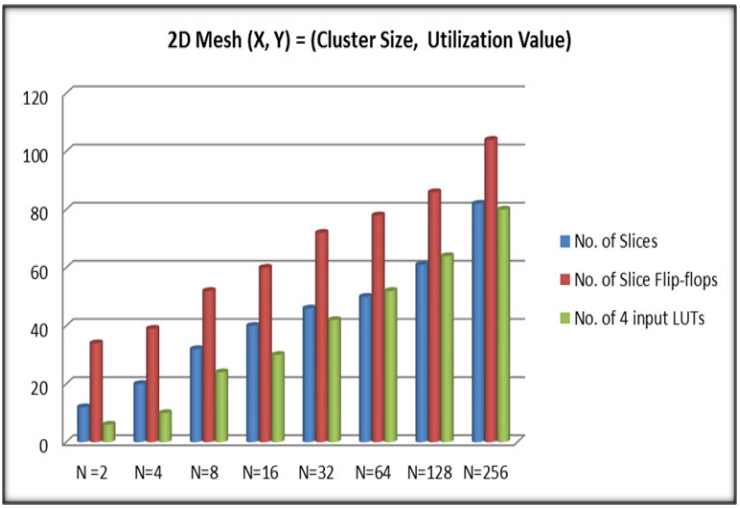


Figure 6.33 Hardware utilization with cluster size in 2D mesh NoC

Table 6.7 Hardware parameters summary for 2D Torus NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	16	38	8	8	1
N=4	22	44	16	16	1
N=8	35	58	26	30	1
N=16	42	65	35	56	1
N=32	49	78	48	84	1
N=64	56	82	55	112	1
N=128	66	90	68	136	1
N=256	**86**	110	92	148	1

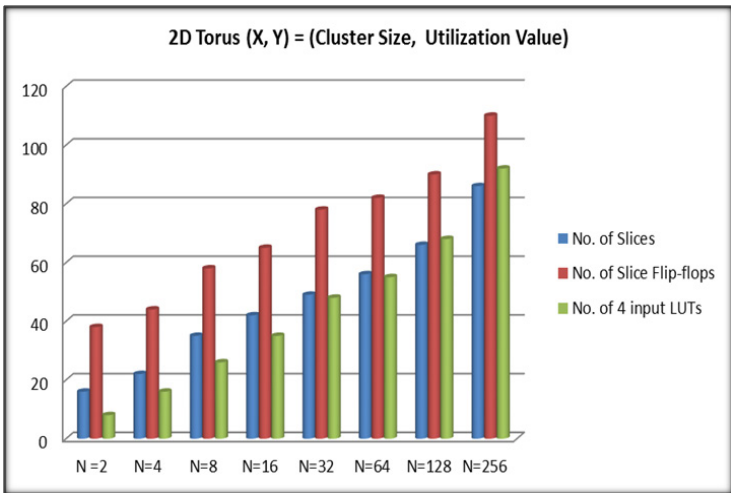


Figure 6.34 Hardware utilization with cluster size in 2D torus NoC

Table 6.8 Hardware parameters summary for 2D ring NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	19	40	9	8	1
N=4	24	46	18	16	1
N=8	36	60	28	30	1
N=16	48	68	36	56	1
N=32	51	84	50	84	1
N=64	58	92	58	112	1
N=128	70	104	72	136	1
N=256	92	110	98	148	1

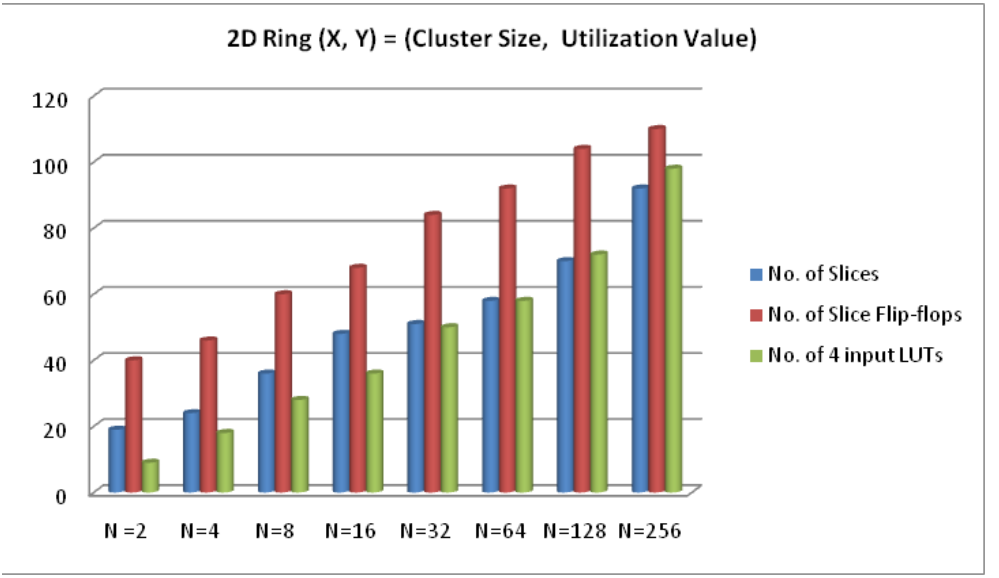


Figure 6.35 Hardware utilization with cluster size in 2D Ring NoC

Table 6.9 Hardware parameters summary for 2D Tree NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	24	42	10	8	1
N=4	28	48	18	16	1
N=8	40	61	30	30	1
N=16	52	69	36	56	1
N=32	55	84	52	84	1
N=64	60	96	58	112	1
N=128	74	110	76	136	1
N=256	96	112	104	148	1

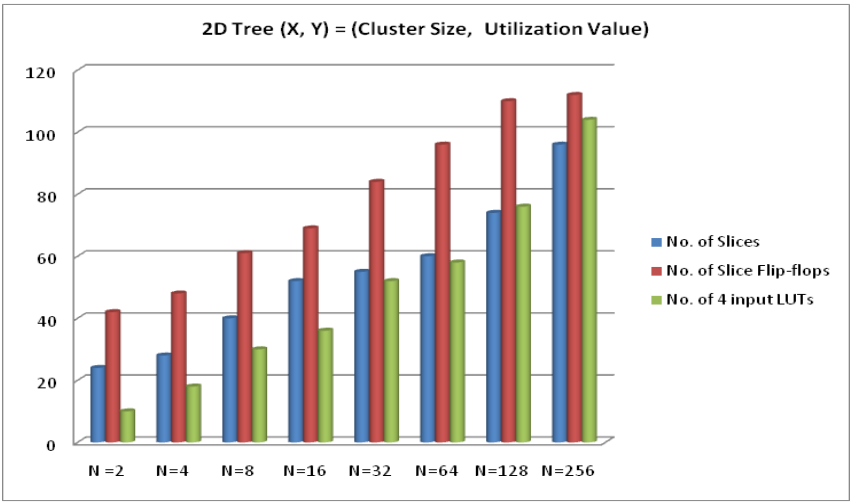


Figure 6.36 Hardware utilization with cluster size in 2D tree NoC

The detailed FPGA hardware for 3D mesh, 3D torus, 3D ring and 3D fat tree topology for cluster size (N= 2, 4, 8, 16, 32, 64, 128, 256) is listed in table 6.10, table 6.11, table 6.12, and table 6.13 respectively. The corresponding graphs for Hardware utilization with cluster size in 3D mesh NoC, 3D torus NoC, 3D ring NoC and 3D fat tree NoC are listed in Fig. 6.37, Fig. 6.38, Fig. 6.39, and Fig. 6.40 respectively.

Table 6.10 Hardware parameters summary for 3D mesh NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	10	33	6	9	1
N=4	18	36	8	18	1
N=8	28	50	23	32	1

N=16	38	57	28	58	1
N=32	45	69	40	86	1
N=64	48	75	51	116	1
N=128	60	83	62	146	1
N=256	80	102	74	160	1

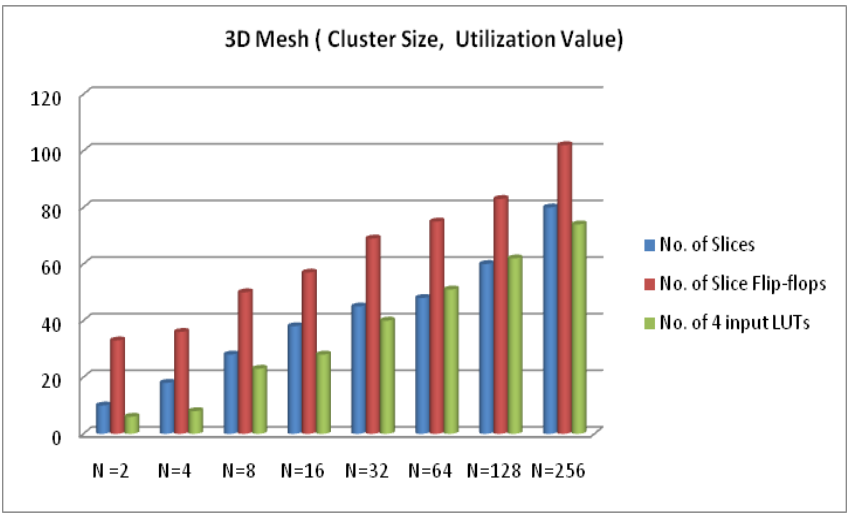


Figure 6.37 Hardware utilization with cluster size in 3D mesh NoC

Table 6.11 Hardware parameters summary for 3D Torus NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	15	36	7	9	1
N=4	20	43	15	18	1
N=8	34	56	25	32	1
N=16	40	64	32	58	1

N=32	46	76	45	86	1
N=64	54	80	52	116	1
N=128	65	88	64	146	1
N=256	82	104	90	160	1

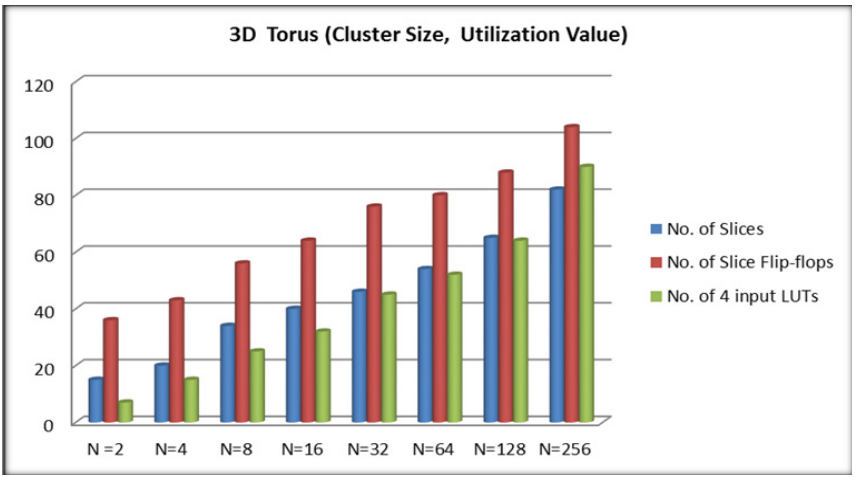


Figure 6.38 Hardware utilization with cluster size in 3D torus NoC

Table 6.12 Hardware parameters summary for 3D ring NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	18	38	8	9	1
N=4	22	45	18	18	1
N=8	34	58	26	32	1
N=16	45	66	35	58	1
N=32	50	80	45	86	1
N=64	56	90	55	116	1
N=128	68	102	70	146	1

N=256	90	106	96	160	1

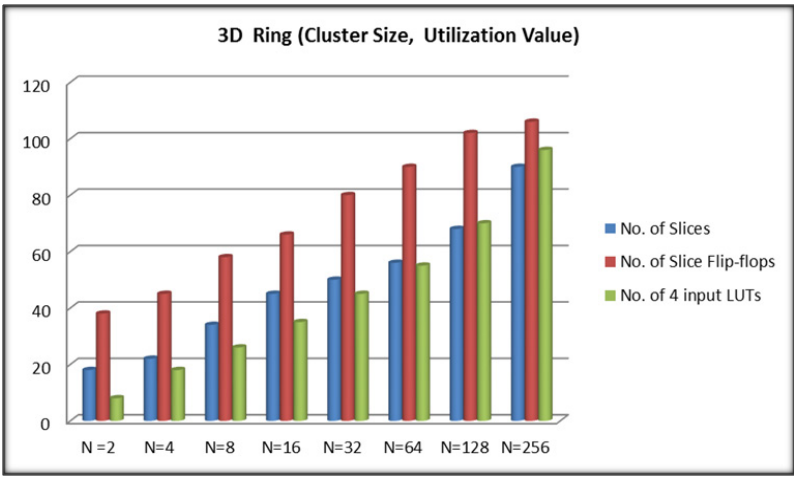


Figure 6.39 Hardware utilization with cluster size in 3D torus NoC

Table 6.13 Hardware parameters summary for 3D Tree NoC

Network Size	Slices Number	Slice Flip-flops number	Four input LUTs number	Bounded IOBs numbers	Number of GCLK
N =2	22	40	10	9	1
N=4	25	46	18	18	1
N=8	39	60	28	32	1
N=16	50	65	36	58	1
N=32	52	82	46	86	1
N=64	58	92	56	116	1
N=128	72	105	74	146	1
N=256	95	110	98	160	1

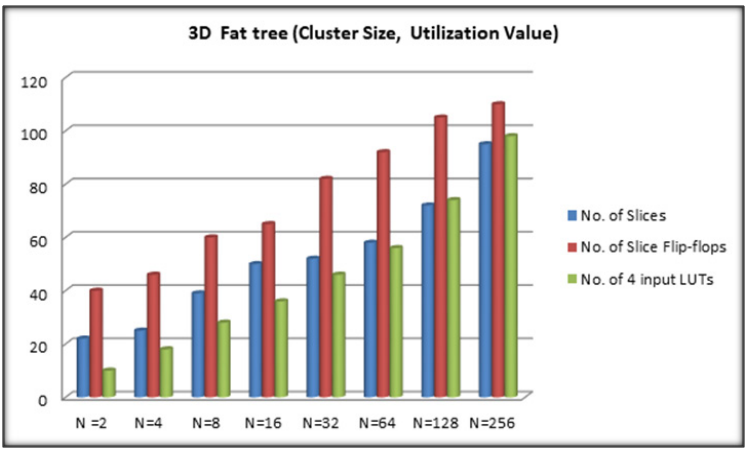


Figure 6.40 Hardware utilization with cluster size in 3D fat tree NoC

It is clear from the figures that the hardware related parameters values is increasing with cluster size of the topology network

6.6 Timing Summary

Timing parameters are very important to understand the network behaviour and performance. The timing parameters are Max Frequency, Time before clk (minimum) (ns), Time after clock (maximum) (ns) and Combinational Path delay (ns). The other parameters are memory usage depends on hardware resources.

Table 6.14 Timing results for 2D and 3D NoC Router

Timing Parameter	2D Router	3D Router
Max Frequency	180.00 MHz	215 MHz
Min Period (ns)	1.496 ns	1.210 ns
Time before clk (minimum) (ns)	4.003 ns	4.870 ns
Time after clock (maximum) (ns)	5.916 ns	5.624 ns
Combinational Path delay (ns)	8.197 ns	9.810 ns
Memory Usage	135411 kB	154923 kB

Table 6.14 Timing results for 2D and 3D NoC Router. The table 6.15, table 6.16, table 6.17, table 6.18 are presenting the results corresponding to 2D mesh, 2D torus, 2D ring and 2D fat tree topology. The timing analysis graph for all topology is presented in Fig. 6.41, Fig. 6.42, Fig. 6.43 and Fig. 6.44 with respect to cluster size (N =2, 4, 8, 16, 32, 64, 128 and 256). From the graph figure it is clear that the timing values such as time before clk (minimum) (ns), time after clock (maximum) (ns) and min period (ns) are increasing with the cluster size of the network. In the same way, table 6.19, Table 6.20, Table 6.21, Table 6.22 are presenting the results corresponding to 3D mesh, 3D torus, 3D ring and 3D fat tree topology. The timing analysis graph for all topology is presented in Fig. 6.45, Fig. 6.46, Fig. 6.47 and Fig. 6.48 with respect to cluster size (N =2, 4, 8, 16, 32, 64, 128 and 256). From these graphs also, it is clear that the timing values such as time before clk (minimum) (ns), time after clock (maximum) (ns) and min period (ns) are increasing with the cluster size of the network.

Table 6.15 Timing detailed parameters for 2D mesh NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)
N =2	135	0.906	2.045	3.011	9123
N=4	145	0.912	2.056	3.089	101450
N=8	170	0.923	2.145	3.110	124781
N=16	195	0.982	2.187	3.188	135190
N=32	215	1.004	2.210	3.214	201012
N=64	235	1.101	2.406	3.248	201130
N=128	400	1.211	2.451	3.349	210412
N=256	925	1.320	2.504	3.451	221005

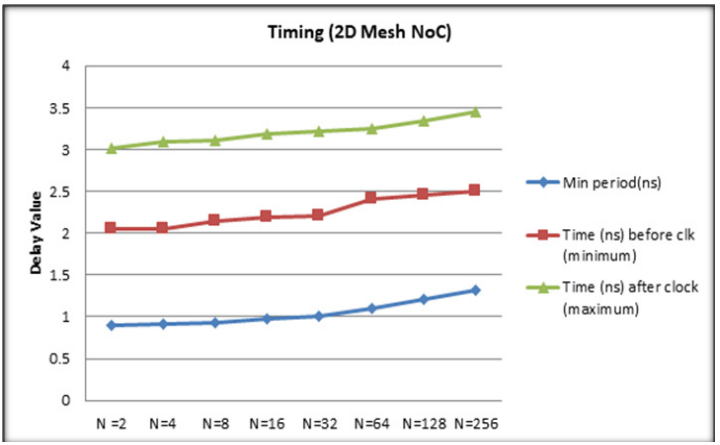


Figure 6.41 Timing values with cluster size in 2D mesh NoC

Table 6.16 Timing detailed parameters for 2D torus NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)
N =2	130	0.880	2.045	3.011	9420
N=4	140	0.902	2.056	3.089	114510
N=8	170	0.911	2.145	3.110	135121
N=16	190	0.971	2.187	3.188	136120
N=32	211	1.000	2.210	3.214	210201
N=64	234	1.009	2.417	3.248	218019
N=128	400	1.194	2.420	3.349	215129
N=256	910	1.318	2.504	3.451	228107

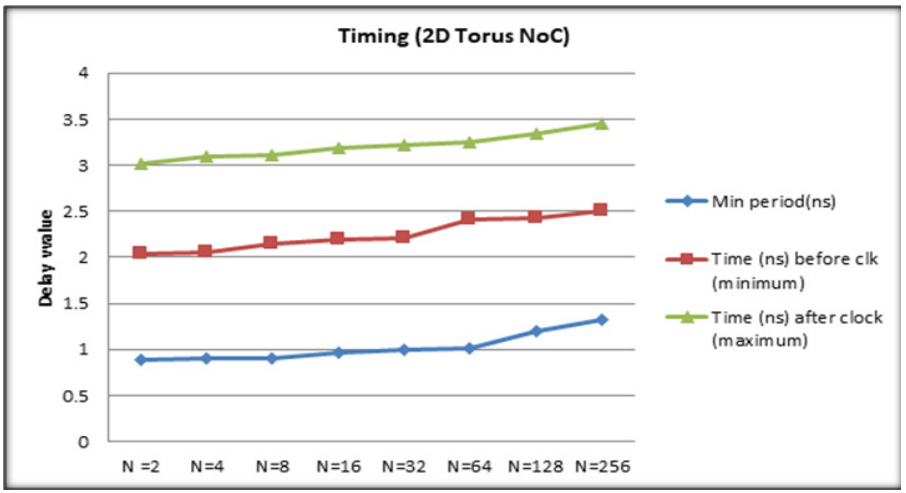


Figure 6.42 Timing values with cluster size in 2D torus NoC

Table 6.17 Timing detailed parameters for 2D ring NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)
N =2	130	0.781	2.012	3.000	9587
N=4	138	0.819	2.051	3.080	122529
N=8	170	0.891	2.089	3.101	138121
N=16	190	0.969	2.178	3.108	136127
N=32	210	0.998	2.200	3.210	214510
N=64	230	1.004	2.416	3.218	218240
N=128	400	1.094	2.417	3.320	223412
N=256	900	1.214	2.459	3.381	251318

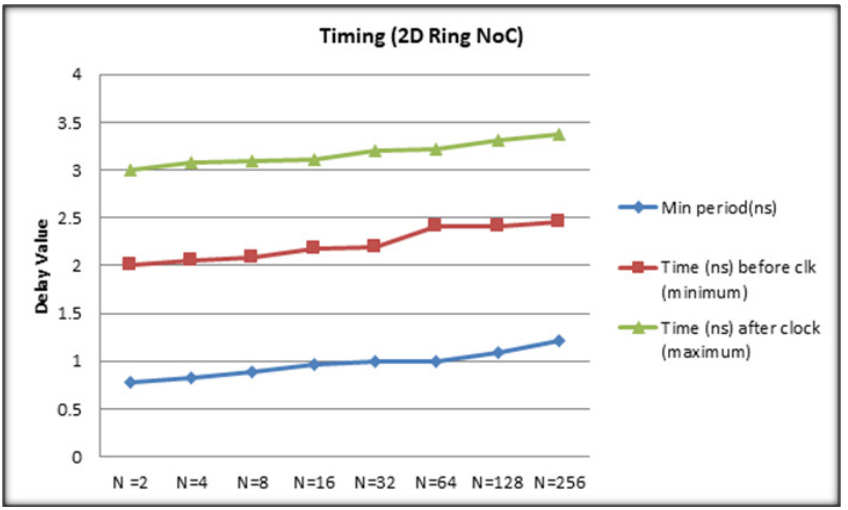


Figure 6.43 Timing values with cluster size in 2D ring NoC

Table 6.18 Timing detailed parameters for 2D tree NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)	Speed Grade
N =2	130	0.779	2.008	2.981	9540	-5
N=4	135	0.790	2.001	3.001	122295	-5
N=8	175	0.882	2.010	3.019	137256	-5
N=16	185	0.911	2.150	3.100	135210	-5
N=32	210	0.921	2.195	3.182	213219	-5
N=64	225	0.917	2.391	3.205	219615	-5
N=128	390	1.002	2.398	3.209	231297	-5
N=256	875	1.179	2.410	3.211	251200	-5

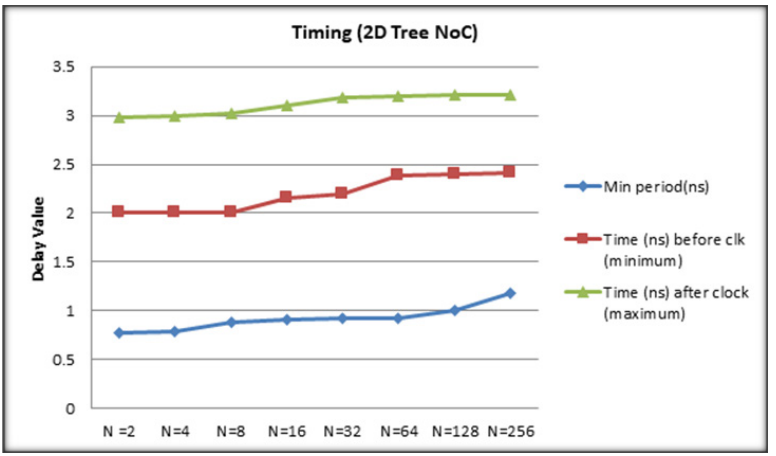


Figure 6.44 Timing values with cluster size in 2D tree NoC

Table 6.19 Timing detailed parameters for 3D mesh NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)	Speed Grade
N =2	235	0.912	2.251	4.239	125681	-5
N=4	320	1.023	2.348	4.312	231467	-5
N=8	365	1.187	2.451	4.417	281369	-5
N=16	400	1.267	2.488	4.512	312034	-5
N=32	425	1.371	2.512	4.621	383412	-5
N=64	675	1.429	2.598	4.901	421628	-5
N=128	825	1.467	2.613	5.014	456721	-5
N=256	987	1.560	2.718	5.128	491234	-5

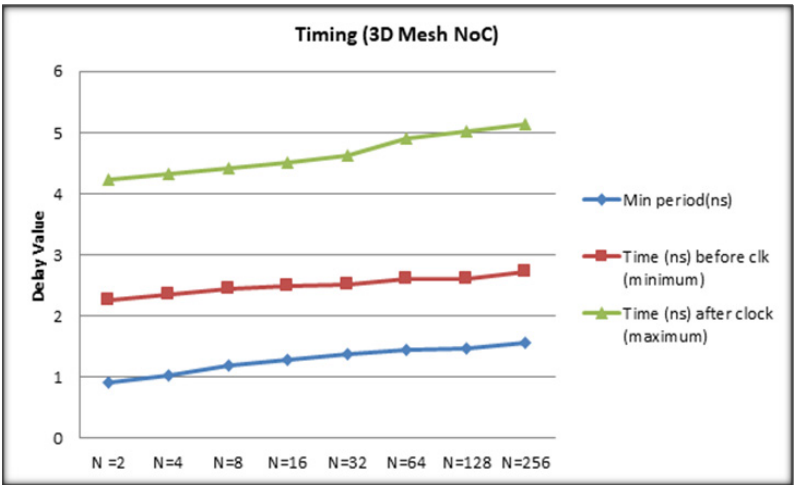


Figure 6.45 Timing values with cluster size in 3D mesh NoC

Table 6.20 Timing detailed parameters for 3D torus NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)	Speed Grade
N =2	232	0.907	2.119	4.230	135967	-5
N=4	315	1.008	2.241	4.300	233456	-5
N=8	360	1.117	2.317	4.399	291249	-5
N=16	395	1.216	2.417	4.401	320120	-5
N=32	418	1.370	2.448	4.599	394312	-5
N=64	667	1.410	2.490	4.812	437827	-5
N=128	821	1.450	2.531	5.001	467612	-5
N=256	950	1.500	2.614	5.119	501234	-5

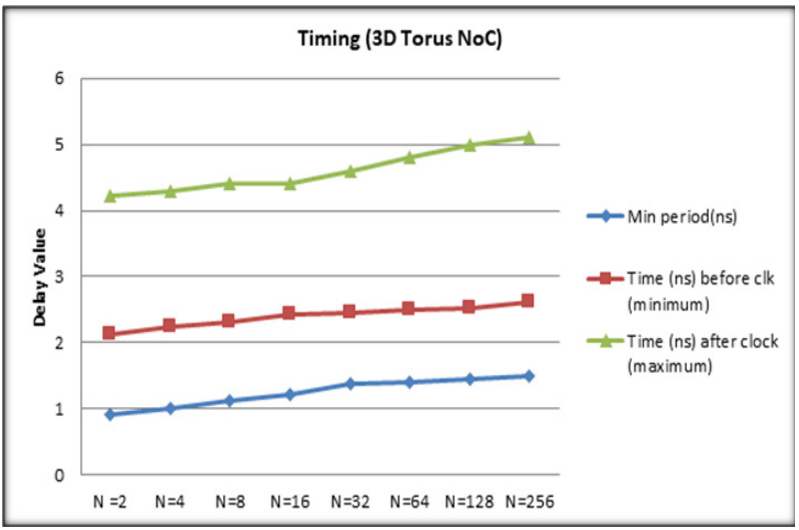


Figure 6.46 Timing values with cluster size in 3D torus NoC

Table 6.21 Timing detailed parameters for 3D ring NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)	Speed Grade
N =2	230	0.900	2.110	4.190	144176	-5
N=4	314	0.991	2.214	4.212	240065	-5
N=8	356	1.017	2.210	4.291	300347	-5
N=16	387	1.196	2.317	4.301	330001	-5
N=32	412	1.370	2.390	4.412	403419	-5
N=64	650	1.391	2.399	4.617	448772	-5
N=128	810	1.450	2.500	4.990	486721	-5
N=256	925	1.500	2.599	5.091	521009	-5

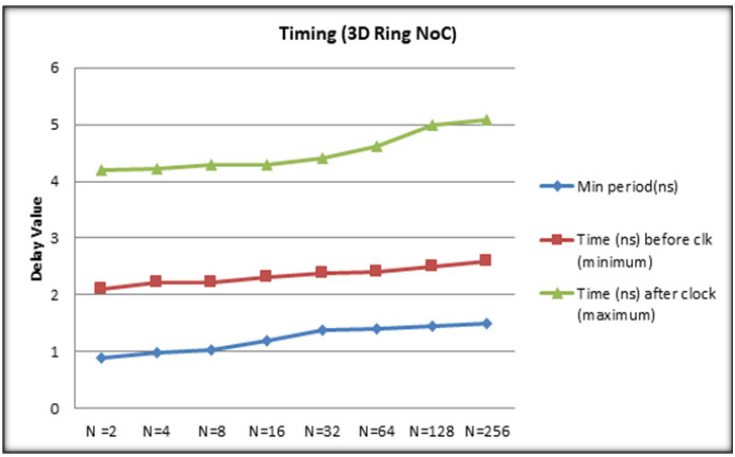


Figure 6.47 Timing values with cluster size in 3D ring NoC

Table 6.22 Timing detailed parameters for 3D fat tree NoC

Network Size	Max Frequency	Min period(ns)	Time (ns) before clk (minimum)	Time (ns) after clock (maximum)	Memory Usage (kB)	Speed Grade
N =2	214	0.812	1.959	3.790	159176	-5
N=4	310	0.928	2.140	3.794	252129	-5
N=8	350	1.009	2.199	4.091	324560	-5
N=16	378	1.161	2.371	4.191	354519	-5
N=32	400	1.271	2.310	4.239	444523	-5
N=64	610	1.311	2.319	4.587	483219	-5
N=128	799	1.405	2.419	4.782	524217	-5
N=256	900	1.500	2.581	4.961	551756	-5

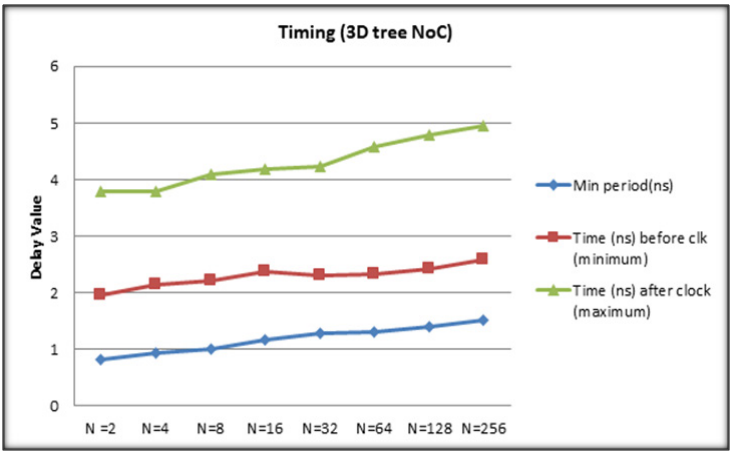


Figure 6.48 Timing values with cluster size in 3D fat tree NoC

6.7 Comparative Analysis

The comparative analysis of the 2D and 3D mesh, torus, ring and fat tree is done with respect to hardware usage and timing values obtained from the simulation on FPGA synthesis results. The comparison among the topology is done based on Frequency support, memory usage, No. of slices and flip-flops required. Fig. 6.49 and Fig. 6.50 presented the frequency (MHz) analysis for 2D (mesh_torus_ring_fatree) and 3D (mesh_torus_ring_fatree) respectively. From the graph, it is clear that mesh has higher frequency support with cluster size. The frequency of 2D and 3D mesh, torus, ring and fat tree is increasing with cluster size. It signifies that the system is becoming faster with cluster size. It is also important to note that the maximum frequency support is in mesh in compassion to torus, ring and fat tree. Fig. 6.51 and fig. 6.52 are presenting the memory usage for 2D (mesh_torus_ring_ fatree) and 3D (mesh_torus_ring_fatree).It is clear that mesh topology is having less memory in comparison to torus, ring and fat tree. It is also noticed that the memory is increasing with cluster size increment. It is obvious that it will increase

because memory directly related to slices and flip-flops. Fig. 6.53 and fig. 6.54 are presenting the slices utilization for 2D (mesh_torus_ring_fatree)and 3D (mesh_torus_ring_fatree). Fig. 6.55 and fig. 6.56 are presenting the flip flops usage for 2D (mesh_torus_ring_fatree)and 3D (mesh_torus_ring_fatree). The slices and flip flops are increasing with cluster size but found of less utilization in mesh topology in comparison to torus, ring and fat tree topology.

When the designer is designing the chip then the slices and flip-flops values can be optimized based on the hardware optimization techniques. The details of all the parameters are extracted directly from the chip design software Xilinx 14.2. The other sub parameters are number of gates utilization, number of multiplexers, number of DSS an element by those are not so important in comparison to memory usage, slices utilization and flip flips. All the simulation results concluded that the 3D mesh topology is the optimal solution in terms of performance, hardware and memory in comparison to the 3D torus, ring and fat tree topology. Another important fact it that the nodes can communicate with equal distance and throughput in the topology is more.

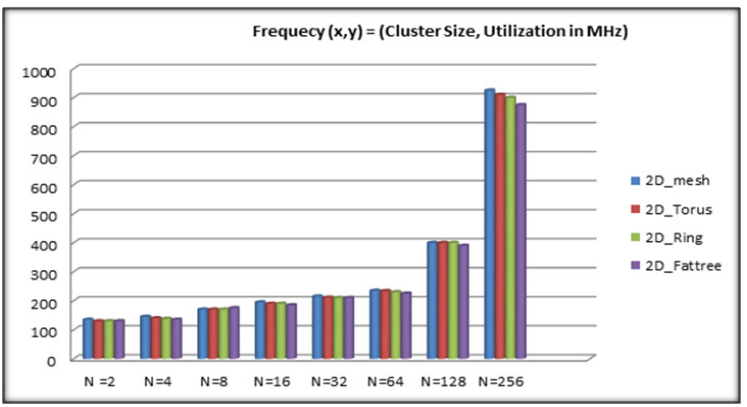


Figure 6.49 Frequency analysis for 2D (mesh_torus_ring_ fatree)

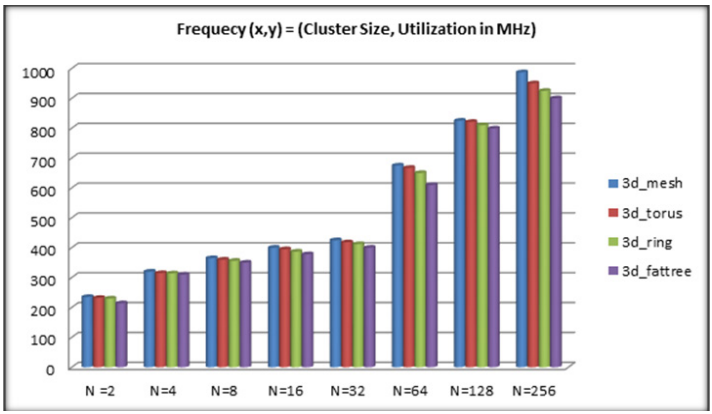


Figure 6.50 Frequency analysis for 3D (mesh_torus_ring_ fatree)

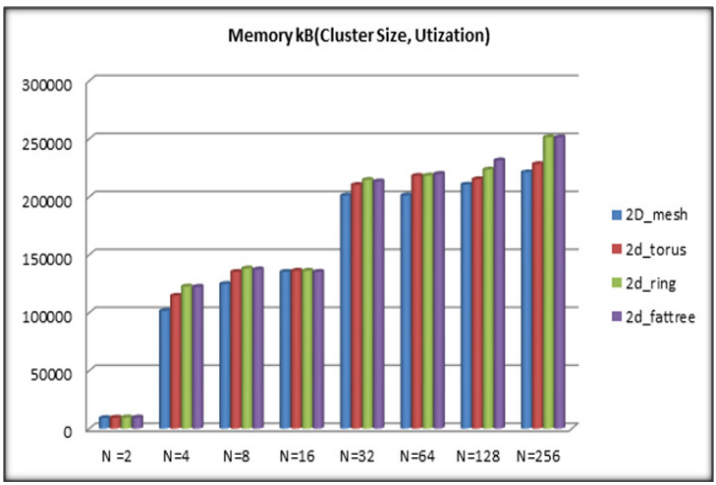


Figure 6.51 Memory usage for 2D (mesh_torus_ring_fatree)

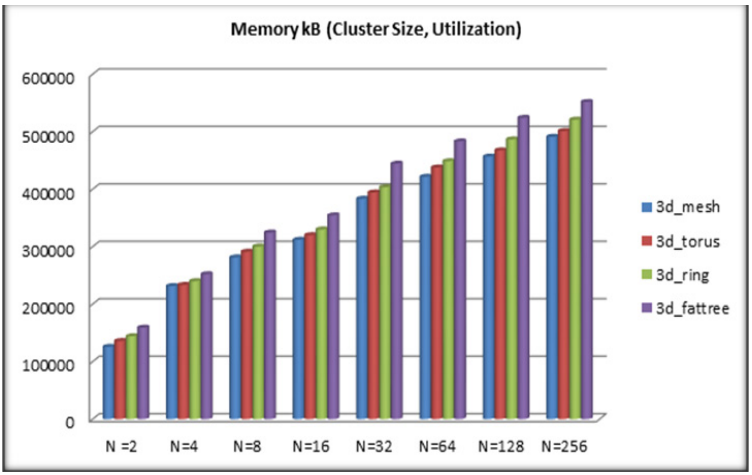


Figure 6.52 Memory usage for 3D (mesh_torus_ring_fatree)

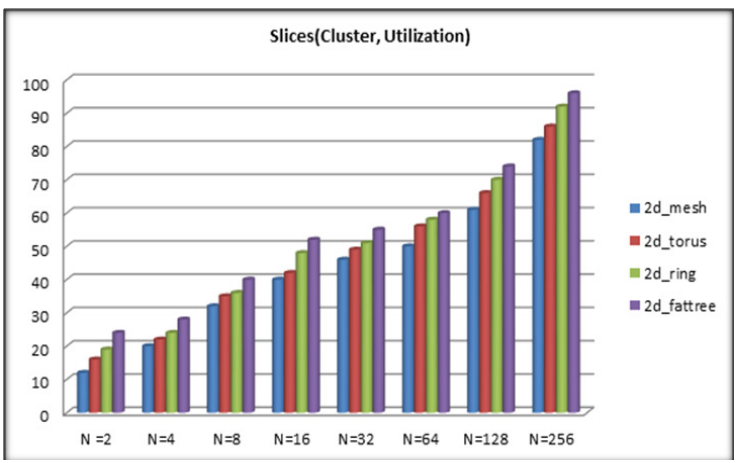


Figure 6.53 Slices utilization for 2D (mesh_torus_ring_ fatree)

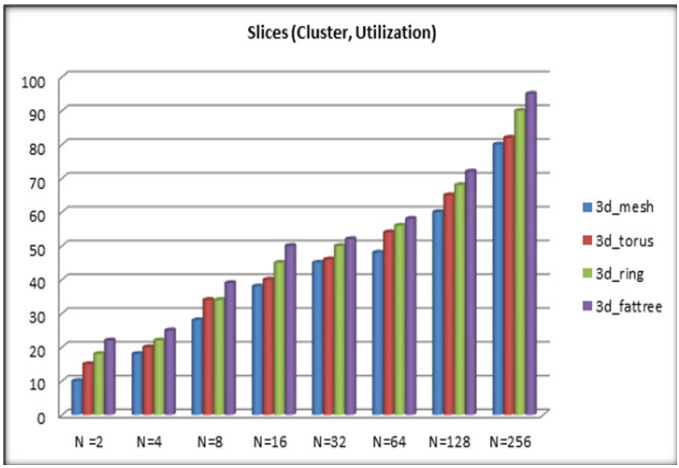


Figure 6.54 Slices utilization for 3D (mesh_torus_ring_ fatree)

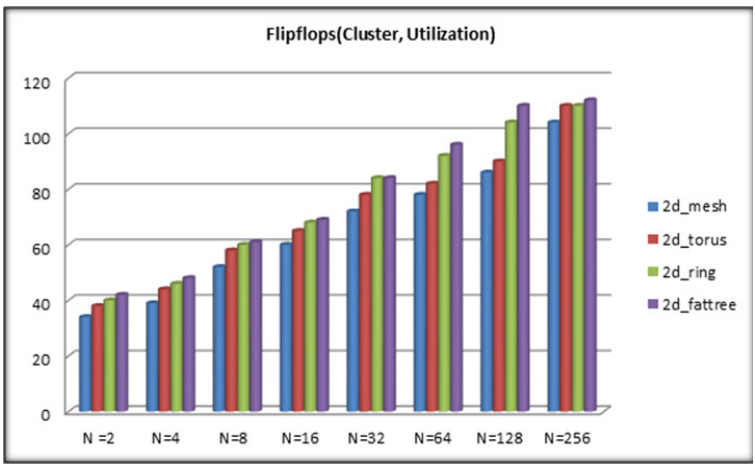


Figure 6.55 Flipflop utilization for 2D (mesh_torus_ring_ fatree)

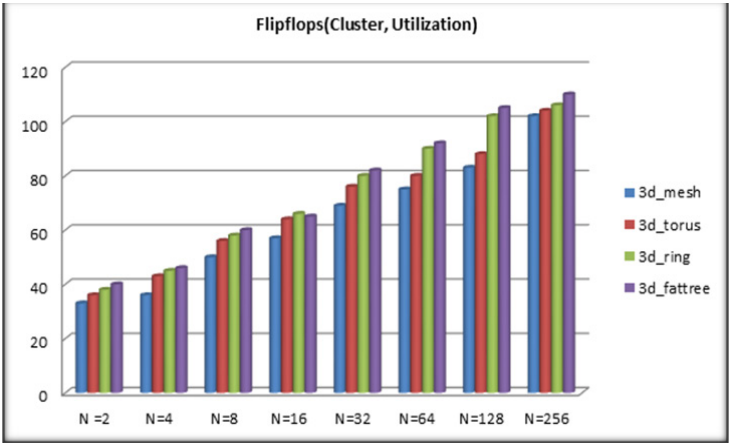


Figure 6.56 Flipflop for 3D (mesh_torus_ring_fatree)

6.8 Verification and Validation

The design of all the developed chip is verified using some test input and validate in the FPGA hardware shown in fig. 6.57 and Fig. 6.58. The control input is given using the VHDL code is burned in Virtex 5 FPGA and the data of the destination node is shown output screen of another PC. It is Virtex 5 FPGA having XC5VLX110T device, manufactured by Digilent Company. Test input are also given using input switches and verified on LEDs as output for small data. The data is also shown on the PC using VGA. Fig.6.59 shows the successful hardware burning of the VHDL program. Fig. 6.60 shows the experimental validation of the receiving the data **TMU@TMU@ComputerTMU@TMU@**Computeron Computer Screen using VGA cable. The verification is done for the following test samples.

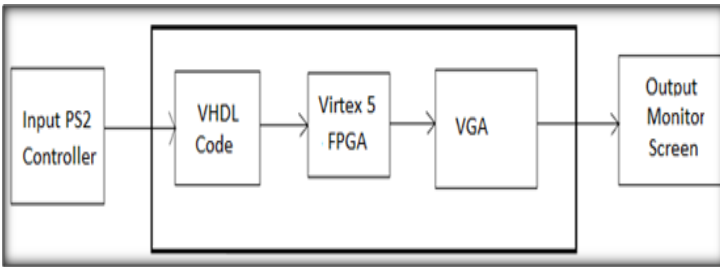


Figure 6.57 FPGA verification process

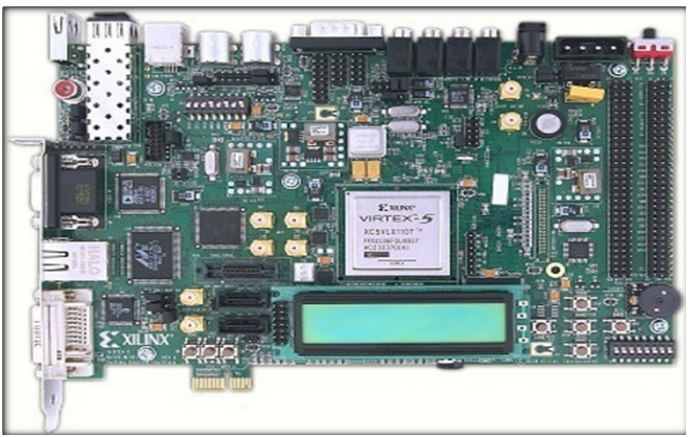


Figure 6.58 Virtex 5 FPGA in pictorial View

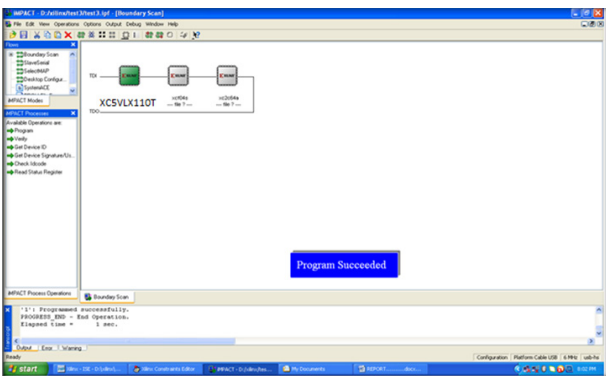


Figure 6.59 FPGA with successful program synthesis

Figure 6.60 Experimental validation and verification

Test -1 (Mesh): in_node_address = "001000000" out_node_address = "010000000", X_address = "010" Y_addess = "000" and Z_address = "000" based on output node, data_in = "43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40" in hexadecimal or **ComputerTMU@TMU@ComputerTMU@TMU@ in ASCII.** Source router R1 <255:0>. The destination node R2 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -2 (Torus): in_node_address = "010000000" out_node_address = "010010000", X_address = "010" Y_addess = "010" and Z_address = "000" based on output node, data_in = "54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 7254 4D 55 40" in hexadecimal or TMU@ ComputerTMU@TMUComputerTMU@ in ASCII. The same data is from source router R6 <255:0>. The destination node

R8 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

Test -3 (Ring): in_node_address = "000001001" out_node_ address = "000010001", based on output node, data_in = "54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 43 6F 6D 70 75 74 65 72" in hexadecimal or TMU@TMU@ComputerTMU@TMU@Computer @ in ASCII. The destination node R15 <255:0> and data_out< 255:0> are getting the same data. Data_out = "0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0100 0110 0101 0111 0010" in binary

Test -4 (Fat Tree): in_node_address = "000001000" out_node_ address = "001001000", "000010000", "000001001 based on output node, data_in = "43 6F 6D 70 75 74 65 72 43 6F 6D 70 75 74 65 72 54 4D 55 40 54 4D 55 40 54 4D 55 40 54 4D 55 40" in hexadecimal or ComputerComputer TMU@TMU@TMU@ TMU@ in ASCII. The same data is from source node M3 <255:0>, the destination node M4 <255:0>, M6 <255:0> , M12 <255:0> getting the data. Data_out< 255:0> are getting the same data. Data_out = "0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0100 0011 0110 1111 0110 1101 0111 0000 0111 0101 0111 0101 0110 0101 0111 0010 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000 0101 0100 0100 1101 0101 0101 0100 0000" in binary.

The data of all the test input is verified on FPGA connected on monitor screen

CONCLUSION & FUTURE SCOPE

The chapter concludes the research work with conclusion and small discussion is also done on future work and ended with the references used to support the research work in nice manner.

7.1 Conclusions

NoC is the network version of any communication chip and new approach to provide signaling and environment to utilize the efficient interconnections and link with the verification of current system on chips. NoC based design are helpful in reduction of the extra hardware used in the design and complex chips interconnections. The low and slow band signals are grouped and multiplexed over a single line and larger bandwidth signals can communication directly using high speed in 2D and 3D topology with parallel paths or ports. The NoC communication of the network is done in 2D mesh, 2D torus, 2D ring and 2D fat tree topology using 2D router. The 2D router has 5 parallel ports to communicate. The NoC communication of the network is done also for 3D mesh, 3D torus, 3D ring and 3D fat tree topology using 3D router. The 3D router has 7 parallel ports to communicate. The chip design, simulation and synthesis of the 2D mesh, 2D torus, 2D ring and 2D fat tree NoC is done for cluster size (N= 2, 4, 8, 16, 32, 64, 128, 256). In the same way the chip design, simulation and synthesis of the 3D mesh, 3D torus. 3D ring and 3D fat tree NoC is done for cluster size (N= 2, 4, 8, 16, 32, 64, 128, 256). The data transfer for the 2D NoC and 3D NoC is verified with the destination node and intercommunicating network. The results are verified for different test cases under data transfer scheme.

- In comparison to the 2D and 3D topological structure, 3D NoC is more efficient and reliable because the data

can be received by 3D router using 7 parallel ports and scheduling is possible

- The hardware utilization and related parameters such as slices, flip-flops, LUTs and Input and output blocks IOBs of 2D mesh, 2D torus, 2D ring and 2D fat tree topology is increasing with the increase in network configuration or the cluster size of the network (N= 2, 4, 8, 16, 32, 64, 128, 256).
- The hardware utilization and time related parameters such as slices, flip-flops, LUTs and Input and output blocks IOBs of 3D mesh, 3D torus, 3D ring and 3D fat tree topology is increasing with the increase in network configuration or the cluster size of the network (N = 2, 4, 8, 16, 32, 64, 128, 256).
- The hardware utilization, memory and timing values/ delay are obvious because network configuration or size is increasing.
- The frequency support and memory usage is also increasing with network cluster (N= 2, 4, 8, 16, 32, 64, 128, 256).
- 3D mesh topology has less hardware, memory utilization, flip flops, slices, LUTs in comparison to other topology such as 3D torus, 3D ring and 3D fat tree topology.
- The frequency support in FPGA hardware for 3D mesh topology is larger in comparison to 3D torus, 3D ring and 3D fat tree topology. It guarantees the designed chip will be more faster in comparison to 3D torus, 3D ring and 3D fat tree topology.
- The router design, FPGA implementation of 2D and 3D mesh is scalable in comparison to other topology. The regular design and structure feature of the mesh NoC is the greatest attraction of all designer because there may be another possible link in case of the failure of the specific link for internode communication. Nodes are placed on the equal distance and address by XYZ routing.

The hardware implementation is a great revolution in the chip technology to optimize the hardware and to configure the programmable interconnects. The main advantage of the programmable structure is that we can identify the faulty node and replace the node easily with configurable structure. From the device utilization parameters and timing parameters it is concluded that 3D mesh has the optimal design, switching capability and higher response comparison to ring NoC. It signifies that torus, ring and fat tree NoC consumes more memory and hardware in comparison to mesh 3DNoC. The frequency support of 3D mesh NoC is larger in comparison to NoC. Hence mesh NoC has faster response in comparison to ring NoC. The geographical area covered by the ring NoC is larger than mesh NoC. Hence it supports the wide band width and coverage. Mesh NoC crossbar structure and has addressing based on row and column address. The geographical area covered by mesh NoC is less than ring NoC. Overall the performance, speed, hardware utilization of developed 3D mesh NoC is the best solution for the development of large scale and programmable switching networksand the industries will be benefited by the research working in the field of computer networks and programmable architectures for the implementation of specific network

7.2 Future Work

In future, we can configure the more number of nodes in 2D and 3D network structures. The number of networks for intercommunication in 2D and 3D NoC can also be increased based on network selection logic. We can implement the same concept for specific wireless sensor network. We can also add the features of security by encryption and decryption of data transfer among nodes. The chip design and development of programmable NoC is a great revolution for implementing programmable switches, reconfigurable NoC and multimedia networks. It is also possible to develop the NoC for wireless communication. The proposed architecture is applicable for inter and intra communication among networks.

Bibliography

1. C. Liang, X.Huang, Mapping parallel fft A. Das, A. Kumar, Fault-aware task re-mapping for throughput constrained multimedia applications on NoC-based MPSoCs, Rapid System Prototyping (RSP), 2012 23rd IEEE International Symposium on IEEE, 2012, October, 149– 155.

2. A. Das, A.K. Singh, A. Kumar, Energy-aware dynamic reconfiguration of communication-centric applications for reliable MPSoCs. Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC), 2013 8th International Workshop on IEEE, 2013, July, 1–7.

3. A. Kamalizad, C. Pan, N. Bagherzadeh, Fast parallel fft on a reconfigurable computation platform, in: Proceedings of the 15th Symposiumon Computer Architecture and High Performance Computing, 2003, pp. 254–259.

4. A. Kumar Singh, T. Srikanthan, A. Kumar, W. Jigang, Communication-aware heuristics for run-time task mapping on NoC-based MPSoC platforms, J. Syst. Architect. 56 (7) (2010) 242–255.

5. A. Kumar Singh, W. Jigang, A. Kumar, T. Srikanthan, Run-time mapping of multiple communicating tasks on MPSoC platforms, International Conference on Computational Science, ICCS 2010, May 2010, 1 (1) (2010) 1019–1026.

6. A. Samih, R. Wang, A. Krishna, C. Maciocco, C. Tai, Y. Solihin, Energy-efficient interconnect via router parking, in: Proceedings of the 19th International Symposium on High Performance Computer Architecture, IEEE, 2013, pp. 508– 519

7. Aamir Zia, Sachhidh Kannan, H. Jonathan Chao, Garrett S. Rose "3D NOC for many-core processors" Microelectronics Journal Vol. 42 (2011), PP (1380–1390).

8. Adesh Kumar, LopamudraBaruah, Amit Sabu "Rotator on Chip (RoC) design based on ring topological NoC" Procedia Computer Science, Elsevier Vol. 45, 2015 pp (540 – 548)

9. Ahmed Hemani, Axel Jantsch, Shashi Kumar, Adam Postula, Johnny Öberg, Mikael Millberg, Dan Lindqvist, Network on a chip: an architecture for billion transistor era, 18th NORCHIP Conference, Turku, Finland, ovember 2000, 166–173.

10. Amir-Mohammad Rahmani, PasiLiljeberga, JuhaPlosila, HannuTenhunen"Developing a power-efficient and low-cost 3D NoC using smart GALS-based vertical channels" Journal of Computer and System Sciences Vol. 79 (2013) pp(440–456)

11. Andreas Hansson, KeesGoossens and Andrei Radulescu "A Unified Approach to Mapping nd Routing on a Network-on-Chip for Both Best Effort and Guaranteed Service Traffic" indawi Publishing Corporation VLSI Design Volume 2007, pp (1-16).

12. Armin Mehran, Ahmad Khademzadeh, Samira Saeidi, DSM: a heuristic dynamic spiral mapping algorithm for network on chip, IEICE Electron. Exp. 5 (13) (2008) 464–471.

13. Ashwini Raina, V. Muthukumar, Traffic aware scheduling algorithm for network on chip, Sixth International Conference on Information Technology: New Generations, 2009, 877–882.

14. Atitallah, Rabie Ben, et al., Estimating energy consumption for an MPSoC architectural exploration, Architecture of Computing Systems-ARCS 2006, Springer Berlin Heidelberg, 2006, 298–310.

15. Aurel A. Lazar, "Programming Telecommunication Networks" IEEE Network, September 1997, pp (8-19).

16. B. Daya, C.-H. Chen, S. Subramanian, W. C. Kwon, S. Park,T. Krishna, J. Holt, A. Chandrakasan, L. S. Peh, SCORPIO: a 36-core research chip demonstrating snoopy coherence on a scalable mesh NoC with in-networkordering, in: Proceedings of the ACM / IEEE 41st International Symposium on Computer Architecture, ACM, 2014, pp. 25–36.

17. Bharati B. Sayankar, S.S. Limaye, Overview of network on chip architecture, 2nd National Conference on Information and Communication Technology (NCICT), 2011.

18. Bin. Xie, Tianzhou. Chen, Hu. Wei, Xingsheng. Tang, Dazhou. Wang, An energyaware online task mapping algorithm in NoC-based system, J. Supercomput. 64 (3) (2013) 1021–1037.

19. algorithm onto smart cell coarse-grained reconfigurable architecture, IEICE Trans. Electron. E93- C(3) (2010) 407–415.

20. Carvalho, Ewerson, Ney Calazans, Fernando Moraes, Heuristics for dynamic task mapping in NoC-based heterogeneous MPSoCs, Rapid System Prototyping, 2007, RSP 2007, 18th IEEE/ IFIP International Workshop on IEEE, 2007.

21. Chen-Ling Chou, Radu Marculescu, Contention-aware application mapping for Network-on-Chip communication architectures, IEEE International Conference on Computer Design, 2008, 164–169.

22. Chen-Ling Chou, Radu Marculescu, Run-time task allocation considering user behavior in embedded multiprocessor networks-on-chip, IEEE Trans. Comput.- Aided Des. Integr. Circuits Syst. 29 (1) (2010).

23. Chen-Ling Chou, Radu Marculescu, User-aware dynamic task allocation in networks-on-chip, Design, Automation and Test in Europe, 2008, 1232–1237.

24. Ciprian Radu,Md. Shahriar Mahbub and Lucian Vintan.," Developing Domain-Knowledge Evolutionary Algorithms for Network-on-Chip Application Mapping.",Microprocessors and Microsystems 37 (2013), page no. 65–78,Elsevier.

25. D. Bertozzi, A. Jalabert, S. Murali, R. Tamhankar, S.Stergiou, L.Benini, G.DeMicheli, Noc synthesis flow for customized domain specific multiprocessor systems-on-chip, Parallel Distrib. Syst., IEEE Trans. 16(2)(2005) 113–129

26. David Atienzaa, Federico Angiolini, Srinivasan Murali, Antonio PullinidLucaBeninic, Giovanni De Michelia, "Network-on-Chip design and synthesis outlook" Integration The VLSI Journal Elsevier, Vol. 41, 2008, pp(340-359).

27. De Moura, Leonardo, Nikolaj Bjørner. Z3: An Efficient SMT

solver. Tools and Algorithms for the Construction and Analysis of Systems, Springer, Berlin Heidelberg, 2008, pp. 337–340.

28. E. Oruklu, X. Xiao, J.Saniie, Reduced memory and low power architectures for cordic-based fft processors, J. Signal Process. Syst., 66 (2) (2012) 129–134

29. E.W. Brião, D. Barcelos, F. Wronski, F.R. Wagner, Impact of task migration in NoC-based MPSoCs for soft real-time applications, Very Large Scale Integration, 2007, VLSI-SoC 2007, IFIP International Conference on IEEE, 2007, October, 296–299.

30. Eman Kamel Gawish, M. Watheq El-Kharashi, M.F.Abu-Elyazeed "Variability-tolerant routing algorithms for Networks-on-Chip" Microprocessors and Microsystems Vol. 38 (2014) pp(1037–1045)

31. Ewerson Carvalho, César Marcon, Ney Calazans, Fernando Moraes, Evaluation of static and dynamic task mapping algorithms in NoC-based MPSoCs, International Symposium on System-on-Chip, (2009) 087–090.

32. Ewerson Carvalho, Ney Calazans, Fernando Moraes, Heuristics for dynamic task mapping in NoC-based heterogeneous MPSoCs, 18th IEEE/IFIP International Workshop on Rapid System Prototyping, 2007, 34–40.

33. Feng Wang, Xiantuo Tang, Zuocheng Xing "Applying partial power-gating to bit-sliced network-on-chip" Microelectronics Journal, Elsevier Vol. 46(2015), pp (1002–1011).

34. Feng Wang, Xiantuo Tang and Zuocheng Xing., "Applying Partial Power-Gating to Direction-Sliced Network-on-Chip.", Vol. 2015 (2015), Article ID 862387, Journal of Electrical and Computer Engineering, Hindawi.

35. Ganghee Lee, Kiyoung Choi, and Nikil D. Dutt, "Mapping Multi-Domain Applications onto Coarse Grained Reconfigurable Architectures" IEEE Transactions on Computer Aided Design on Integrated Circuits and Systems, Vol. 30, No.5, May 2011, pp (637-650)

36. Haoyuan Ying, Ashok Jaiswal, Thomas Hollstein, Klaus Hofmann "Deadlock-free generic routing algorithms for 3-dimensional Networks-on-Chip with reduced vertical link density topologies" Journal of Systems Architecture, Elsevier, Vol. 59, (2013), pp (528–542).

37. Haytham Elmiligi, Fayez Gebali, M. Watheq El-Kharashi "Power-Aware Mapping for 3D-NoC Designs using GeneticAlgorithms" 2014 International Workshop on the Design andPerformance of Networks on Chip (DPNoC), Procedia Computer Science, Elsevier Vol. 34 (2014), pp(538 – 543).

38. Hiroaki Morino, Thai Thach Bao, Nguyen Hoaison, Hitoshi Aida Tadao Saito "A Scalable Multistage Packet Switch for Terabit IP Router Based on Deflection Routing and Shortest Path Routing" © 2002 IEEE, IEEE Xplorer, pp (2179-2185)

39. HyungGye Lee and Naehyuck Chang, Umit Y. Ogras and RaduMarculescu "On-Chip Communication Architecture Exploration: A Quantitative Evaluation of Point-to-Point, Bus, and Network-on-Chip Approaches" ACM Transactions on Design Automation of Electronic Systems, Vol. 12, No. 3, August 2007, pp (1-20)

40. I. Cho, C.-C. Shen, Y. Tachwali, C.-J. Hsu, S. Bhattacharyya, Configurable, resource-optimized fft architecture for ofdm communication, in: Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 2746–2750

41. J. Kim, Low-cost router micro architecture for on-chip networks, in: Proceedings of the 42nd Annual IEEE/ACM International Symposium on Micro- architecture, ACM, 2009, pp. 255–266.

42. J. Kwong, M. Goel, a high performance split-radix fft with constant geometry architecture, in: Proceedings of the Design, Automation Test in Europe Conference Exhibition, 2012, pp. 1537–1542,

43. J. S. Kim, M. B. Taylor, J. Miller, D. Wentzlaff, Energy

characterization of a tiled architecture processor with on-chip networks, in: Proceedings of the 2003 International Symposium on Low Power Electronics and Design, IEEE, 2003, pp. 424–427

44. J. Soumya, S. Chattopadhyay, Application-specific network-on-chip synthesis with flexible router placement, J. Syst. Architect., 59 (7) (2013) 361–371.

45. Jason Cong, Yuhui Huang, and Bo Yuan "A Tree-Based Topology Synthesis for On-Chip Network" Computer Science Department University of California, Los Angeles Los Angeles, USA, IEEE Conference proceedings, (2011) pp (650-658)

46. Jia Huang, Christian Buckl, Andreas Raabe, Alois Knoll, Energy-aware task allocation for Network-on-Chip based heterogeneous multiprocessor systems, 19th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2011, 447–454.

47. Jingcao Hu, Radu Marculescu, Energy-aware communication and task scheduling for network-on-chip architectures under real-time constraints, Design, Automation and Test in Europe Conference and Exhibition, 1, 2004, 234–239.

48. Jingcao Hu, Radu Marculescu, Energy-aware mapping for tile-based NoC architectures under performance constraints, 2003, 233–239.

49. Junxiu Liu, Jim Harkin, Yuhua Li, Liam Maguire "Low cost fault-tolerant routing algorithm for Networks-on-Chip" Microprocessors and Microsystems Elsevier, Vol. 39 (2015) pp (358–372)

50. K. Hamwi, O. Hammami, Design and implementation of mpsoc single chip with butterfly network, in: Proceedings of the 18th IEEE/IFIP International Conferenceon VLSI and System-on-Chip, VLSI-SoC, 2010, pp. 143–148.

51. KamelMessaoudi, HichemMayache, AtefBenhaoues, El-Bay Bourennane , Salah Toumi "Connection of H.264/AVC hardware IPs using a specific Networks-on-Chip" Microprocessors

and Microsystems Journal Elsevier Vol. 39 (2015) 609–620.

52. Ke Pang, Virginie Fresse b, Suying Yao and Otavio Alcantara De Lima Jr., "Task mapping and mesh topology exploration for an FPGA-based network on chip", Microprocessors and Microsystems 39 (2015), page no. 189–199,Elsevier.

53. L. Benini, G. De Micheli, Networks on chips: a new SoC paradigm, Computer 35 (1) (2002) 70–78.

54. L. Chen, D. Zhu, M. Pedram, T. M. Pinkston, Power punch: towards non- blocking power-gating of NoC routers, in: Proceedings of the IEEE 21st International Symposiumon High Performance Computer Architecture, IEEE, 2015, pp. 378–389.

55. L. Song, Y. Jiang, Fast parallel fft on ctaiji: A coarse-grained reconfigurable computation platform, in: Parallel and Distributed Processing and Applications, Springer, 2005, pp. 188–195.

56. Leandro Möller, Leandro Soares Indrusiak, Luciano Ost, Fernando Moraes, Manfred Glesner, Comparative analysis of dynamic task mapping heuristics in heterogeneous NoC-based MPSoCs, International Symposium on System on Chip (SoC), 2012, 1–4.

57. Liyang Zhou, Ming-e Jing, Xiaoyang Zeng, Task-binding Based branch-andbound algorithm for NoC mapping, IEEE International Symposium on Circuits and Systems (ISCAS), May 2012, 648–651.

58. Loghi, Mirko, Massimo Poncino, Luca Benini. Cycle-accurate power analysis for multiprocessor systems-on-a-chip, Proceedings of the 14th ACM Great Lakes symposium on VLSI, ACM, 2004.

59. M. Kistler, M. Perrone, F. Petrini, Cell multiprocessor communication network: built for speed, IEEE Micro 26 (3) (2006) 10–23.

60. M. R. Casu, M. K. Yadav, M. Zamboni, Power-gating technique for network-on- chip buffers, Electron. Lett. 49(23) (2013) 1438–1440.

61. Majid Rezazadeh, FarshadSafaei, MahsaMoazez "Flattening: An efficient approach to improving the performance of conventional MINs" Microelectronics Journal, Elsevier, Vol. 46, (2015), pp 875–892.

62. Mike Santarini "FPGA Command Centre Stage in Next Gen Wired Networks" XCell Journal Today, Vol.1 issue 67, (2009), pp (10-14).

63. Mohammad Abdullah Al Faruque, Rudolf Krist, Jörg Henkel, ADAM: run-time agent-based distributed application mapping for on-chip communication, Proceedings of the 45th Annual Design Automation Conference, 2008, 760– 765.

64. N. Prasad, I. Chakrabarti, S. Chattopadhyay, Noc based multiplier -less constant geometry fft architecture, in: Proceedings of the Fourth International Conference of Emerging Applications of Information Technology (EAIT), 2014, pp. 173–178,

65. N.Prasad, Santanu Chattopadhyay, IndrajitChakrabarti "Reconfigurable data parallel constant geometry fast Fourier transform architectures on Network-on-Chip" Microprocessors and Microsystems Elsevier, Vol. 39, (2015) pp (741–751).

66. P. Boopal, M. Garrido, O. Gustafsson, A reconfigurable fft architecture for variable-length and multi-streaming ofdm standards, in: Proceedings of the International Symposium on Circuits and Systems (ISCAS), IEEE, 2013, pp.2066–2070,

67. P. Meher, J. Valls, T. - B. Juang, K. Sridharan, K. Maharatna, 50 years of cordic: algorithms, architectures, and applications, Circ. Syst. I: Regul. Pap., IEEE Trans. 56 (9) (2009) 1893–1907

68. Pradip Kumar Sahu, Santanu Chattopadhyay "A survey on application mapping strategies for Network-on-Chip design" Journal of Systems Architecture, Vol 59 (2013) pp (60–76).

69. R. Airoldi, F. Garzia, J. Nurmi, Fft algorithms evaluation on a homogeneous multi-processor system-on-chip, in: Proceedings of the 39th International Conference on Parallel Processing Workshops (ICPPW), 2010, pp. 58–64,

70. R. Chen, N. Park, V. Prasanna, High throughput energy efficient parallel fft architecture on fpgas, in: Proceedings of the 2013 IEEE High Performance Extreme Computing Conference (HPEC), 2013, pp.1–6,

71. R. Das, O. Mutlu, T. Moscibroda, C. R. Das, Application-aware prioritization mechanisms for on-chip networks, in: Proceedings of the 42nd Annual IEEE/ ACM International Symposium on Micro architecture, ACM, 2009, pp. 280–291.

72. RaduMarculescu, Umit Y. Ogras, Li-ShiuanPeh, Natalie Enright Jerger, and YatinHoskote, "Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 28, No. 1, 2009, pp(3-22).

73. Ran Manevich, Israel Cidon and Avinoam Kolodny, "Design and dynamic management of hierarchical NoCs", Microprocessors and Microsystems (2015), Elsevier.

74. S. Kaushik, A.K. Singh, T. Srikanthan, Computation and communication aware run-time mapping for NoC-based MPSoC platforms, SOC Conference (SOCC), 2011 IEEE International, 26–28 September 2011, 185–190.

75. S. Ranganathan, R. Krishnan, H. Sriharsha, Efficient hardware implementation of scalable fft using configurableradix - 4/2, in: Proceedings of the 2nd International Conference on Devices, Circuits and Systems (ICDCS), 2014, pp. 1–5

76. S. Saponara, M. Rovini, L. Fanucci, A. Karachalios, G. Lentaris, D. Reisis, Design and comparison of fftvlsi architectures for soc telecom applications with different flexibility, speed and complexity trade-offs, Circ., Syst. Signal Process .31 (2) (2012) 627–649,

77. S. Subha, A scheduling algorithm for network on chip, International Conference on Advances in Computing, Control, and Telecommunication Technologies, (2009) 289–291.

78. S. Tosun, O. Ozturk, M. Ozen, An ILP formulation for

application mapping onto Network-on-Chips, International Conference on Application of Information and Communication Technologies (AICT), 2009, 1–5.

79. S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, N. Borkar, An 80-tile 1.28tflops network-on-chip in 65 nm cmos, Solid-State Circuits Conference, 2007, ISSCC 2007, Digest of Technical Papers, IEEE International, (2007) 98– 589.

80. S.Ranganathan, R. Krishnan, H. Sriharsha, Efficient hardware implementation of 1024 point radix-4 fft, Advances in Signal Processing and Intelligent Recognition Systems, Advances in Intelligent Systems and Computing, 264, Springer International Publishing, 2014, pp. 203–211

81. Suleyman Tosun, "Cluster-based application mapping method for Network-on-Chip", Advances in Engineering Software, 42 (2011) 868–874, Elsevier.

82. T. - Y. Sung, H. - C. Hsin, Y. - P. Cheng, Low - power and high - speed cordic based split -radix fft processor for OFDM systems, Digit. Signal Process., 20 (2) (2010) 511–527

83. T. C. Xu, T. Pahikkala, P. Liljeberg, J. Plosila, H. Tenhunen, Optimized multi core architectures for data parallel fast fourier transform, in: Proceedings of the 14th In-ternational Conference on Computer Systems and Technologies, in: Comp Sys Tech'13, ACM, NewYork, NY, USA, 2013, pp. 75–82,

84. T. Murata, Petri nets: properties, analysis and applications, Proc. IEEE 77 (4) (1989) 541–580.

85. T.T. Ye, L. Benini, G. De Micheli, Analysis of power consumption on switch fabrics in network routers, Design Automation Conference, 2002, Proceedings, 39th, 2002, 524–529.

86. Tahir Maqsood , Sabeen Ali, Saif U.R. Malik, Sajjad A. Madani "Dynamic task mapping for Network-on-Chip based systems" Journal of Systems Architecture, Elsevier Vol. 61, 2015,

pp(293-306)

87. Tassadaq Hussain "HMMC: A memory controller for heterogeneous Multi-core System" Microprocessors and Microsystems, Elsevier, Vol. 39 (2015) pp (752–766)

88. TeijoLehtonen, PasiLiljeberg, and JuhaPlosila "Online Reconfigurable Self Timed Links for Fault Tolerant NoC" Hindawi Publishing Corporation VLSI Design (2007), pp (1-13).

89. Umit Y. Ogras, Paul Bogdan, and RaduMarculescu "An Analytical Approach for Network-on-Chip Performance Analysis" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 29, No. 12, 2010, pp(2001-2014).

90. Vasilis F. Pavlidis, Eby G. Friedman "3D Topologies for Networks-on-Chip" IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 15, No. 10, October 2007 pp (1081-1091)

91. Vincenzo Rana, David Atienza, Marco Domenico Santambrogio, Donatella Sciuto, Giovanni De Micheli, A reconfigurable Network-on-Chip architecture for optimal multi-processor SoC communication, 16th IFIP/IEEE International Conference on Very Large Scale Integration, Rhodes, Greece, October 13–15, 2008.

92. W. Quan, A.D. Pimentel, A system-level simulation framework for evaluating task migration in MPSoCs. Compilers, Architecture and Synthesis for Embedded Systems (CASES), 2014 International Conference on IEEE, 2014, October, 1–9.

93. W.J. Dally, B. Towles, Route packets, not wires: on-chip interconnection networks, Design Automation Conference, 2001, Proceedings, (2001) 684–689.

94. Wenbiao Zhou, Yan Zhang, Zhigang Mao, An application specific NoC mapping for optimized delay, International Conference on Design and Test of Integrated Systems in Nanoscale Technology, 2006, 184–188.

95. Wen-Chung Tsai,Ying-CherngLan,Yu-Hen Hu, and Sao-Jie Chen "Networks on Chips: Structure and Design Methodologies" Hindawi Publishing Corporation., Journal of Electrical and Computer Engineering (2012), pp (1-13).

96. Y. Hoskote, S. Vangal, A. Singh, N. Borkar, S. Borkar, A 5-GHz mesh inter-connect for a tera flops processor,IEEE Micro 27(5)(2007) 51–61,

97. Y. Wang, H. Liu, D. Liu, Z. Qin, Z. Shao, E.H.M. Sha, Overhead-aware energy optimization for real-time streaming applications on multiprocessor systemon- chip, ACM Trans. Des. Automat. Electron. Syst. (TODAES) 16 (2) (2011) 14.

98. Yang Quansheng, Wu Zhekai "An Improved mesh topology and Its Routing Algorithm for NoC" IEEE Xplorer, 2010, pp(1-4)

99. Yung-Chang Chang, Ching-Te Chiu "A Study of NoC Topologies and Switching Arbitration Mechanisms" 14th International Conference on High Performance Computing and Communications, IEEE Xplorer, 2012, pp (1643-1647).