International Journal of Advanced Research Methodology in Engineering & Technology, Volume 1, Issue1, December 2016

Comparison of Different Routing Strategies in Network On Chip (NOC)

Savieo sebastian[1], Harikrishna Sabu[2], Midhunlal Sobhanan[3], Nigil.M.Kurian[4] and Dr.Sanju V[5]

1. Muthoot Institute of Technology and science (MITS) Varikoli P.O, Puthencruz 682308. .
Email: savieoseb@gmail.com

2. Muthoot Institute of Technology and Science (MITS) Varikoli P.O, Puthencruz 682308.
Email: harigopikrishnan@gmail.com

3. Muthoot Institute of Technology and Science (MITS) Varikoli P.O, Puthencruz-682308.
Email: midhunlalsobhanan8@gmail.com

4. Muthoot Institute of Technology and Science (MITS) Varikoli P.O, Puthencruz-682308.
Email: annetsnigil@gmail.com

5. Head of the Department, Computer Science and Engineering. Muthoot Institute of Technology and Science(MITS) Varikoli P.O, Puthencruz-682308
Email: sanjuv21@gmail.com

ABSTRACT

The performance requirement of modern devices and systems are growing exponentially. Therefore, the number of components and Hardware requirements are also going high. Unfortunately, the existing methodology and technological advancement is not adequate to provide enough support. Therefore, the need for a much faster and efficient design method should be employed. Hence we introduce the concept of Network on Chip (NOC). By which we can overcome the disadvantage of existing Bus architecture. In bus architecture as we already know data transfer between different parts of the system occur via buses. This mode of communication is not much efficient and has slow bus response and also scalability problem. On the other hand, Network on Chip architecture increases the efficiency of communication between different modules. Its significantly reduces the amount of wire required to route data between modules and has high operating frequency and increased scalability. The objective of this paper is to introduce some concepts of Network on Chip (NOC) architecture involving the comparison of different routing strategies in Network on Chip.

Key Words: Routing Algorithm, Store and Forward, Virtual Cut Through, Worm Hole, X-Y Routing.

1. INTRODUCTION

The main objective of this paper is to bring some clarity on to the topic Network On Chip(NOC). Network On Chip (NoC) essentially means using the network designs used in traditional communication networks and incorporating them on chips. Shortly we can define it as the connection map between PEs (Processing Element). It is mainly adopted from large scale networks and parallel computing. A topology is said to be good if it satisfies the requirements of the traffic at reasonable costs. This essentially means that all the problems faced in the traditional communication networks has to be

addressed by the designers while incorporating the networks on the chip. Network On Chip has been proposed as a highly structured and scalable solution to handle communication problems in System on Chip (SOC). It is accomplished by designing a simulator to implement different routing strategies on an efficient Network On Chip (NOC) architecture. We use a mesh network for this purpose since the implementation and various other characteristic measurements are easier. The main purpose of the simulator is to simulate a mesh network and are measured. Routing is the process of sending a particular message from one part of a node to another node in a network. In order to accomplish this a dedicated path is established first and corresponding routing algorithm is applied in order to make sure that the message delivers at the correct destination. There are various routing strategies by which packets are routed from the source node to destination node. Different performance characteristics like latency, buffer size, node delay of various routing strategies are measured. This project is to develop a simulator to measure the performance of 2D mesh topology using Store and Forward, Virtual Cut Through and Wormhole Routing strategies.

2. INTERCONNECTING PATTERNS

Interconnecting pattern refers to the type of pattern used to interconnect the various components with in the network. There are various types of interconnecting pattern used and they involve mesh, torus, ricobit, star and Butterfly. The architecture which we are mainly focusing here is Mesh architecture. In this pattern the interconnection is of the form of a matrix with m rows and n columns. Two dimensional mesh topology will be used throughout this. The routers are situated in the intersections of rows and columns. The addresses of routers and resources are easily defined as X-Y coordinates in mesh.

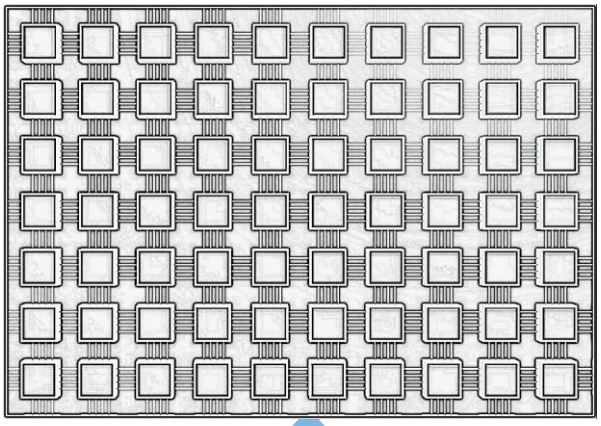

Figure 1: Node Interconnecting Pattern

3. NODE ORGANISATION

A node can be defined as a connection point or an end point which is the basic building block of a network. In a mesh topology, every node is connected to every other node directly and hence it can handle high volume traffic. Another advantage is that even if one connection fails, the other remains intact. In this section we discuss about the internal working of a node. A node actually works in two modes unidirectional and bidirectional mode. In unidirectional mode, the node can receive or send packet at any instant of time but it can send or receive a single control signals at a time. Where as in bidirectional mode the communication between nodes can takes place in both directions.

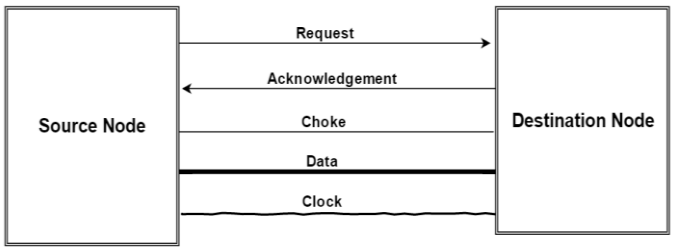

Figure 2: Node Organisation.

The various control signals involved in the working of a node is as follows:

- Clock: This signal works in synchronous with the system clock and used while transferring packets.
- Request: This signal is used to denote that a particular node wants to send some data to some other node in the network.
- Data: This signal is made high when a particular data is being send from one node to another node.
- Acknowledgement: Whenever a node receives a request from another node and if the receive buffer is not full it will send back an acknowledgment signal back to the node.
- Choke: This signal is used to ensure efficient transfer of packet between nodes. Whenever a particular node wants to send a data it checks for the choke bit, it will only send the packet if the choke bit is low. The choke bit is made high if the receive buffer becomes full or any other error occur in the system.

The mesh node has four interfaces which are used to communicate with the neighboring nodes using the X-Y routing algorithm. The class definition below illustrates mesh node.

```
class mesh_node

    {
        Interface Left_Interface;
        Interface Right_Interface;
        Interface Top_Interface;
        Interface Bottom_Interface;


    }
```

4. PACKET STRUCTURE

Packet is the fundamental unit of information in a network. Information is send in the form of a packet from the source node to destination node.

A packet may contain the following details:

- Source Id and Destination Id: identify the end points of the connection.

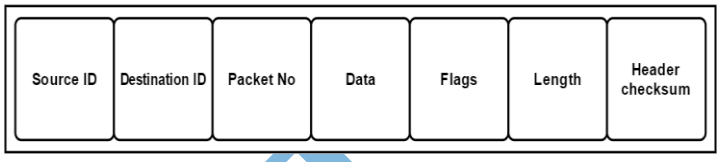

Figure 3: Packet Structure.

- **Packet Id**: Each packet is given an id, in order to identify the packet. Packets are ordered according to their packet id.
- **Data**: Data refers to the particular information which is being send.
- **Flags**: contains the various flags such as:
- **ACK**-indicates that acknowledgement number is valid.
- **RST**-Reset the connection.
- **SYN**-Synchronizes sequence numbers to initiate a connection.
- **FIN**-Means that the sender of the flag has finished sending data.
- **Header Checksum**: This field is used to keep checksum value of entire header which is then used to check if the packet is received error-free.

A packet generator is used to generate packet in to the network according to the required information. The packet generates packets under various traffic patterns and injunction rate. Before transmission, each packet is decomposed into multiple flits with fixed size. Each node has one packet generator including packet control module and configure module.

5. PERFORMANCE METRICS

Some of the most important performance characteristics that are used in evaluating the performance of NOCs are as follows:

5.1 Latency

Network latency can be defined as the time required to transfer n bytes of data from its source to its destination. Latency consists of routing delay, contention delay, channel occupancy and overhead.

5.2 Bandwidth

Communication bandwidth is the amount of data that can be moved using a communication link in a unit time period.

5.3 Throughput

Throughput can be defined as how many units of information a system can process in a given amount of time.

6. ROUTING STRATEGIES

Routing strategies can be explained as the different strategies used to send packets of data from one module to some other module. There are various types of routing strategies but not all of them are suited for the particular scenario. Thus we have selected the below mention three routing strategies.

6.1 Store and Forward

This is the simplest type of routing strategy. In this mode, the entire message is divided into fixed length packets and each node contains an input and output buffers for one entire packet. The packets are stored in the buffer memory corresponding to each node. Every packet is routed individually from the source to the destination. One operation of the store and forward switching is called hop. It includes transferring of one whole packet from one output buffer to the next input buffer. Routing decisions are only made after the whole packet is completely received in its input buffer. This mode is advantageous, when the message is very short in size and frequent since only one channel will be made busy at a time. The main disadvantage of this mode is the requirement of a large buffer memory to buffer the whole packet, this makes the design expensive and also leads to high latency. The latency is directly proportional to the product of packet size and distance; thus it is mainly used for shortest path routing. If the size of a packet is L and the average bandwidth of the network is W and the total number of hops is D, then it can be shown that the communication latency will be (D+1) *(L/W), which is proportional to the number of hops.

6.2 Virtual Cut Through

This is an improved version of store and forward method in which instead of waiting for the entire whole packets to be received, the incoming header packet is cut through in to the next router as soon as the routing decision was made. Hence the need for a large buffer memory is avoided and reduced latency.

6.3 Worm Hole Routing

In this mode the entire packet is divided into a number of flits (flow control digits) and transmitted. The path of the transmission is determined by the path of the first flit called the header flit and the rest the flit follows the same path The main advantage of this technique than the others is that routers do not have buffers for the whole packets instead every routers has as mall buffer for storing one or few flits which comparatively very small in size. It comes under the flow-control methods and is called Flit Buffer Flow control. This mode of transmission is much efficient than the other two due to less network latency and less buffer requirement.
If the size of a flit is F, then the latency for wormhole routing is (L/W) + (F/W) *D.

Various other routing strategies include mad-postman routing, Deflection routing and Chaotic routing. They are mainly used in low level routing ,in this paper we mainly focuses on the above mentioned three techniques.

7. ISSUES IN ROUTING

The various issues faced while sending information from one node to another node are discussed below:

7.1 Deadlock

Deadlock arises in a network when cyclic buffer dependencies occur from the topology and routing algorithm of the network. As a particular buffer memory fills, the packets begin waiting for the corresponding resource. If a cycle of waiting packets develops, then no further progress will be made and the packets will be in a waiting stage forever. Various deadlock avoidance techniques are used to establish a dead-lock free system.

7.2 Live-lock

Livelock occur when a packet is continually routed from one node to another without reaching its destination. Livelock mainly occurs when a non-minimal routing algorithms are used. Different livelock-avoidance techniques involving misrouting randomly are used to avoid this.

7.3 Starvation

Starvation occurs when some packets never reaches its destination. In some routing algorithm each packet is given priority and packets are transmitted according to their priority. Thus when a high priority packet arrives low priority packets have to be in a waiting stage.

7.4 Packet loss

This issue occurs when one or more packet does not reach the respective destination due to the errors introduced in the network, it may include lack of buffer size, contention etc.

8. SIMULATION RESULTS

Among the commonly used switching techniques such as Store and Forward, Virtual Cut Through and Worm-hole Routing, wormhole switching seems to be the most promising one for typical NoC applications due to limited buffering resources and stringent latency requirements. Performance based on various switching techniques are listed below.

The three different cases listed in table (1,2,3) shows the corresponding execution time and packet latency for three different routing strategies including Store and Forward, Virtual Cut Through and Worm Hole routing.

9. CONCLUSION

The implementation of the above simulator has helped us to understand the various characteristics and working of the various routing strategies including Store and Forward, Virtual Cut Through and Worm Hole Routing. The main objective is to have a thorough study of these routing techniques and to develop new algorithms for the development of Network On Chip(NOC) based systems.

The results from the simulator indicates that although worm hole routing suffers from dead-lock it is much simple, faster and much efficient compared to the other two strategies. Since most Network-On-Chip systems need more buffering space and has a high latency requirement, the wormhole switching method the most suitable switching method for reliable transfer of data across a network.

10. FUTURE WORK

The ultimate goal of this project is to develop a NoC simulator system. The work conducted in this thesis is the first part of the whole project. Future work includes the extension of the NoC simulation system to support more network topologies and the implementation of the simulator to multi core processors.

11. ACKNOWLEDMENT

This project is the result of my hard work wherein I have been helped and supported by several persons and institutions directly and indirectly. Now it is the time to acknowledge their contributions. First of all,

to the Great Almighty, the author of knowledge and wisdom for his countless love. I would like to sincerely thank our guide and Head of the Department Dr. Sanju V., for his support and valuable guidance. His timely advice, meticulous scrutiny, scholarly and scientific approach has helped me complete this project on time.

We would like to express our sincere gratitude to our principal, Dr S. Pushpakumar and the management of Muthoot Institute of Technology and Science for their support and to all those people who helped us complete throughout the entire project. Last but not the least we are grateful to all our friends and parents for their valuable motivation and support.

Table 1: First Case

Performance Comparison						
Network Size	Routing Stratergy	Message	Source id	Destination id	Execution Time	Packet Latency
50 ,50	Store And Forward	This is the first test run	[1] ,[1]	[45] , [45]	36320 millisec	6320 millisec
50 ,50	Virtual Cut Through	This is the first test run	[1] ,[1]	[45] , [45]	15320 millisec	2000 millisec
50,50	Worm Hole Routing	This is the first test run	[1] , [1]	[45] , [45]	4000 millisec	500 millisec

Table 2: Second Case

Performance Comparison						
Network Size	Routing Stratergy	Message	Source id	Destination id	Execution Time	Packet Latency
100,100	Store And Forward	This is the first test run	[1] ,[1]	[65] , [30]	42567 millisec	7987 millisec
100 ,100	Virtual Cut Through	This is the first test run	[1] ,[1]	[65] , [30]	17324 millisec	2321 millisec
100 ,100	Worm Hole Routing	This is the first test run	[1] , [1]	[65] , [30]	8322 millisec	1255 millsec

Table 3: Third Case

Performance Comparison						
Network Size	Routing Stratergy	Message	Source id	Destination id	Execution Time	Packet Latency
100 ,100	Store And Forward	This is the first test run	[1] ,[1]	[90] , [45]	59654 millisec	9565 millisec
100 ,100	Virtual Cut Through	This is the first test run	[1] ,[1]	[90] , [45]	24334 millisec	4233 millisec
100 ,100	Worm Hole Routing	This is the first test run	[1] , [1]	[90] , [45]	7233 millisec	1343 millisec

12.REFERENCES

[1] Route Packets Not Wires: on chip interconnection network, William J. Dally, Brian Towles, DAC 2001.

[2] Network On Chip Routing Algorithms, Ville Rantala, Teijo Lehtonen, Juha Plosila, TUCS Technical Report No 779, August 2006.

[3] Deadlock Free Routing Algorithms for Mesh Topology NoC Systems with Regions, Shashi Kumar, Axel Jantsch, JuhaPekka Soininen, Martti Forsell, Mikael Millberg, Johny berg, Kari Tiensyrja and Ahmed Hemani.

[4] Network on chip: a new paradigm for system on chip design, Nurmi, Proceedings 2005 International Symposium on System on Chip, 1517 November.

[5] Networks on chips: A new SoC paradigm, L. Benini and G. De Micheli, Networks on chips: A new SoC paradigm, Computer, vol. 35, no. 1, 2002, pp. 7078.

[6] Performance Evaluation of mesh with source routing for Packet Loss, Lalit Kishore Arora, et. al. IJ ARCSSET Volume-1, Issue-5, August 2012.

[7] Low-latency virtual-channel routers for on-chip networks, R. Mullins, A. West, and S. Moore, in Proc. Int. Symp. Comput. Architecture, Jun.2004, pp. 188197.

[8] Flit-reservation flow control, L. Peh and W. J. Dally, in Proc. Int. Symp.High-Performance Comput. Architecture, Jan. 2000, pp. 7384.