Design and Implementation of Packet Switched NoC Fabric Using YX Algorithm

Ashwini.L.Kanth[1], Mr. Amaresh S K[2], Dr. Siva S Yellampalli[3]
[1]*UTL Technologies Ltd. Bangalore*
[2]*Asst. Professor, UTL Technologies Ltd. Bangalore*
[3]*Principal, UTL Technologies Ltd. Bangalore*

***Abstract*- Nowadays the idea of using network on chip (NoC) as viable on chip communication fabrics for the future multiprocessor system-on-chip (MPSoCs) and system on chip (SoC). The scalability and parallelism properties made NoC efficient than others. Bus based on-chip communication architectures suffers from scalability and speed,whereas NoC uses layered data transfer. In NoC routing algorithm is responsible for routing data from source to destination. In this paper designed packet switched NoC with XYis proposed in order to avoid the deadlocks in network.**

***Index Terms*- Network On chip (NoC), System on Chip (SoC), First in First OUT (FIFO).**

I. INTRODUCTION

Among all designs communication a major role belongs to integrated components. In system they use traditionally tristate buffers and MUX based interconnection architectures. With advances in CMOS technology i.e., scaling, the design have entered into DSM era. The signal integrity and inadequate performance are resulted due to bus architecture. In addition to it ,DSM effects will create severe signal integrity problems that make error in data communication on buses. The signal integrity problem can be defined as the received signal at the destination is not same from the transmitted signal at the source. This happens because of noises in the signal. Some of the important DSM effects that can cause noise on buses are crosstalk, electromagnetic interference, transmission line effects and soft errors. These draw backs of bus based architecture prevents the system to fulfil the performance requirements.

For SoC, Network on chip is the embedded switching network. Network on chip is the emerging trend that is being developed in recent SoC's. Network on chip provides parallel communication even in intensive communication network and also congestion free routing. In network work on chip switching strategy and routing algorithm is key decision maker for routing.

Different network topologies, switching strategies and routing algorithms can be used to meet requirements of performance, hardware overhead, and power consumption. Based on arbitration and routing algorithm the respective data injected is transferred through the respective direction.

The remaining paper is organised as follows, in section II NoC environment is discussed, section III explains the detailed process of switching techniques [2]. The arbitration[4] techniques and routing algorithm are explained in section IV. The experimental results and conclusion are given in section VI.

II. NOC ENVIRONMENT

As discussed in section I NoC consists of processing elements, crossbar switch, buffers, and arbiter. The performance of NoC depends on switching strategy, Network topology and Routing algorithms. Network topology[3] gives the physical organization of processing elements and nodes. General topology used is direct 3x3 mesh network[1] topology which has direct connection to neighbouring nodes. Switching techniques defines the granularity of data transfer from source to destination with packet switching technique. The routing algorithms provide path to route data from source to destination with static distributed routing algorithm.

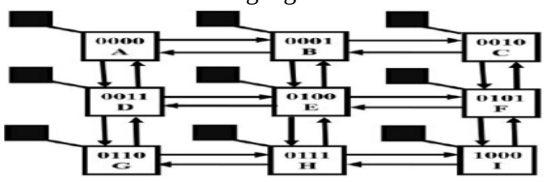

Figure 1: 2-D Mesh NoC

Arbitration schemes are used in NoC to avoid congestion in routing path with static priority arbiter. The simple NoC switch block diagram is as shown in figure 1.2.each switch consists of a crossbar and internal memory blocks. The processing elements generate the message and it is transferred in form of packets which is as shown in figure 2.3, these packets are divided into FLIT's (flow control unit).

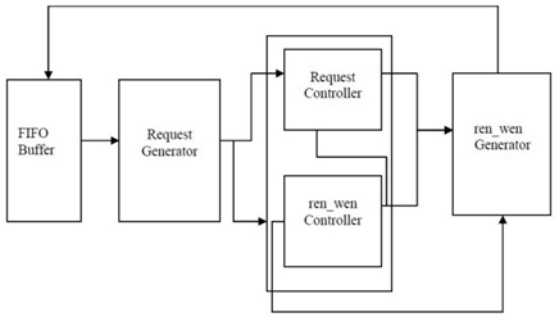

Figure 2: NoC switch block diagram

FLIT's consists of head flit, body flit and tail flit. Head flit ensures the arrival of data, body flit consists of destination address and tail flit gives the acknowledgement, once the data reaches destination. FLIT is classified into physical units known as PHIT. The packet structure consists of four FLITs and each PHIT size is of 10 bit. FIFO buffer size is same as the FLIT size.

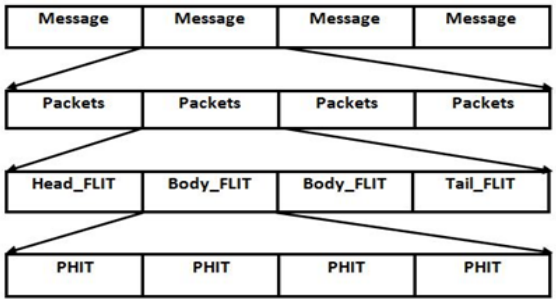

Figure 3: Packet structure

DESTINATION	SOURCE									
		Switch_a	Switch_b	Switch_c	Switch_d	Switch_e	Switch_f	Switch_g	Switch_h	Switch_i
	Switch_a	100 eject	011 west	011 west	000 north	000 north	000 north	000 north	000 north	000 north
	Switch_b	010 east	100 eject	011 west	000 north	000 north	000 north	000 north	000 north	000 north
	Switch_c	010 east	010 east	100 eject	000 north	000 north	000 north	000 north	000 north	000 north
	Switch_d	001 south	001 south	001 south	100 eject	011 west	011 west	000 north	000 north	000 north
	Switch_e	001 south	001 south	001 south	010 east	100 eject	011 west	000 north	000 north	000 north
	Switch_f	001 south	001 south	001 south	010 east	010 east	100 eject	010 east	000 north	000 north
	Switch_g	001 south	001 south	001 south	001 south	001 south	001 south	100 eject	011 west	011 west
	Switch_h	001 south	001 south	001 south	001 south	001 south	001 south	010 east	100 eject	011 west
	Switch_i	001 south	001 south	001 south	001 south	001 south	001 south	010 east	010 east	100 eject

Table 1: Switch Arbitration

The NoC Block diagram is shown in figure 1.2. The working procedure of NoC is as discussed in below steps.

1. Processing elements generates data that to be routed, the data is transferred in form of packets which is given as input to the FIFO buffers.
2. Based on the head flit the address is decoded from the FLIT. By consulting the routing table, next switch where data to be sent is identified. For example the current source is switch A and the destination is switch I, from the routing table shown in table 1 the next switch is west (Switch B).
3. Based on the decoded address, request to o the output link is posted for arbiter. This posted request is used to generate the control signals for crossbar which route input data the respective output link.
4. Once the destination address matches with the current switch address, data is ejected via eject port. Otherwise the process from step 1 for the current switch.

As mentioned in above procedure a NoC Switch must be tested against all possible data transmissions from one switch to the other along with routing paths. The above NoC consists of buffers, switches, Arbiters and links. Which makes the design complex and also NoC uses packet structure to route the data. Testing NoC using randomly generated test patterns will not give assurance of complete data routing coverage. Hence functional test patterns are generated to test the NoC environment; these functional test patterns are designed in such a way that all the possible routing paths are triggered. Functional patterns are designed giving respective destination address at the source location. The routing paths between the source and destination addresses are also covered by loading these functional patterns, ensuring the greater test coverage compared to that of random test patterns.

III SWITCHING TECHNIQUES

The switching strategy in NoC determines the flow of data, that flows in the routers between the network and also the switching strategy explains the granularity of data or message transmission.

The switching strategies are of two types namely:

1. Circuit Switching

2. Packet Switching

1. Circuit Switching

In the circuit switching, physical link between source and destination will be examined and reserved in prior to transfer the data and path includes the links and routers. The message header flit travels through the network i.e., from the source to destination and reserves the links along the path traversing. If the header of message reaches the destination without any conflicts then the links in path are available and it will be reserved for transmission of data along an acknowledgement will be sent to source router. Once after receiving the acknowledgement, the data will be transferred from source to destination through the reserved path. If the link is blocked or used by the neighbouring router, then negative acknowledgement will be sent to source. The reserved path will be held until all the data is transmitted, after transmission of data the path is torn out as a part of tail flit and this is the drawback of this switching technique.

2. Packet Switching

In case of packet switching the path is not reserved in prior as it is done in the circuit switching for transmitting data. The packets are transmitted on their own i.e, based on the load condition on link. In case of circuit switching startup waiting time and the fixed minimal latency in the routers is seen. In Packet switching there is no startup time, but there is variable delay due to traffic in the routers along packet's path. Since there is no path reservation, multiple packets arriving from different sources and tries to access the same router. In order avoid such confusion arbitration scheme is used to access the link.

Store And Forward is a type of simple packet switching technique, where the packet is transmitted from source router to destination router when there is a buffer space in the receiving router. The data from routers will be transmitted only when it is received completely.

IV ARBITRATION

Arbiter is an authority to provide access to a particular resource and it performs by observing number of request signals to use the link in NoC switch. In NoC the inputs are from four directions i.e., North, South, East and West. When all these input links tries access one output link, then it leads to confusion. At that instant NoC requests arbiter to provide priority to the inputs links for sending the data to that particular output link. An arbiter with complex arbitration scheme reduces the complexity of the system. The arbiter function mainly depends upon the arbitration algorithm employed by it. Arbitration algorithms are classified into different types, they are:

1. Static priority algorithm
2. Dynamic priority algorithm
3. Round robin algorithm
4. TDM algorithm

1. Static Priority Algorithm

Static priority algorithm is most commonly used algorithm in NoC's. In this algorithm, each input links will be assigned to a fixed priority. In NoC, inject input will be given priority first and then North, South, East and West. When all these inputs routes the data in one particular direction, at that instant inject input will be given access to send the data in that output link. Once the inject input is routed, later the rest will be given priority i.e., North, South, East and West. Static priority algorithm is simple to implement, almost all the system use this algorithm. The main disadvantage here is data starvation will affect the performance of the system. Here in this project YX routing algorithm is used in order to avoid congestion in network and is as shown in figure 4.

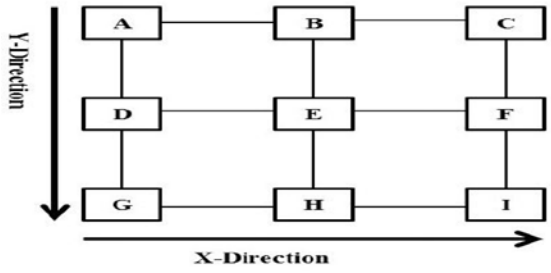

Figure 4: YX algorithm

2. Dynamic priority Algorithm

In dynamic priority algorithm access to the input link will be given based on the load conditions. In NoC access to the input link will be given based on buffer status of the output link, traffic load on the network and which input link requests first to access particular output.

3. Round Robin Algorithm

In round robin algorithm all the inputs will get chance of accessing output links. The main principle of this algorithm is that the priority is given to the input links in round robin fashion i.e., inject input, north input, south input, east input and west input. Hence each input link will get chance to access the particular output link. The main advantage of this algorithm is simple to implement and reduced latency.

4. TDM Algorithm

Linking time of inputs are divided into number of slots in this TDM algorithm. And access output links by providing time slots to input links. Where each slot is reserved for an input link by using timing wheel. Hence in this algorithm many time slots will go to waste. An example is explained show the wastage of time slots is, if inject input is assigned for 7 slots, bit the data receives the data at the output link in 4 slots then the remaining 3 slots will be waste.

V RESULTS AND SIMULATIONS

1. FIFO Buffer

The simulation results of FIFO buffer is as in figure 5. The FIFO buffer is used to store the data from IP address and transfer it to its respective destination IP address.

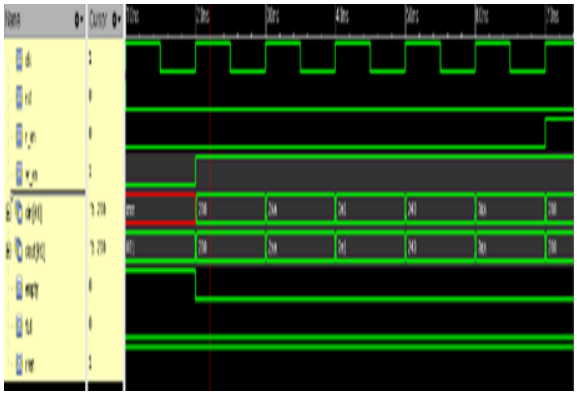

Figure 5: Simulation of FIFO buffer

2. Request Generator

The simulation result of request generator is as shown in figure 6. The input of request is the output of FIFO and this requested signal is generated by using YX algorithm. If head flit is high in the data then the generator sends the signal to which direction it as to pass.

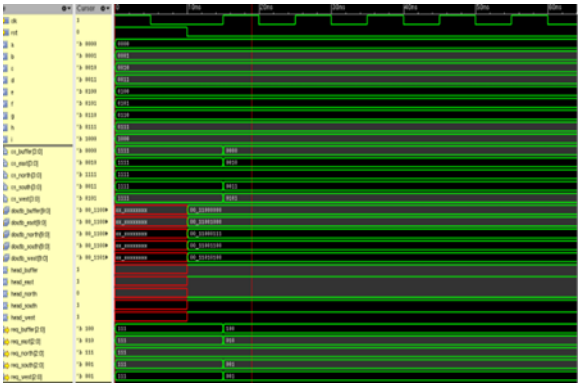

Figure 6: Simulation result of request generator

3. Crossbar Switch

The figure 7 shows the simulation results of crossbar switch which consists of mux control signals and generate flag signals. Based on request signal and full condition data is transferred from input to destination address.

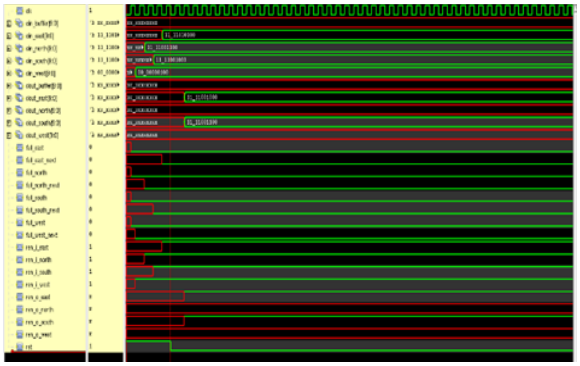

Figure 7: Simulation of Crossbar Switch

4. Read-write generator

The figure 8 is the simulation result of read-write generator. Based on request signals and mux control signals the read-write generator enables the signals. The input given to FIFO buffer are ren and wen signals.

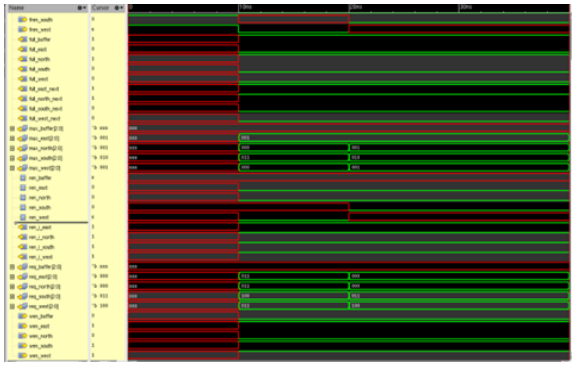

Figure 8: Simulation result of Read-write generator

5. Single switch architecture

The simulation and synthesis results are as shown in figure 9 and figure 10 respectively.

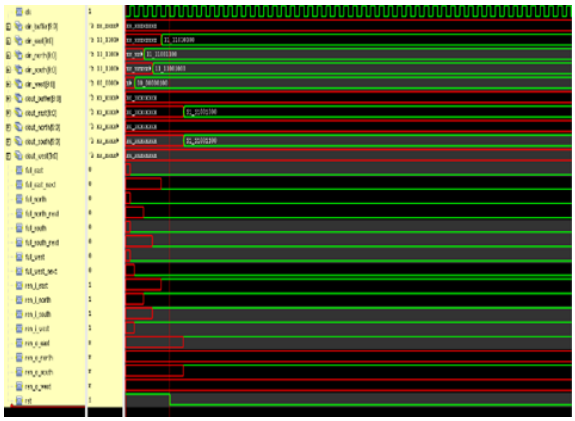

Figure 9: Simulation of Switch Architecture

Figure 10: Synthesis of switch architecture

VI CONCLUSION

NoC is an embedded communication architecture used in SoC. Each processing element in SoC is embedded with a switch. These switches consists of FIFO buffer to store and send the data, request generator, read and write enable generator, routing algorithm and crossbar switch. The switch architecture is designed by using verilog code and cadence encounter and simulation results are shown above.

REFERENCE

[1] Priya D.S, Ravikiran.N, "NoC Self –Test" IEEE International conference on computational intelligence and computing researchICIC-2015, at vikram college of engineering,Madurai,2015

[2] Kurt shuler, "Prioritize Fault tolerance in Design for automotive, Industrial, Military and Medical SoCs", Arteris, December 2014.

[3] JoseFlich, DavidBertozzi, "Designing NoC Architecture " In NOC ,Taylor And Francis. Ed., 2nded.USA:CRC group,2011,pp.33-66.

[4] phi-hung pham,jongsun park,phoung mau "Design And Implementation of Backtracking Wave-pipeline switch to support guaranteed throughput in Network-On-Chip" VLSI,IEEE , vol.20,No.2,feb 2012.

[5] Radu Marculescu, Umit Y.Kgras, Natalie Enright Jerger" Outstanding Research Problems in NoC Design: system, microarchitecture, and Circuit perspectives" Computer- Aided Design of Integrated Circuits and Systems, IEEE, vol.28, No.1, Jan 2009.

[6] Xingang ju, Liang yang "Design and Implementation Of 2*4 2D Torus Topology "Computer-Aided design Of Integrated circuits and systems, IEEE, vol.27, No.4, August 2011.

[7] Partha Pratim pande , RyanGarykim Wang Choi,Zhuo chen,Diana Marculeseu, Radu Marculeseu, "The Power of less wiring: Enabling energy efficiency in many core platforms through wireless NOC" ICCAD 2015 IEEE/ACM International Conference on pp.165-169,2015