Area-Efficient Network-on-Chip Router In Packet Switching

S.Jayashree, K.Kavitha, R.Porselvi

Abstract-Networks-on-chip (NoC) has been widely proposed as the future communication paradigm for use in the next generation System-on-chip (SoC). Conventional analytic models for the performance analysis of Network-on-chip often possess a surplus amount of area and delay constraints, where the number and position of processor elements or faulty blocks vary during run time. Indeed, we propose an efficient router which reduces the number of slices and eliminates the use of arbiter which in turn reduces the area. A detailed comparative analysis with the existing method is performed in terms of area and delay. It is captured in the Verilog Hardware Description Language. Moreover, simulation results have confirmed that the proposed system is the most efficient one regarding with its performance.

Index Terms—NoC- Network on Chip, SoC- System on Chip, CMP- Chip Multi Processor, FIFO-First In First Out,PDA- Path Diversity Aware, PE- Processing Element, NI- Network Interface, R- Router

—————————————————————————(—————————————————————————

1 INTRODUCTION

Network on Chips are critical elements of modern System on Chip (SoC) as well as Chip Multiprocessor (CMP) designs. Network on Chips (NoCs) helps to manage high complexity of designing large chips by decoupling computation from communication. SoCs and CMPs have a multiplicity of communicating entities like programmable processing elements, hardware acceleration engines, memory blocks as well as off-chip interfaces. With power having become a serious design constraint, there is a great need for designing NoC which meets the target communication requirements, while minimizing power using all the tricks available at the architecture.

As the number of architectural blocks that are integrated on a single chip continues to rise, overall system performance and cost become increasingly dependent on the efficiency of the NoC implementation. Although regular topologies are preferred for building NoCs, heterogeneous blocks, fabrication faults and reliability issues derived from the high integration scale may lead to irregular topologies. In this situation, efficient routing becomes a challenge. Although table-based routing allows the use of most routing algorithms on any topology, it does not scale in terms of latency and area.

The importance of low latency lies in the fact that the delay of packet sent from source to destination is greatly reduced, yielding a much more balanced traffic load. Therefore, achieving low latency is an ultimate goal for us in designing a router. The routers can be arranged in an arbitrary topology and are connected with an arbitrary number of modules. Furthermore, a flit (the basic transmit unit in NoCs) can be propagated using different routing, switching and arbitration schemes. On the one hand, this large variety of parameters is the essence for high flexibility. On the other hand, it spans a very large design space.

_____

• S.Jayashree is currently pursuing bachelors degree program in electronics and communication engineering in tagore engineering college under Anna University, India, PH-04427447156. E-mail: jayashree.ece13@gmail.com
• K.Kavith is currently pursuing bachelors degree program in electronics and communication engineering in tagore engineering college under Anna

University, India, PH-9566472236. E-mail: kavi6813@gmail.com

This makes the optimization of the interconnection infrastructure challenging. A high accuracy is essential to acquire reliable information for the design optimization already in an early design phase. The authors of this paper observed that existing analytic models are not able to provide a sufficient accuracy. This is especially the case for NoC routers using the popular round-robin arbitration scheme. This arbitration scheme offers a low complexity and local fairness and is therefore used in many existing network-on-chip designs for handling best-effort traffic.

The arbitration scheme has a strong impact on the whole network throughput, bottlenecks and path latencies. Therefore, it influences design decisions significantly. But Wrong design decisions can lead to over-provisioning, i.e., waste of chip area. On the other hand, performance requirements may not be fulfilled, which is even worse. Thus, it is essential to employ effective models that offer a high accuracy. Therefore, in this paper, we alter a latency model, which simultaneously considers the number of output path and buffer status, to predict the latency condition of the output channels.

Based on this model, we propose valuable packet switching method to overcome congestion problem in NoC. Also it can be further efficiently capitalize on the flexibility and is expected to achieve with a reduction of area. Based on this, many important network performance metrics, such as mean Latencies or network throughput can be derived. Like in any other network, router is the most important component for the design of communication back-bone of a NoC system.

In a packet switched network, the functionality of the router is to forward an incoming packet to the destination resource if it is directly connected to it, or to forward the packet to another router connected to it. It is very important that design of a NoC router should be as simple as possible because implementation cost increases with an increase in the design complexity of a router.

1.1 NoC Architecture

A variety of interconnection schemes are currently in use, including crossbar, buses and NOCs. Of these, later two are dominant in research community. However buses suffer from poor scalability because as the number of processing elements increases, performance degrades dramatically. Hence they are not considered where processing elements are more.
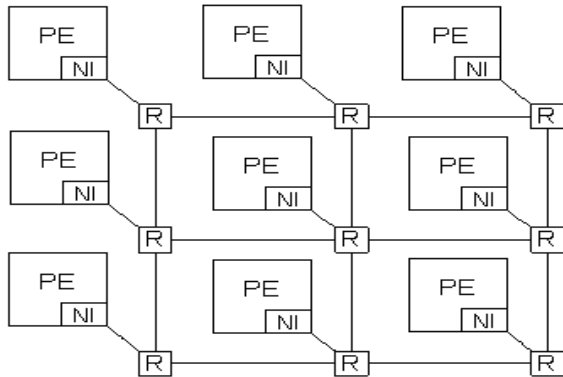

Fig 1.1: NoC Architecture

PE- Processing Element
NI- Network Interface
R- Router

To overcome this limitation attention has shifted to packet-based on-chip communication networks, known as Network-On-Chip.Fig 1.1 shows the architecture of NoC. A typical NoC consists of computational Processing Elements (PEs), Network Interfaces (NIs), and routers. The latter two comprise the communication architecture. The NI is used to packetize data before using the router backbone to traverse the NoC. Each PE is attached to an NI which connects the PE to a local router. When a packet was sent from a source PE to a destination PE, the packet is forwarded hop by hop on the network via the decision made by each router. For each router, the packet is first received and stored at an input buffer. Then the control logics in the router are responsible to make routing decision and channel arbitration. Finally, the granted packet will traverse through a crossbar to the next router, and the process repeats until the packet arrives at its destination.

1.1.1 Processing Element

Different Processing Element can be used for NoC communication. This may include several processors, memories and dedicated hardware. All these Processing Elements can be arranged in different topologies for efficient communication and optimum performance.

1.1.2 Network Interface

The Network Interface is used to packetize data before using the router backbone to transverse the NoC. Each PE is attached to an NI which connects the PE to a local router. The network interface is the essential component of NoC

architecture. The communication between switch and resource is carried out through NI. Proper communication mechanism along with well defined network interface forms the backbone for NoC, with this concrete backbone network can be easily established just by putting resources at the interfaces. Such type of design practice improves the reusability and modularity of the network. Network interface can be implemented at both software and hardware levels.

1.1.3 Router

Router plays an important role in NoC. Router decides the path of packet travel from source to destination. When a packet was sent from a source PE to a destination PE, the packet is forwarded hop by hop on the network via the decision made by each router. For each router, the packet is first received and stored at an input buffer. Then the control logic in the router is responsible to make routing decision and channel arbitration. Finally, the granted packet will traverse through a crossbar to the next router, and the process repeats until the packet arrives at its destination.
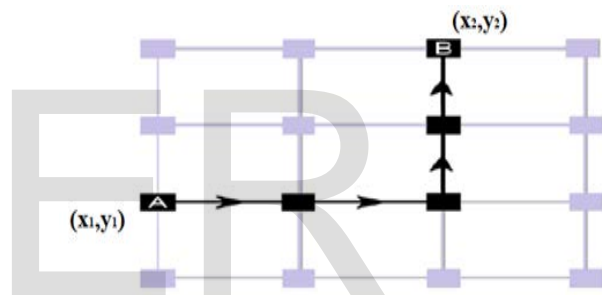
1.2 XY Routing Algorithm

Fig 1.2: XY Routing Algorithm

Fig 1.2 shows theXY routing algorithm. It routes the data first in X direction (Horizontal) to the correct column and then in Y direction (Vertical) to the receiver. The routing process is taken like this way. Then the routing operation is done on the basis of the conditions.

$$X=Y, X>Y, X<Y$$

In this the address of the router are their XY coordinates. It uses 2 dimensional mesh topology. A data is forwarded horizontally till the target column is reached and is then forwarded vertically to the destination router.

2 ROUND ROBIN ARBITRATION

Packetswith the same priority and destined for the same output port are scheduled with a round-robin arbiter which is shown in figure.2. Supposing in a given period of time, there was many input ports request the same output or resource, the arbiter is in charge of processing the priorities among many different request inputs. The arbiter will release the output port which is connected to the crossbar once the last flit has finished the transmission. So that other waiting packets could use the output by the arbitration of the arbiter. A round robin arbiter operates on the principle that the request was just

served should have a lowest priority on the next round of arbitration depending upon the control, logic arbiter generates
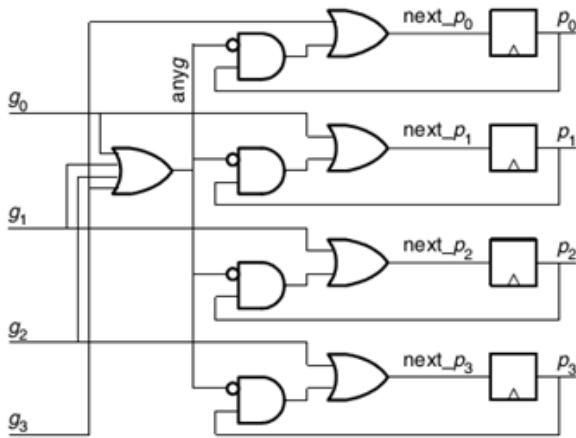

Fig 2:Architecture of Round Robin Arbitration

select lines for multiplexer based crossbar and read or write signal for FIFO buffers.

3 PACKET SWITCHING METHOD

In packet switching the data transfers in the form of packets between cooperating routers and independent routing decision is taken. The store and forward flow mechanism is best because it does not reserve channels and thus does not lead to idle physical channels The arbiter is of rotating priority scheme so that every channel once get chance to transfer its data. In this router both input and output buffering is used so that congestion can be avoided at both sides. A router is a device that forwards data packets across computer networks. Routers perform the data "traffic direction" functions on the Internet.

A router is a microprocessor-controlled device that is connected to two or more data lines from different networks. When a data packet comes in on one of the lines, the router reads the address information in the packet to determine its ultimate destination. Then, using information in its routing table, it directs the packet to the next network on its journey.Data packet moves in to the input channel of one port of router by which it is forwarded to the output channel of other port. Each input channel and output channel has its own decoding logic which increases the performance of the router. Buffers are present at all ports to store the data temporarily.

The buffering method used here is store and forward. Control logic is present to make arbitration decisions. Thus communication is established between input and output ports. According to the destination path of data packet, control bit lines of FSM are set. The movement of data from source to destination is called switching mechanism. The packet switching mechanism is used here, in which the flit size is 8 bits. Thus the packet size varies from 0 bits to 8 bits.

3.1 Proposed Router
In the proposed system packet switching network is used. The

PEs and IPs can be connected directly to any side of a router. Therefore, there is no specific connection port for a PE or IP. Three port networks are used as a router.
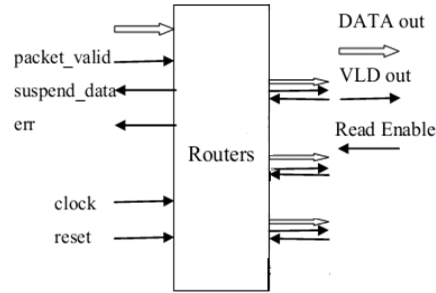

Fig 3.1: Proposed Router

Fig 3.1 shows the structure of proposed router which consists of data in, packet valid, suspend data, clock, reset, error, data out, valid out, and read enable. Based on the packet valid, the data is sent into the router.

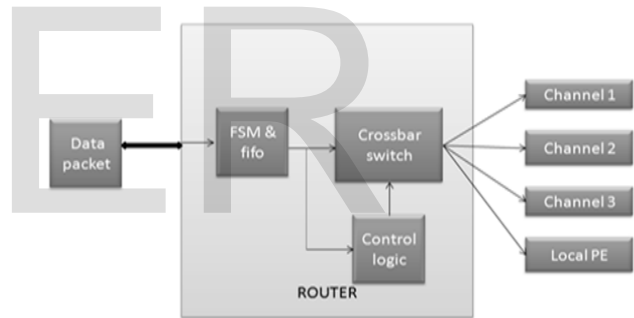### 3.2 Block Diagram of Address Based Packet Switching Method

Fig 3.2: Block Diagram of address based packet switching method

Fig 3.2 shows the block diagram of the proposed address based packet switching method. In this the packet is sent to the FSM and FIFO block. Then the corresponding operations are done. Finally the packets are routed to the respective output channels.

3.3   Router Elements

3.3.1 FIFO
In the FIFO (First In First Out), the inputs are stored and forwarded. So this method is also called as store and forward technique.

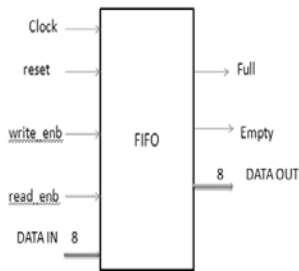Fig 3.3.1: FIFO operations

The FIFO can do 5 different operations
1) Write Operation
2) Read operation
3) Read and Write Operation
4) Full
5) Empty
The functionality of fig 3.3.1 is explained below
Write operation:
The FIFO write operation is done by when the data from input data_in is sampled at rising edge of the clock when input write_enb is high and FIFO is not full. In this condition, FIFO Write operation is done
Read Operation:
The data is read from output data out at rising edge of the clock, when read_enb is high and FIFO is not empty.
Read Write Operation:
Read and Write operation can be done concurrently.
Full:
It indicates that all the locations contained by FIFO have been written.
Empty:
It indicates that all the locations of FIFO are unfilled.
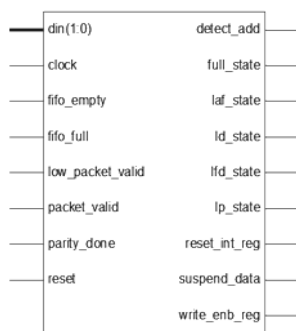
3.3.2 FSM

Fig 3.3.2: FSM Block

Fig 3.3.2 shows the FSM Block. It defines the state. Initially the header is fixed along with the address followed by the data. For example, to route the data for the third output channel, the enable 3 pin is selected, then the data is sent. Packet valid line has its importance during traffic conditions. If the desired channel is busy, FSM makes the packet valid to zero and no data will be sent. Similarly

packets are sent to the desired channel only when the packet valid is high.

3.3.3 Synchronizer
The synchronizer is used to synchronize the inputs with the outputs. It is also used to check whether the correct data is received at the output side. For example, if there is fifth data in the FSM means the synchronizer will check that the FIFO has sent the fifth data.

3.3.4 Error Checking Unit

For error checking we use parity generator. In our proposed system even parity is used. In order to check and correct the error, the sender may inform the receiver which kind of parity is used.
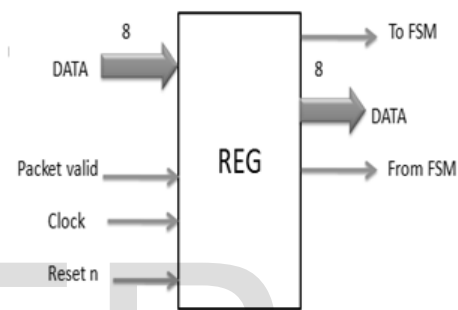

Fig 3.3.4: Error checking block

Fig 3.3.4 shows error checking block. It contains status, data and parity registers required by router. All the registers in this module are latched on rising edge of the clock.

3.4 Proposed Router Packet Format
In the proposed router packet format, it is designed with the help of parity. In every packet, 64 bytes of data is sent. Here error can also be found and can be corrected.
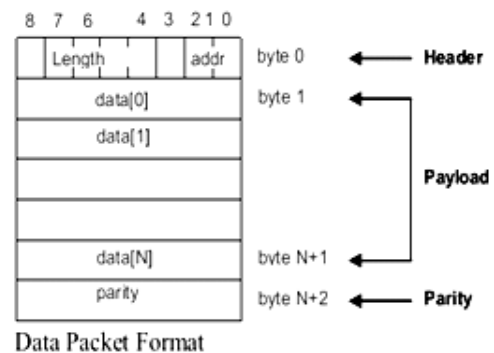

Fig 3.4: Packet format

Fig 3.4 shows the packet format of the proposed system. In this, the last two bits are the address field and the remaining bits are the length of data. This forms the header of the packet format. The payload consists of 64 bytes of data. In this method, parity is separately defined. The length of the data is XORed with the accumulator input and this follows for the

entire payload. Then the whole packet is sent. Based on the parity given, the error can be found.

3.5 Router Input Protocol

Router input protocol defines how the packet has been sent to the channel or a buffer.Fig. 3.5 reveals the significance of err line and suspend line in the router input protocol. The err line is made high in case of any error have been occurred. During the clock input, the data is sent only when the packet_valid line and the reset is enabled. When the same data is sent again, the suspend line is made high which ceases the transmission of data.
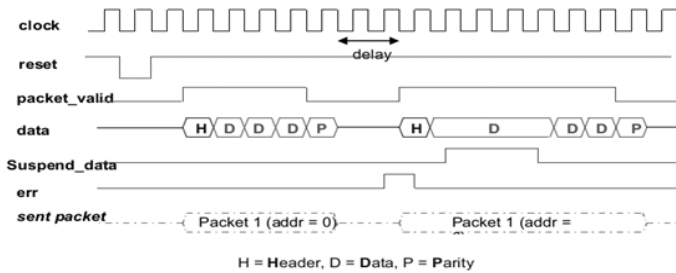

Fig 3.5: Router Input Protocol

At the same time, err line is made high to indicate the transmission of same data.

3.7 Router Output Protocol

The clock is given and the reset is set high. In the output protocol, the data can be received only when the status of read enable line is high irrespective of the status of valid out line. Based on the clock, reset, packet valid, the same packet can be received at the output. The data is sent when the packet valid status is high.
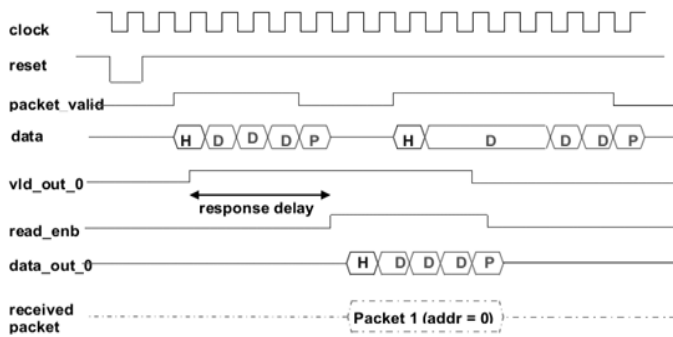

Fig 3.7: Router Output Protocol

Fig 3.7 reveals the router output protocol structure. In this, the same output packet is received which was sent at the input side.

4 SWITCHING TECHNIQUE

Store and forward switching technique is used in the proposed system. In this technique, every packet is individually routed from the source to the destination. One step of the SF switching is called hop. It consists of copying the whole packet from one output buffer to the next input buffer. Routing decisions are made by each intermediate router only after the whole packet was completely buffered in its input buffer.
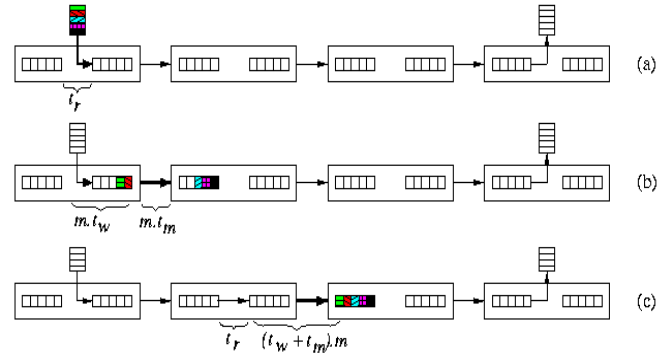

Fig 4: Store and forward mechanism

(a) Routing decision is being made in the first router.
(b) The packet is performing the first hop to the second router after has been copied to the output buffer of the first router.
(c) The whole packet after the second hop.

This technique is advantageous, when messages are short and frequent, since one transmission makes busy at most one channel from the whole path.

4.1 Flow Chart

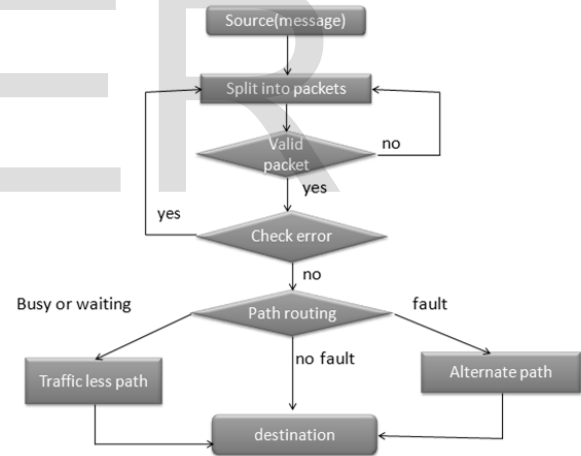Fig 4.1 shows the flow diagram for the transmission of data.

Fig 4.1: Flow chart

The source or the message is first splitted into packets. If the incoming packet is a valid one, then it is further given to the error checking unit where the error has been detected. The data packet containing error will be again splitted into packets and checks whether it is valid. The error free data packet will be routed to the destination considering three different states of affair.

They are as follows.

1) If there is no liability, the packets will be routed to the destination using the shortest path.

2) If any failure occurs making the shortest path unapproachable, the router selects an alternate path in order to route the packets.

3) In case of traffic during the function, router will select a path which is having less traffic.

5 SIMULATED OUTPUTS OF ADDRESS BASED PACKET SWITCHING METHOD

Fig 5 shows the channel output for address based packet switching method.The data will be sent only when the packet valid and reset is made high. Once the channel is selected number of data can be sent irrespective of the last two bits which indicate the address. For changing the output channel or port, first the packet valid is made low followed by enabling the read enable line.
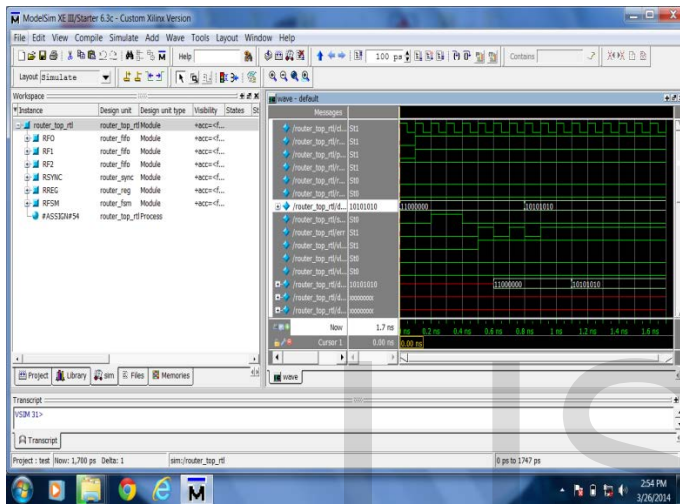

Fig 5: Simulated Output

6 RESULTS AND CONCLUTIONS

6.1 Area Analysis

The emulations of round robin arbiter and packet switching are implemented on FPGA platform.
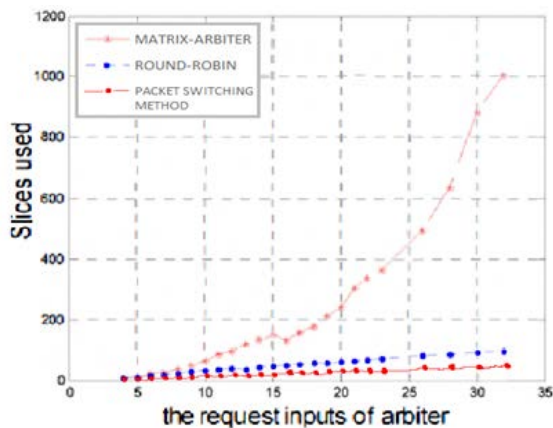

Fig 6.1: Area Analysis

Initially, we set different numbers of request inputs. We get the statistics about the resource utilization, throughput and power consumption of the two different arbitration mechanisms.

Once the packets from the virtual channel of the input simultaneously request the crossbar switch, the number of the request inputs of arbiter increased. But our proposed system consists of packets having 64 bytes to be transmitted for one clock cycle.

Fig. 6.1 shows that the matrix arbiter and Round robin arbiter cost similar resource when there are a few requests; nearly about 100 slices are consumed. When the number of input requests increases, Matrix-arbiter will employ abundant resource compared to the round robin method. The proposed system consumes much less resources than the previous methods. When the request inputs approach 32, the Matrix-arbiter will utilize 1003 slices, the Round-robin arbiter just uses 98 slices while the switching technique uses around 50 slices.
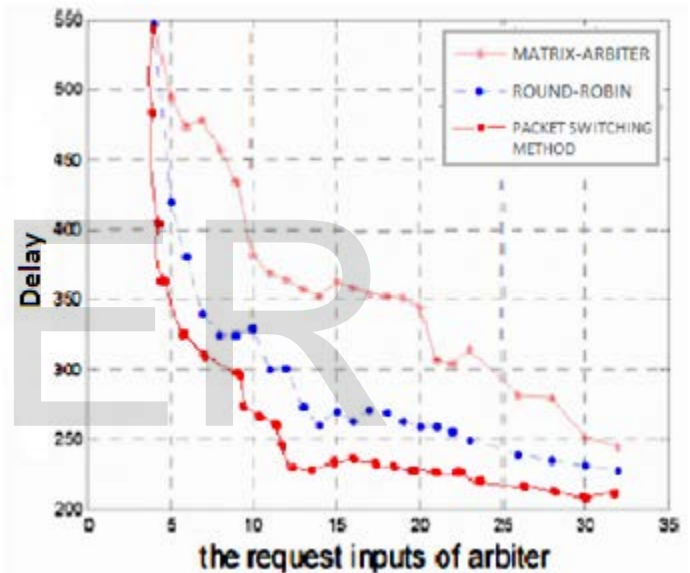
6.2 Delay Analysis

Fig. 6.2: Delay Analysis

Finally, we analyze and compare the delay constraints of two mechanisms. In Fig 6.2, we can see that the delay will increase as the number of inputs is increasing. The graph shows that the address based packet switching method produce less delay than round-robin arbiter. In the design of our proposed scheme, we should make a trade-off among the resource, area, delay and power consumption, and choose suitable mechanism according to that.

7 CONCLUSIONS

The proposed switching technique fulfilled the requirement of implementing a low area and low delay communication path for on-chip networks. Two mechanisms such as round robin arbiter method and address based packet switching method are designed and simulated using Xilinx. Our proposed system is analyzed in terms of area and delay by comparing with the round robin arbitration. The analysis shows that the

arbiter which is designed based on address based packet switching method is having less area and delay compared with the existing round-robin and conventional matrix arbiter method.

REFERENCES

[1] Suyog K.Dahule, Dr. M.A.Gaikwad, "Design and analysis of matrix arbiter for NoC architecture", *International Journal of Advanced Research in Computer Science and Electronics Engineering* Volume 1, Issue 5, July 2012.

[2] Cedric Killian, Camel Tanougast, Fabrice Monteiro, and Abbas Dandache, "Smart reliable network-on-chip", *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 2013.

[3] M. Hosseinabady, M. Kakoee, J. Mathew, and D. Pradhan, "Low latency and energy efficient scalable architecture for massive NoCs using generalized de Bruijn graph," *IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst.*, vol. 19, no. 8, pp. 1469–1480, Aug. 2011.

[4] .A. Ejlali, B. Al-Hashimi, P. Rosinger, S. Miremadi, and L. Benini, "Performability/energy tradeoff in error-control schemes for on-chip networks," *IEEE Trans. Very Large Scale Integr.(VLSI) Syst.*, vol. 18, no. 1, pp. 1–14, Jan. 2010.

[5] K. Sekar, K. Lahiri, A. Raghunathan, and S. Dey, "Dynamically config- urable bus topologies for high-performance on-chip communication," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol.16, no. 10, pp. 1413–1426, Oct. 2008.

[6] U. Ogras, P. Bogdan, and R. Marculescu, "An analytical approachfor network-on-chip performance analysis,"*Computer-Aided Design ofIntegrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 12,pp. 2001 -2013, Dec. 2010.

[7] A. E. Kiasari, Z. Lu, and A. Jantsch, "An analytical latency model fornetworks-on-chip," *Very Large Scale Integration (VLSI) Systems, IEEETransactions on*, vol. PP, no. 99, pp. 1 – 11, 2012.

[8] R. Marculescu, U. Ogras, L.-S. Peh, N. Jerger, and Y. Hoskote, "Outstanding research problems in noc design: System, microarchitecture, and circuit perspectives," *Computer-Aided Design of Integrated Circuitsand Systems, IEEE Transactions on*, vol. 28, no. 1, pp. 3 –21, Jan. 2009.

[9] F. Worm, P. Ienne, P. Thiran, and G. DeMicheli, "A robust self-calibrating transmission scheme for on-chip networks," *IEEE Trans. VLSISyst.*, vol. 13, no. 1, pp. 126–139, Jan. 2005.

[10] W. Dally and B. Towles, "Route packets, not wires: on-chip interconnection networks," in *Proc. of DAC*, 2001.