Research Journal of Applied Sciences, Engineering and Technology 7(22): 4714-4719, 2014
DOI:10.19026/rjaset.7.856
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: December 18, 2013 Accepted: January 20, 2014 Published: June 10, 2014

Research Article

An Effective Bidirectional Network on Chip with Pipelining and Self Reconfigurable Channel

[1]A. Rajalingam and [2]Bechtel Brabi
[1]Shinas College of Technology, Oman
[2]Saveetha Engineering College, India

Abstract: The Bidirectional Network on Chip (Bi-NoC) architecture is more efficient than the conventional architecture because in conventional architecture the unidirectional flow is used where as in Bi-NoC the two way data flow can be performed. The XY routing algorithm is used to perform the architecture. The Bi-NoC architecture allows each channel to transmit in all direction and increases the bandwidth, reduces the access latency and reduces power consumption. The pipelining architecture is used in this study instead of parallel architecture because this performs the reduction in size of the architecture with better result. Also it produces better transmission in the router with reduced traffic and also has less amount of access latency which enhances the performance through better resource utilization. In this proposed method the number of flip flops reduced by 35.9% and the access latency is reduced by 18.9% when compared to existing method.

Keywords: Channel allocator, masked and unmasked arbiter, pipelining, virtual channel

INTRODUCTION

The Network on Chip is the emerging technique in the field of Very Large Scale Integrated Circuits (VLSI). This NoC is used to communicate two or more IP cores in System on Chip (SoC). There are many SoC available whose interconnections are critical. That interconnection ensures reduced design time and IP reuse. NoC has the better improvement over the System On chip busses and also over the cross bar switches. This also reduces the power consumption while compared to the SoC i.e., the requirement of source power in the SoC is high and it can be reduced by the NoC architecture. By the use of NoC architecture in the field of VLSI it reduces the size of the architecture since it has reduced busses lines and transmission lines. The unit such as memories, processor cores and IP blocks exchanges the data the public transportation for the traffic information.

The NoC architecture can operate simultaneously on different data packets hence by using this NoC architecture the high level of parallelism is achieved. The benefit of NoC is that multiple point to point link can be connected with the help of the switches. By the use of switches the messages or the data can be relayed from source node to the destination node over several links. The computation time is separated from the calculated time. In the NoC architecture there are two routers the conventional router and the bidirectional router. In this conventional router the data can be send in only one direction whereas in bidirectional router the data can be send and receive simultaneously. This connects to the multipoint to multipoint link or the point to point connects using the same dedicated wire. In the NoC links the wires are designed to reduce the complexity for the predictable speed, reliability, noise, power etc. The ports in the bidirectional are reconfigurable. This is done by using the local information. Now why there is the need for NoC in the field of VLSI? This interconnection ensures reduced design time; also it is reduced and reuses architecture. Bus based interconnections were sufficient until now. But now:

- Shared bus is slow
- More components increase loading, hence drop in speed
- Lower performance and high power consumption

In the NoC router design with smaller foot prints and lower latency. The motivations in the NoC router is that more repeaters are needed into long wires in the case of SoC router this increases the cost of the architecture; also it has the larger chip size. In this NoC router it uses the critical global wiring resources by sharing them across different senders and receivers. By using the critical global wiring resource it simplifies overall design. The NoC architecture consists of data link, network and transport layer. The wiring is done using the physical layer.

Corresponding Author: A. Rajalingam, Shinas College of Technology, Oman

LITERATURE REVIEW

In the router architecture not all the buffers are used in a time hence the router architecture can sustain with its performance. In this study the different buffers can be dynamically reconfigure for each channel. It can use either the whole buffer slots or the part of the buffer slots based on the requirements of the neighboring buffers. According to Matos *et al*. (2011) each buffer a slot uses only its neighboring buffers on either side this reduces the connection cost. It has each channel will have more buffers i.e.; three times more buffer slots than the original at design time. In his manuscript he gave the importance to the buffers and the power consumption. According to Srinivasan *et al*. (2006) the sum of power consumed by header, switches etc., is equal to the total power consumed. Usually the router consists of five ports each port has its own buffer these ports can send as well as receive the data in the same port. The power consumption rate is with respect to the injection ratio and also due to the arbiter used in the output buffer. The router can connect to either the point to point link or multi point to multipoint link as shown in Ito *et al*. (2008). In this a transceiver is used i.e., two differential amplifier is used which acts both as a transmitter and receiver. In multi drop transmission couplers consumes larger power than the transmitter. In the virtual channel router each input and output is connected by a unidirectional link. In on chip 2D mesh the nearest neighbors will have two links close to each other. These links delivers the packets to the neighbors (Cho *et al*., 2009). We can describe the implementation of bandwidth adaptive network in the form of two dimensional meshes with adaptive bidirectional links. The reconfiguration can be done rapidly in response to changing traffic demands. The path of the packet is designed using the source and receiver (Ni and Mckinley, 1993).

An autonomous network reconfiguration system that enables a multi radio WMN to recover automatically from local link failures to preserve network performance (Kim and Shin, 2011). This outperforms existing failure-recovery schemes in improving channel-efficiency by more than 90% and in the ability of meeting the applications' bandwidth demands by an average of 200%. Wireless Mesh Networks (WMNs) are being developed actively and deployed widely for a variety of applications, such as citywide wireless Internet services public safety and environment monitoring (Akyildiz *et al*., 2005). The control algorithms such as credit based, such as on-off and ack/nack mechanisms regulate the traffic flow *locally* by exchanging control information between the neighboring routers (Ogras and Marculescu, 2010). The actual packet rate at the traffic source level cannot be directly controlled by the switch to switch flow control. In end to end control the major drawback is that larger overhead incurred while sending the feedback information.

Problem statement: The larger buffer size is required for transferring the packet to the destination. This larger buffer size can be avoided by many techniques (Lan *et al*., 2009). In the general router more power is needed for transferring the packet, caches and the independent memory controllers. The problems found in the SoC system are that it has the larger chip size, larger buffer size and number of transmission lines. The dead lock condition occurs if the packets transferred in the same direction at the simultaneous time. This forms the circular ring structure. This dead lock condition can be avoided by breaking the ring structure. Due to the presence of larger buffer size there exits higher latency (Bertsekas and Gallager, 1992).

METHODOLOGY

Proposed pipelining architecture: The BiNoC router dynamically reconfigure within the ports shown in Fig. 1. Each router has a set of ports and these ports communicate with each other. It consists of north, south, east, west and local port. The packets are received and forwarded to any of the appropriate ports. Each port consists of a buffer this temporarily stores the incoming packets.

This consists of an in-out port, cross bar switches, buffer, arbiter and the finite state machine. Where as in the bidirectional network on chip architecture it has the in-out ports, switch allocator, virtual channel, virtual channel allocator, round robin arbiter, channel control, routing logic and the cross bar switches.

Input output ports: The input and output port used in bidirectional NoC architecture is based on the priority. In this one port is designed with the higher priority and the other port is designed with the lower priority. A channel control protocol is used to determine its own transmission direction from a router to another router. By this bidirectional ports doubling of channel bandwidth is possible. This is done when two channels sends the packets concurrently to the same destination. The finite state machine is used in the input output channels shown in Fig. 2.

The use of input port and the output port are determined by using control channel. The control channel generates the in out select line. There is no chance of occurring conflicts and unpredictable situations when the in out select signals are assigned properly hence.

Bidirectional channel: The bidirectional channels are extensively used for supporting electronic design of SoC. The challenge in NoC is the distributed CDC protocol that would achieve certain criteria's such as correctness, high performance and low cost. In bidirectional channel the proposed direction decision mechanism is centralized and the packets are

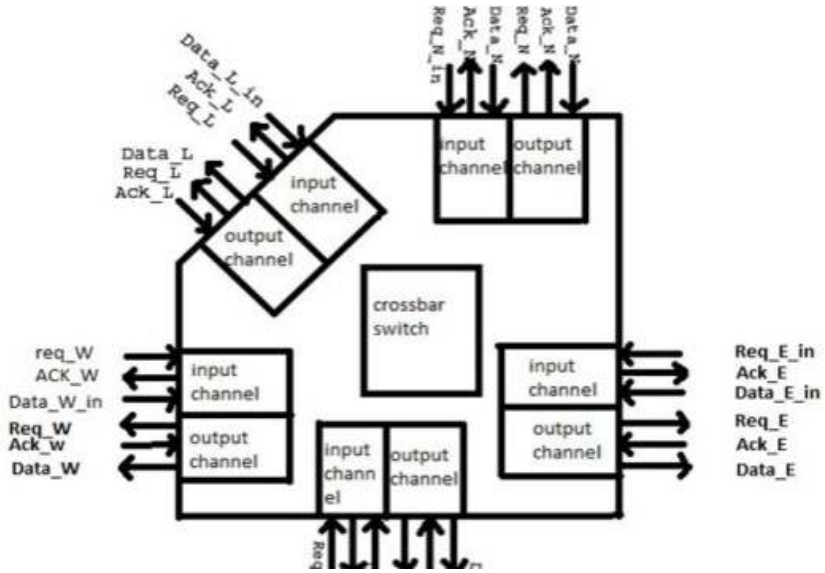

Fig. 1: General router

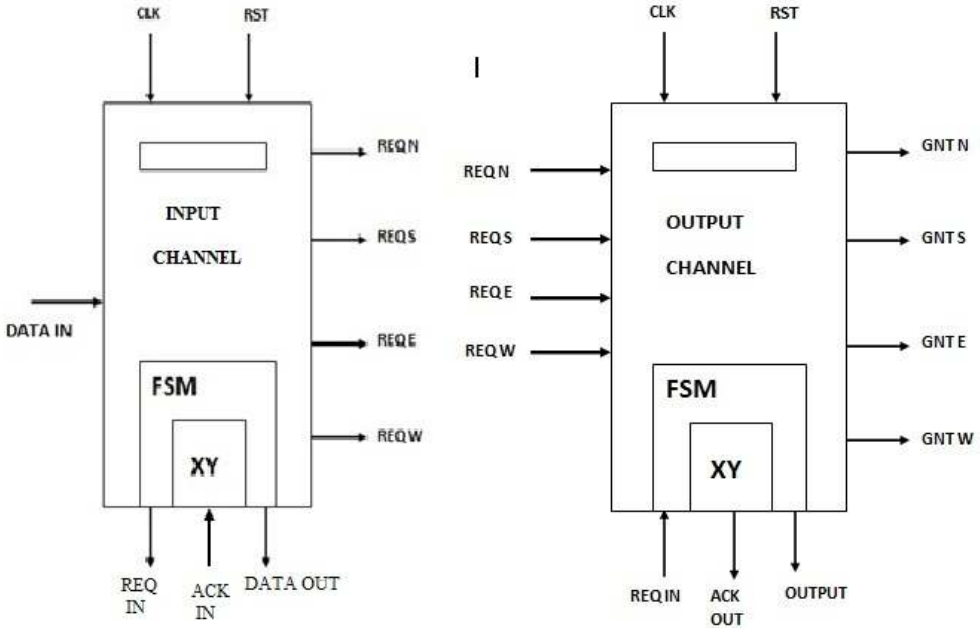

Fig. 2: Input and output channel

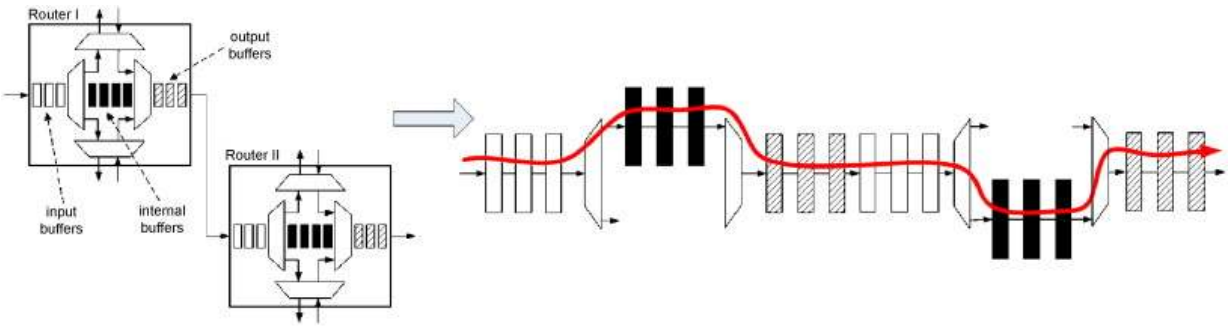

Fig. 3: Bidirectional channel

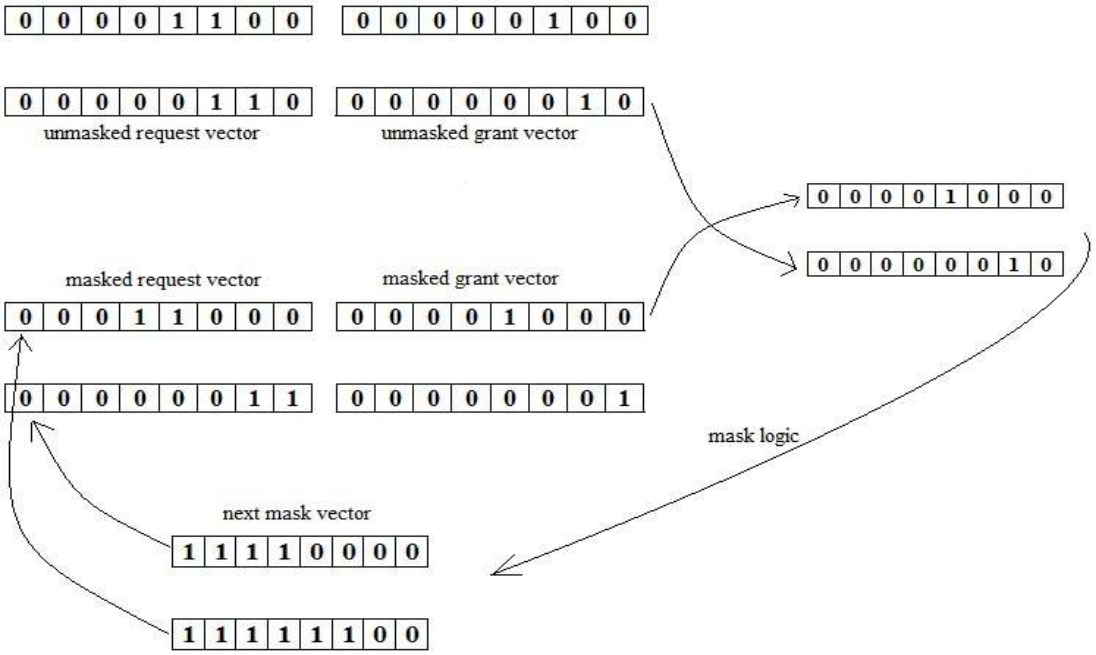

Fig. 4: Arbiter

configured according to application analysis results. This can send the data and receive the data at the same port. Also the packet or data from one router can be sending to the next router shown in Fig. 3. Hence these channel acts as the bidirectional channel.

Arbiter: The arbiter controls arbitration and resolves contention problem. It has the updates of all the ports and knows which port is busy and which is free. The packets from the input port may have the same priority and also the same destination. In this case the round robin arbiter is used. If two or more ports request the same destination port then the arbiter processes the data or packet by its priorities among different requests. The output port is released by the arbiter who is connected to the crossbar when the last packet has been finished transmission. Hence the packets can make use of the output ports by the arbiter. It works on the principle that the request which is just served should attain the lowest priority on the future rounds. If a port transmits a data from source to the destination the arbiter generates the control signal shown in Fig. 4.

In this study we use the masked and unmasked priority arbiter. In the masked arbiter, one vector is loaded to the mask register the grant is send. Since the vector is loaded in the register the currently served request cannot be served again. This force the arbiter to grant the next request.

The weight is calculated based on the counter; it counts the number of grants to requestor. Once it reaches the predefined weight of requestor the request will be blocked and tends to select the next requestor. The grant vector changes for each time if it is acknowledged in the very first cycle the counter starts the value from 1 else starts from 0.

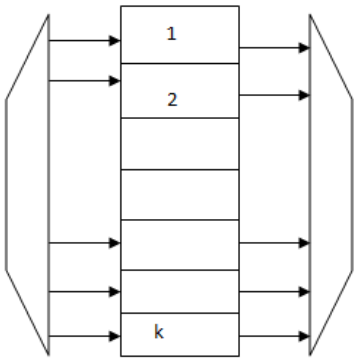

Fig. 5: Virtual channel

Virtual channel: The virtual channel in the router consists of number of channels with a small buffer. Also each channel will have a multiplexer and a de-multiplexer on either side.

Figure 5 VC performs the pipelining operation. By inspecting the physical link state and speculating the packet transferring maximizes the throughput and it avoids the useless VC allocations and there by increases throughput and propagating the packets of other VCs when a VC is congested. If the VC is dynamically allotted then the need for the flits are less. The packets from any of the inputs is stored as flits in the buffer and then send to the output port which is ready to accept the flit. These packets are sending to the destination based on priority. The virtual channel allocator allocates the

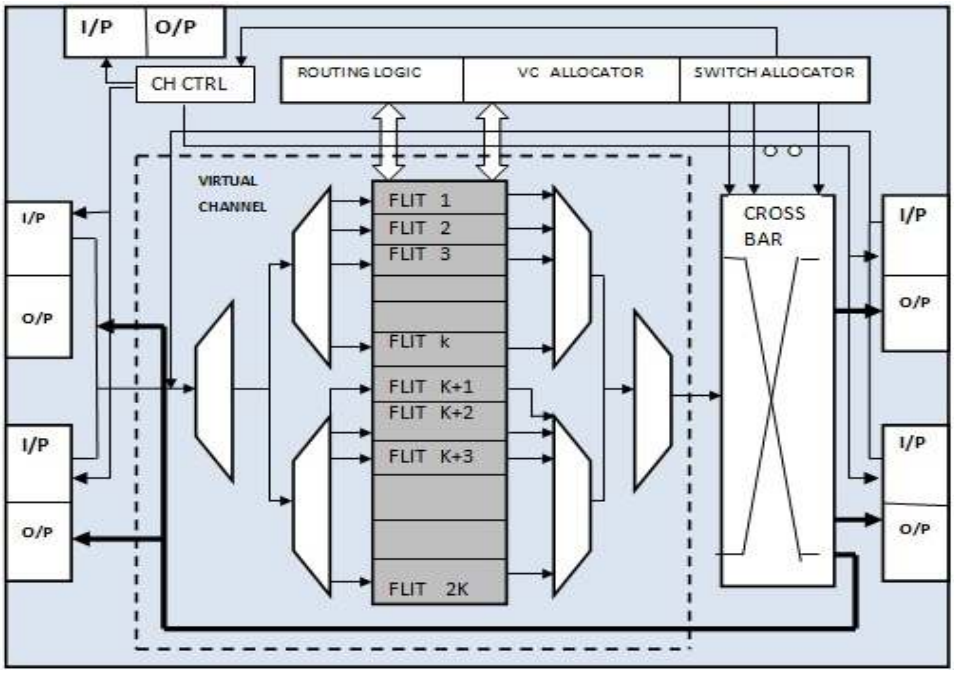

Fig. 6: Proposed router

channel to be processed. The each unidirectional virtual channel is managed by the pair of flit. The packets share the channel on flit by flit basis from flit 1 to the flit k.

The virtual channels are designed to avoid the dead lock condition in the packet transfer. This can also be used to improve the latency and throughput. Once the buffer accepts a flit from a packet it does not accept a flit from any other packet other than the first packet. i.e.,; the first flit of the second packet will be transferred only after transferring the whole flit from the first packet. The three situations when the packets not reach the destinations are live lock, starvation and deadlock. The physical channels are assigned to the virtual channel only if the channel is ready to route a flit.

Basic pipelines undertaken by the routers are divided into four steps.

First, a head flit arriving at input port gets decoded and buffered according to its VC id in the Buffer Write (BW) stage. Second, the routing logic performs the computation to decide its output port. Third, the VC Allocation (VA) unit decides the head flits requesting for VCs. Fourth, after the VC obtaining process, the packet flits proceed to the Switch Allocation (SA) stage which arbitrates for the output ports shown in Fig. 6.

Switching techniques: Switching examines the header and decides where to send the message and then it starts forwarding immediately. When a head of message is blocked message stays strung out over the network and potentially blocking the other messages. Hence it produces the low latency and fewer amounts of buffers are required. The head flit builds path in the network and the remaining flits in the packet follow in the pipeline. For the given packets, the flits occupy the sequence of buffers and links forms the wormhole routing. The length of the packet is equal to the number of flits in packet. It spans the path between the source and destination.

Deadlock: The deadlock occurs when packets are allowed to hold resources when requesting the other. This dependency forms a cycle. This makes all the packets to wait forever. These deadlocks can be avoided by using virtual channels. The deadlock can also be avoided by breaking the cycle. There are many dead lock avoidance techniques such as turn model based routing algorithm and dimension ordered routing algorithm. The possible dead lock condition is pressure based channel direction control. This requires ration bandwidth among different flows to their corresponding pressure.

Routing protocol: The routing protocol selects the routes between two nodes in a network. Also it specifies how the routers communicate with each other. There are two basic routing protocols adaptive and oblivious routing protocol. In this adaptive routing protocol consists of many types and each type will have the special features. In this XY routing algorithm is used. It routes between given pair of nodes and follows the same path between the nodes. XY routing checks first in horizontal x direction and then checks the vertical y direction and forms the route to the receiver.

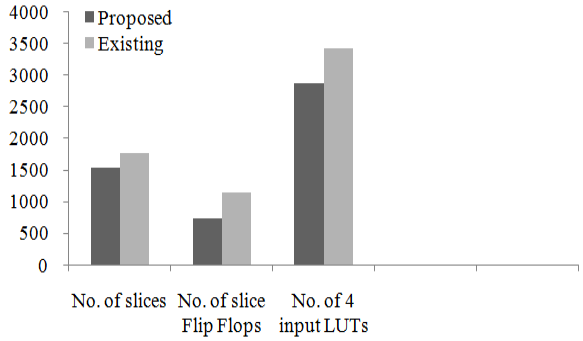

Fig. 7: Comparison of slices, flip flops and LUTs

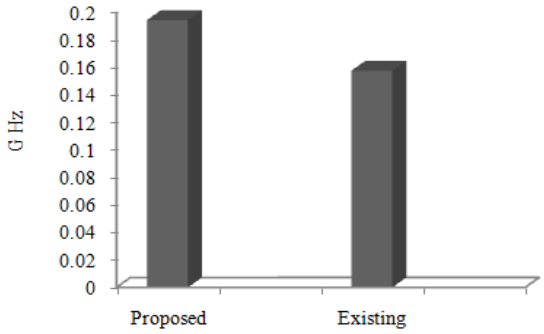

Fig. 8: Frequency chart

By using this XY routing algorithm the traffic does not extends over the whole network. Hence the whole network is equalized from traffic load. The feature of the XY routing algorithm is simple and it never runs the deadlock and lives locks.

EXPERIMENTAL RESULTS

In this we are using the Spartan 3E which has a xc3s 400 FPGA to functionally verify the stand alone router. We use the Xilinx 10.1 to synthesis. Also we are using the modals XE III 6.3C to simulate the model. The BiNoC architecture increases the speed in the router. Since the virtual channel is used it avoids the deadlock condition and the packets transferred need not to be waited for long time to reach the destination. The size of the architecture is minimized and hence the power consumption is reduced in bidirectional NoC architecture. This architecture performs self reconfigurable; it is done using the local information. The access latency is very less while using this architecture.

Figure 7 shows that there is an efficient reduction in the number of slices, slice flip flops and input LUTs in the proposed system than those obtained in the existing generic router these reduction rates are given as, The slices reduced by 12.5%, the number of flip flops reduced by 35.9% and the input LUTs are reduced by 16.4%.

Figure 8 shows that the frequency in the proposed model is higher than that of the existing system. Hence the latency is less in the proposed system. The latency reduction rate is found as 18.9% than the generic router.

CONCLUSION

The bidirectional network on chip architecture enhances speed in the router. In this the pipelining channel is used hence the access latency is 18.9% less than the generic router architecture. The virtual channel performs the pipelining architecture also the virtual channel avoids the dead lock condition in the router architecture. The XY routing algorithm prevents the dead lock and live lock condition in the router. The total number of flip flops used in the bidirectional router is 35.9% lesser than the generic router.

REFERENCES

Akyildiz, I.F., X. Wang and W. Wang, 2005. Wireless mesh networks: A survey. Comput. Netw., 47(4): 445-487.

Bertsekas, D.P. and R.G. Gallager, 1992. Data Networks. 2nd Edn., Prentice Hall, Englewood Cliffs, New Jersey, ISBN: 0132009161.

Cho, M.H., M. Lis, M. Kinsy, K.S. Shim, T. Wen and S. Devadas, 2009. Oblivious routing in on-chip bandwidth-adaptive networks. Proceeding of 18th International Conference on Parallel Architectures and Compilation Techniques (PACT'09), pp: 181-190.

Ito, H., M. Kimura, K. Miyashita and T. Ishii, 2008. A bidirectional-and multi-drop-transmission-line interconnect for multipoint-to-multipoint on-chip communications. IEEE J. Solid-St. Circ., 43(4): 1020-1029.

Kim, K.H. and K.G. Shin, 2011. Self-reconfigurable wireless mesh networks. IEEE ACM T. Network., 19(2): 393-404.

Lan, Y.C., S.H. Lo, Y.C. Lin, Y.H. Hu and S.J. Chen, 2009. Binoc: A bidirectional noc architecture with dynamic self-reconfigurable channel. Proceeding of 3rd ACM/IEEE International Symposium on Networks-on-Chip (NoCS, 2009), pp: 266-275.

Matos, D., C. Concatto, M. Kreutz and F. Kastensmidt, 2011. Reconfigurable routers for low power and high performance. IEEE T. VLSI Syst., 19(11): 2045-2057.

Ni, L.M. and P.K. Mckinley, 1993. A survey of wormhole routing techniques in direct networks. J. Comput., 26(2): 62-76.

Ogras, U.Y. and R. Marculescu, 2010. Prediction-based flow control for network-on-chip traffic. Proceeding of 43rd ACM/IEEE Design Automation Conference, pp: 839-844.

Srinivasan, K., K.S. Chatha and G. Konjevod, 2006. Linear-programming-based techniques for synthesis of network-on-chip architectures. IEEE T. VLSI Syst., 14(4): 407-420.