Asynchronous Router for Network-on-Chip on FPGA

Gauri Suresh Bhosale
ME student
Department of Electronics
K. J. Somaiya College of Engineering

Arati S. Phadke
Associate Professor
Department of Electronics
K. J. Somaiya College of Engineering

ABSTRACT
In Network-on-chip router is the main block where one of the major decisions about the route direction is taken. This paper presents asynchronous router implemented using handshaking signals. Distributed routing with 3X3 Mesh topology is used in this design. 2D Mesh is the most common topologies due to its grid-type shape and regular structure which is most appropriate for the two dimensional layout on a chip. The design is synthesized for the Stratix II EP2S15F484C3 FPGA using Quartus II software. The router supports maximum of five simultaneous routing requests.

General Terms
XY routing algorithm is used in this design for deadlock free and livelock free router for network on chip. Wormhole switching technique is used to minimize area requirement.

Keywords
Network on Chip; Wormhole switching; XY routing algorithm; Asynchronous router

1. INTRODUCTION
Now days, with advanced fabrication technology, intellectual property (IP) cores such as processors, input-output units and different types of memories are fabricated on single chip, also called as System-on-Chip(SOC). Traditionally bus architecture was used for communication between the IP cores. But with increase in number of IP cores, complexity of communication system also increases. Also number of cores to be connected to the bus is limited [4]. Hence Network-on-Chip (NOC) design approach is adopted for communication between the IP cores.

In NOC approach, IP cores communicate by sending packets to one another over the network. Instead of connecting the IP cores by dedicated wires, they are connected to the network. Each IP core communicates with all other cores, not just its neighbours, through the network by sending packets [3]. The network logic utilizes the small amount of area (maximum 2% for this design) in each core. Number of neighbours of the IP core is decided by topology of the network. In this paper router is designed for 3X3 mesh topology [6] as shown in Fig.1. In Mesh topology middle routers are connected to the four adjacent routers and one IP core. The routers at the edges and corners have four and three connections respectively.

XY routing algorithm [8] is used as it is deadlock free and livelock free routing. Wormhole switching technique is used in this design. In wormhole switching, the packets are divided into fixed length flow control units called as flits. Advantage of this technique is input and output buffers are expected to store only a few flits. As a result, the buffer space requirement is small compared to that generally required for packet switching.
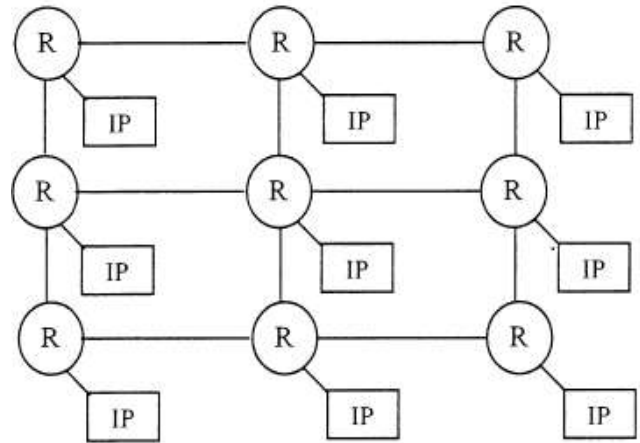

Fig 1: 3X3 Mesh Topology

The first flit, that is, header flit, of a packet contains routing information. Header flit decoding enables to establish the path and subsequent flits simply follow that path.

The rest of this paper is organized as follow: Section II presents the related work for NoC routers. Router design is given in section III.   Section IV presents the simulation results. Conclusion and future work is presented in section V.

2. RELATED WORK
Different types of NoC routers are suggested by researchers and academic institutions [1]-[8].

In [3], a simple and efficient mechanism is proposed to increase the throughput of router in NoC. Generally packets are divided into flits; Head flit, data flit and End flit. In wormhole switching, for the   Head flit routing decision is taken and remaining flits of that packet follow the same path. At Head flit more time is require for routing decision as compared to other flits. In synchronous routers, generally single clock is used for all the flits. Hence in fully adaptive wormhole routers, the routing decision time causes performance degradation. References [3] uses different clocks in a head flit and body flits, because the body flits can be forwarded immediately and the FIFO usually operates faster than route decision logic. This technique improves throughput. But it results in increase in area requirement because of extra hardware for clock boosting circuit.

Reference [2] uses handshaking communication protocol. The buffers are designed at input as well as at the output ports. This reduces router latency but at the cost of increase in design area.

Reference [1] also uses handshaking communication protocol but clock signal is completely removed in this design. Area requirement is less as buffer is designed only at input port with buffer size equal to one flit. In [1] source routing is used. Hence logic circuit for routing decision is not required. This feature reduces the area of the router. But it requires routing table at the network adapter of each node. With increase in number of nodes area of the routing table also increases. In this design distributed routing is designed. Area requirement of the router is more compared to [1] but with increase in number of nodes area of network adapter will not increase as the routing table is not required at network adapter.

3. ROUTER DESIGN

The router is designed for 3X3 mesh topology having four ports (West, South, East and North) to connect with other routers and a local port to connect with IP. That is the router has total five input ports and five outputs ports [1]. The Wormhole switching technique [7] is used in this design as the buffer space requirement in the switches is small. Flit size is kept equal to 16 bits. There are three types of flit; Head flit, Data flit and End flit. Head flit contains source and destination addresses. Data flit contains the message and End flit indicate end of the packet. The last flit that is End flit is used to indicate end of the packet. First two bits in the flit indicate type of the flit.

Table 1. Flit type

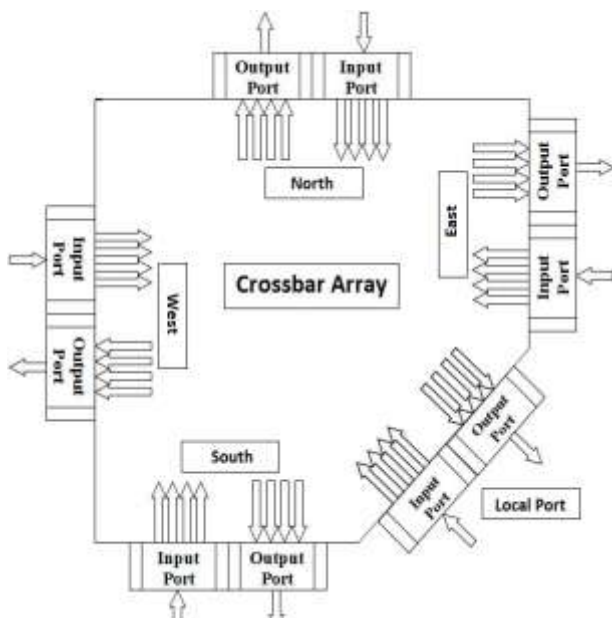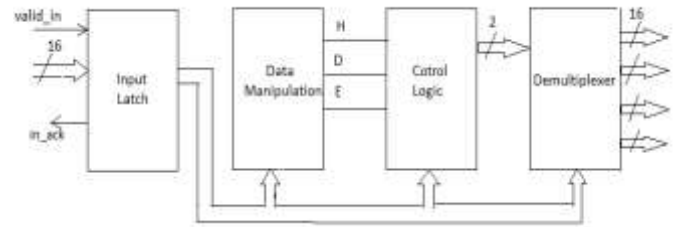| First 2 bits	Type of the flit
10	Head flit
0X	Data flit
11	End flit

Fig 2: Block diagram of the router

Fig 3: Block diagram of router input port architecture

XY routing algorithm is used for deadlock free and livelock free router. The router is implemented such that when a flit enters the router it can only be routed to the output ports of the other four interfaces and cannot be routed back in the same direction. Therefore, it is only necessary to have four connections from each input port to the four output ports of the other channels [1]. The block diagram of the router design is shown in Fig. 2.

3.1 Router Input Port

The input port comprises of input latch, data manipulation stage, control logic and demultiplexer. When flit arrives at input port with valid_in handshaking signal, it will be latched in input latch. Data manipulation stage checks the type of the flit. If received flit is head flit, control logic generates the two bit control signal depending upon the destination address given in the flit. These control signals will remain unchanged for entire packet. This means subsequent flits like data flits, end flit follow the same path as head flit. When new head flit arrives in input latch, the control signals will be updated. Depending upon the control signal generated by control logic stage, demultiplexer directs the flit towards one of the output port. For example, North input port will send the flit towards the East or West or South or Local output port. Fig. 3 shows block diagram of input port architecture.

3.2 Router Output Port

The output port of the router design comprises arbitration stage and output latch. The arbitrator is designed such that it is able to handle up to four request signals asserted simultaneously. In the case of two or more header request signals are asserted simultaneously, the proposed design arbitrate between them according to priority and give access to one signal only. The other requests are held until they gain access.

Table 2. Priority level assigned to input ports

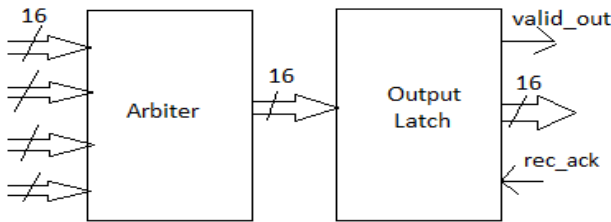| Priority level	East output port	West output port	South output port	North output port	Local output port
Highest	West	Local	West	West	West
	South	South	Local	South	South
	Local	East	East	East	East
Lowest	North	North	North	Local	North

Fig 4: Block diagram of router output port

However, it keeps the access open for the whole packet before granting it to another header request signal. Fig. 4 shows architecture of output port. Table 2 shows priority level assigned to the output ports at each input port.

3.3 Asynchronous Communication Mechanism

Handshaking communication protocol is implimented in this paper, when the data is put on the line; the existence of the valid data is informed to the next router. Next router takes the data from the line and transmits its confirmation to the sender router [2]. So in addition to the flits sending and receiving channels, valid_in, valid_out, in_ack, rec_ack signals are required. Valid_out is the output and whenever the data is ready in the output port, this signal makes transition from high level (logic 1) to low level (logic 0) and waits for rec_ack signal. Likewise each input port after finding high to low transition on valid_in signal, reads the data on this port and make high to low transition on in_ack signal. The link of two ports from two neighbour routers is shown in Fig. 5.
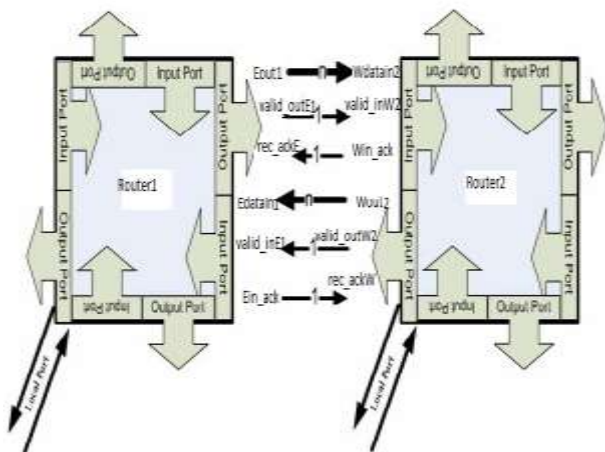

Fig 5: Communication mechanism between two routers

4. SIMULATION RESULTS

The proposed router design incorporated in 3X3 2D-Mesh Topology NoC with XY routing algorithm is designed using Verilog Hardware Description Language (Verilog HDL) and simulated using the Modlesim simulator. The design is synthesized for the Stratix II EP2S15F484C3 FPGA using Quartus II software.

4.1 Results

The functionality of the router is verified by performing various tests as described further. The arbitration was verified by applying four packets to the input at exactly the same time by asserting their header request signals simultaneously. All the header flits had the same routing direction which enabled the simulation to test the worst case of arbitration at the output port. Result is shown in Fig. 6. In Addition, the five data flits with different routing direction were applied simultaneously to all the available input ports. Result shown in Fig.7 indicates ability of router to support five simultaneous requests.

4.2 Hardware Utilization

Synthesis report shows that only 6% of the combinational ALUs and less than 1% logic registers are used for implementing the router.

Table 3. Hardware Utilization

Resources	Used	Available	Utilization
Combinational ALUs	707	12480	6%
Dedicated logic registers	1 05	12,480	<1%
Total pins	185	343	54%

Synthesis is also done using SPARTAN-3 FPGA. Results show that 344 slices were utilized for implementing the router where as in [1] only 266 slices were used. Area of the router in this paper is more but it will not increase the overhead of the network adapter with increase in number of nodes in SOC. In [1] routing path is given in header flit. Therefore, with increase in number of nodes in SOC, size of the header flit as well as routing table will also increase. In this paper distributed routing [8] is used hence no need of routing table.
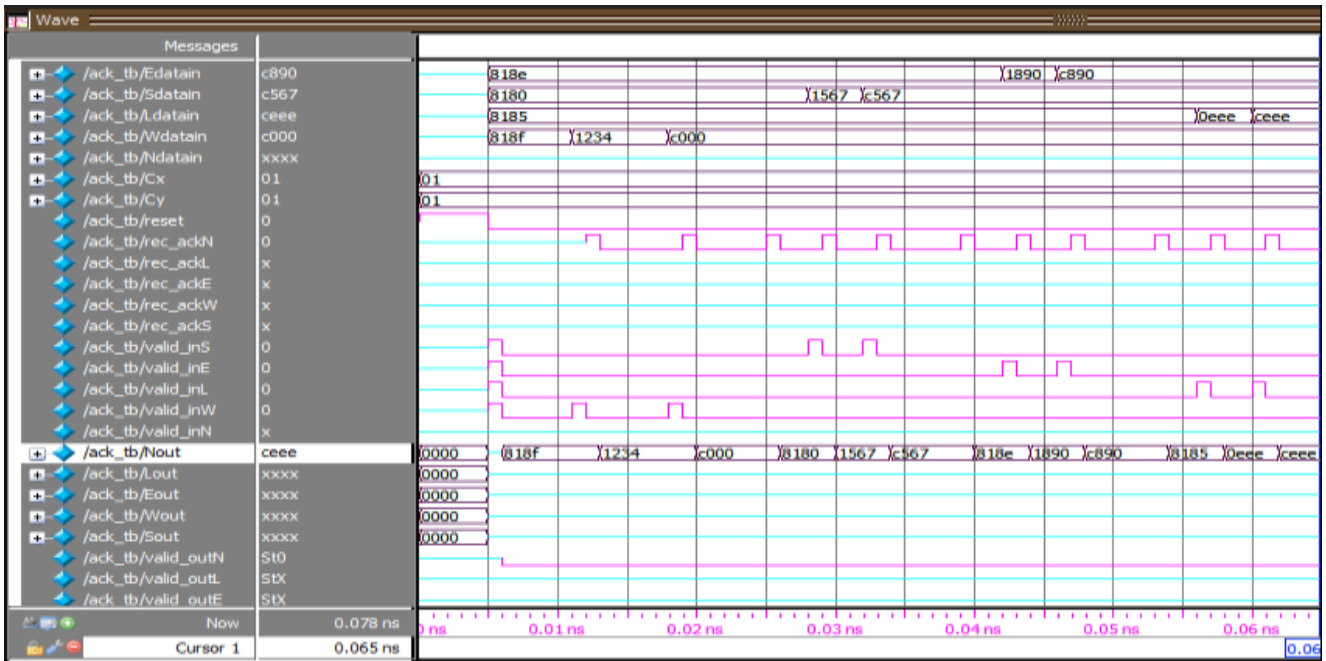
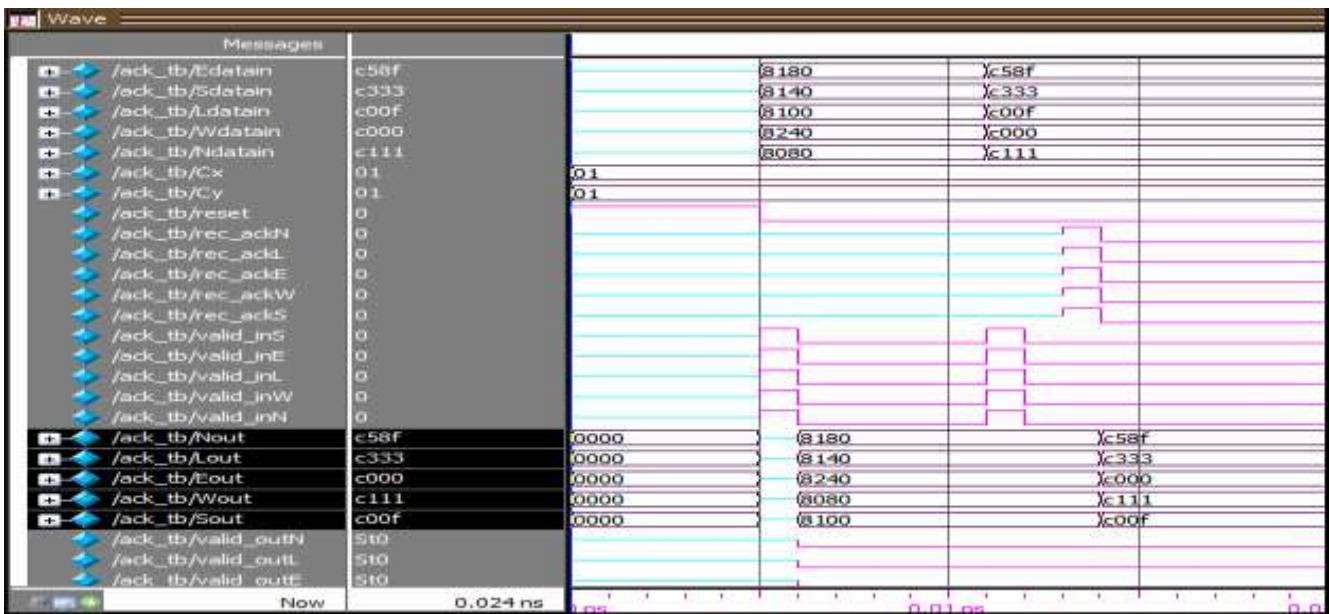**Fig 6: Packets from different input ports to destination simultaneously**

Fig 7: Packets from different input ports with different destination addresses simultaneously.

5. CONCLUSION AND FUTURE WORK

Design of an Asynchronous low area clock-less NoC router for an FPGA based on handshaking protocol is presented. The router supports up to five routing request simultaneously by utilizing low area and reduced logic design complexity. Clock signal is completely removed in this design. The work on the router is in progress to enhance the latency by implementing buffers at input ports.

6. REFERENCES

[1] Hatem, F. O., & Kumar, T. N. (2013, April). A low-area asynchronous router for clock-less network-on-chip on a FPGA. In Computers & Informatics (ISCI), 2013 IEEE Symposium on (pp. 152-158). IEEE.

[2] Asghari, S. A., Pedram, H., & Khademi, M. (2009, October). A flexible design of network on chip router based on handshaking communication mechanism. In

Computer Conference, 2009. CSICC 2009. 14th International CSI (pp. 225-230). IEEE.

[3] Dally, W. J., & Towles, B. (2001). Route packets, not wires: on-chip interconnection networks. In Design Automation Conference, 2001. Proceedings (pp. 684-689). IEEE.

[4] Pande, P. P., Grecu, C., Jones, M., Ivanov, A., & Saleh, R. (2005). Performance evaluation and design trade-offs for network-on-chip interconnect architectures. Computers, IEEE Transactions on, 54(8), 1025-1040.

[5] Lee, S. E., & Bagherzadeh, N. (2006, October). Increasing the throughput of an adaptive router in network-on-chip (NoC). In Proceedings of the 4th international conference on Hardware/software codesign and system synthesis (pp. 82-87). ACM.

[6] Sonal S. Bhople, M. A. Gaikwad. A Comparative Study of Different Topologies for Network-On-Chip Architecture. International Journal of Computer Applications (0975 – 8887) "Recent Trends in Engineering Technology-2013" (pp. 27-29)

[7] Pande, P. P., Grecu, C., Jones, M., Ivanov, A., & Saleh, R. (2005). Performance evaluation and design trade-offs for network-on-chip interconnect architectures. Computers, IEEE Transactions on, 54(8), 1025-1040.

[8] Chawade, S. D., Gaikwad, M. A., & Patrikar, R. M. (2012). Review of XY Routing Algorithm for Network-On-Chip Architecture. International Journal of Computer Applications, 43(21).

[9] Zhang, W., Wu, W., Zuo, L., & Peng, X. (2009, December). The buffer depth analysis of 2-Dimension mesh topology Network-on-Chip with Odd-Even routing algorithm. In Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on (pp. 1-4). IEEE.

[10] Glass, C. J., & Ni, L. M. (1992, April). The turn model for adaptive routing. In ACM SIGARCH Computer Architecture News (Vol. 20, No. 2, pp. 278-287). ACM.