

A Dynamic NOC Arbitration Technique using Combination of VCT and XY Routing

¹Bharati B. Sayankar, ²Pankaj Agrawal

¹ Electronics Department, Rashtrasant Tukdoji Maharaj Nagpur University, G.H. Raisoni College Of Engineering, Nagpur, Maharashtra, India

² Electronics Department, Rashtrasant Tukdoji Maharaj Nagpur University, G.H. Raisoni Academy of Engineering & Technology, Nagpur, Maharashtra, India

Abstract - Continuous scaling of CMOS expertise makes the integration of large number of heterogeneous devices which results in efficient communication on a single chip. This is the purpose for which competent routers are desirable as a result of which communication takes place between these devices. The proposed methodology gives the method for on-chip routers based on combination of the XY and (Virtual cut through) VCT router for better and optimized data transfer. In this paper the proposed design of on-chip router give the outcome as the optimized routing output because of the priority allocate to the input data packet. The dynamic arbitration technique using VCT and XY is explained in this paper.

Keywords - Arbiter, Network on chip (NOC), On chip Router, XY Router, Virtual out through (VCT) router, Cross Bar Switch.

1. Introduction

In order to meet the growing computation-intensive applications as well as low-power necessities for high performance systems, the amount of computing resources on a single-chip has increased. Inadvertently to build a System-on-Chip (SoC), by adding many computing resources such as specific IPs, CPU, DSP etc. the interconnection between resources is another challenging issue. In nearly all SoC applications, a shared bus interconnection which requires arbitration logic to serialize several bus access requests is designed to aid communication with integrated processing units which is due to its simple control traits of low-cost. However, this type of shared bus architecture does not scale very well because at a time only one master can utilize the bus that means all the bus accesses should be serialized by the arbitrator.

This scalable bandwidth constraint could be removed by using packet-switched on-chip micro-network of

interconnects, popularly known as Network-on-Chip (NoC) architecture. The fundamental initiative originated from conventional major multi-processors and dispersed computing networks.

1.1 Network on Chip

A variety of interconnection techniques are presently in use counting crossbar, buses and NOCs. Of these, later two are dominant in research community. But buses bear reduced scalability which is because of the increase in number of processing elements that degrades the performance drastically. Therefore they are not taken care of where the processing entities are greater in number. To remove this constraint the concentration has dawned to the packet-based on-chip communication networks, called as Network-On-Chip (NOC). Figure 1 shows the basic Network-On-Chip Architecture [1].

A conventional NoC comprises of computational processing elements (PEs), network interfaces (NIs), and routers as shown in Fig. 1. The latter two comprise the communication architecture. Before utilizing the router backbone to go across the NoC the NI packetizes the data. Each PE is attached to an NI which connects the PE to a local router. The packet is moved further hop by hop on the network listening to the decision made by each router, when a packet was transferred from a source PE to a destination PE. The packet is first received and then stored in an input buffer for every router. After this the control logics in the router are accountable to take routing decision and channel arbitration. At last, the packet which is granted will traverse through a crossbar to the next router. The procedure is repeated till the packet arrives at its destination.

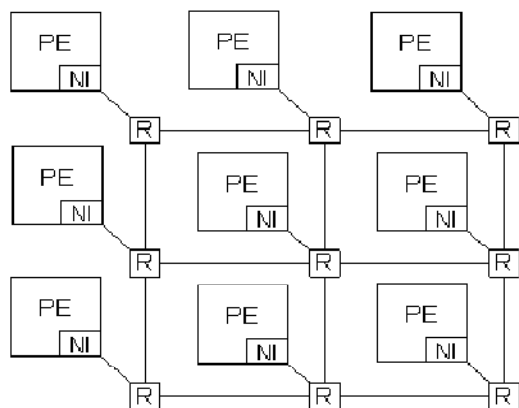


Fig. 1 PE – Processing Element
 R - Router
 NI – Network Interface
 Fig:1 Typical NoC Architecture

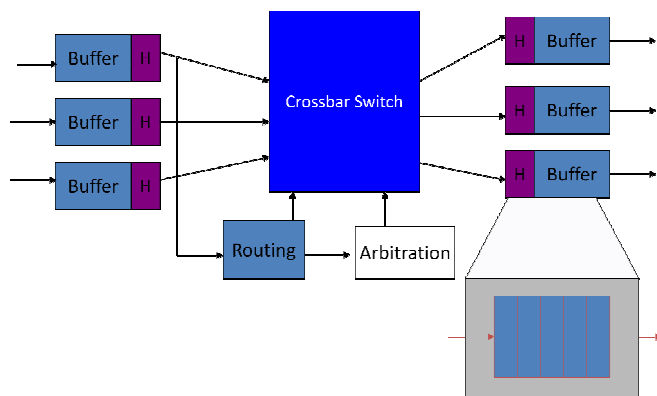


Fig. 2 Typical NoC Router

1.2 On-Chip Router

The main part of an on-chip network is the router, which undertakes crucial task of coordinating the data flow. Fig. 2 shows the typical Network-On-Chip router. The router operation revolves around two fundamental regimes: (a) the data path and (b) the associated control logic. The data path consists of number of input and output channels to facilitated packet switching and traversal [1] [2].

Generally 5 input X 5 output router is used. Out of five ports four ports are in cardinal direction (North, South, East, and Waste) and one port is attached to its local processing element. 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.

2. Routing Techniques

In this paper for the creation of the NoC router we are going to use two different routing techniques and they are as follows:

1. X-Y Routing
2. VCT (Virtual cut through) Routing

2.1. X-Y Routing

X-Y routing is determined completely from their addresses. In X-Y routing, the message travels “horizontally” (in the X-dimension) that is from the source point to the “column” comprising the destination, from where the message starts travelling vertically. In X-Y routing X direction is determined first and then the Y direction is determined. There are four possible

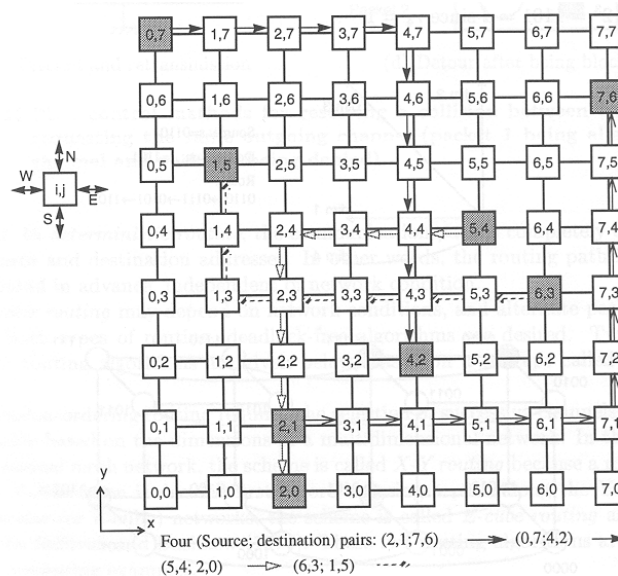


Fig. 3 X-Y Routing Example

direction pairs in X-Y Routing, east-south, east-north, west-south and west-north. Fig. 4 shows the simple example of X-Y routing [4].

Advantages for X-Y routing:

- Very simple to implement
- Deadlock-free

2.2. VCT Routing

Virtual Cut Through routing is very similar to the message switching, with the difference that is as a message comes

in an intermediate node plus its chosen departing channel is free, then in distinction to message switching, the message is sent out of the adjacent node towards its destination prior to its complete reception at the node. If the message is not allowed due to busy output, the channel comes with a message buffered in an intermediate node. Therefore, the interruption due to unnecessary buffering in front of an unused channel can be avoided [5]. VCT routing technique used to create number of virtual channel to pass the data packet allowed by the arbiter in the sequential order.

3. Router Design

The design of router mainly consists of three parts:

1. FIFO
2. Arbiter
3. Crossbar

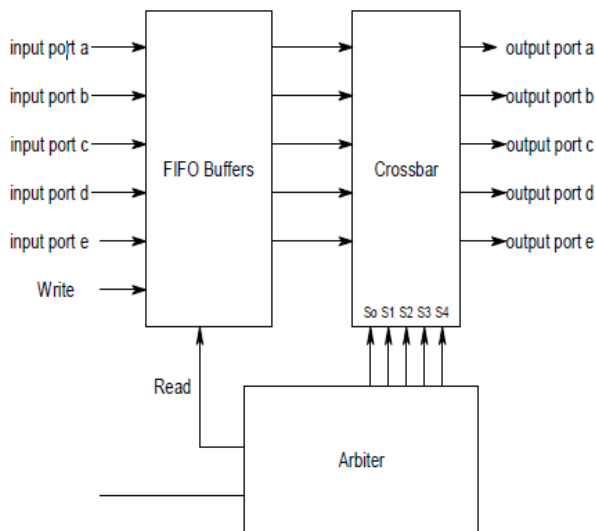


Fig.4 5x5 Input-Output port Unidirectional On-Chip Router

as shown in Fig.4.

3.1 FIFO Buffer

Buffering is extremely important in NoC for congestion control and flow control. Calculating proper size of buffer is the solution for receiving best performance. In this design the depth of buffer is used to provide input buffering. The extent of buffer is equivalent to the packet size. Because the depth of buffer is four, minimum four clock signals are requisites for first packet to come out of the buffer.

3.2. Arbiter

Arbiter controls the resolution of the ports and solves contention problem. The updated status of all the ports is

kept by it and knows the number of ports that are free and the ports are communicating with one another. In proposed work, we are using “Round Robin Arbitration Algorithm”. Packets with the same priority and meant for the similar output port are listed with a round-robin arbiter. Fig. 5 demonstrates the example of dynamic priority round robin arbiter.

If in a given period of time, many input ports request were there with the similar output or resource, the arbiter is responsible completing the priorities amongst lots of diverse request inputs. The arbiter will liberate the output port that is connected to the crossbar, once the last packet has finished diffusion. This would result in the use of output by the arbitration of arbiter by other waiting packets.

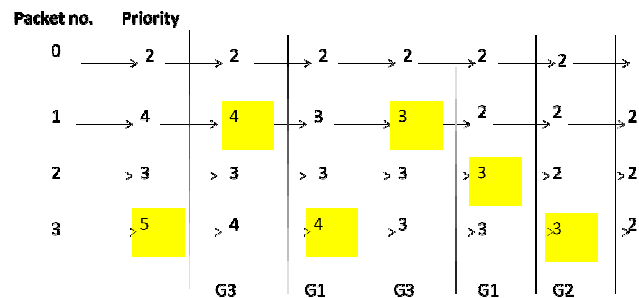


Fig. 5 Dynamic priority round robin arbiter.

A round-robin arbiter operates on the principle that a request which was just served should have the lowest priority on the next round of arbitration. Depending upon the control logic arbiter generates select lines for MUX based crossbar and read or write signal for FIFO buffers.

3.3. Crossbar

A crossbar switch is a switch connecting numerous inputs to multiple outputs in a matrix manner. The diagram of crossbar switch has 4 outputs and 4 inputs. In design given in Fig.6, every input port is enforced to share a single crossbar port even when multiple flit could be sent from different virtual-channel buffer. This restriction allows keeping crossbar size small and independent of the number of virtual-channel.

3.3.1 Reconfigurable Cross Bar Switch

Reconfigurable crossbar switch has mainly three blocks: (1) connection matrix, in which each and every topologies are implemented; (2) decoder, it converts the reconfigurable bits for a matrix bits set and (3) pre-header analyzer. this third block could be added in the packet with the output destination by Network processor.

Reconfigurable crossbar switch (RCS) uses reconfiguration bits to employ the topology in the space. Only the Reconfiguration Unit and instruction set of the network processor are capable to change those bits in order to implements new topologies. Although one instruction can modify the 01 and 10 formats the 00 format is restricted to reconfiguration unit [3].

4. Proposed Router Architecture

Figure 7 shows the overall proposed on-chip router technique we implemented in this paper. The input data packets gives to arbiter with priority as shown in Fig.7, so the dynamic output from the arbiter is produced. Then the output is send to VCT router to find the shortest path for the transferring the data packet through the bus. Then X-Y router is used for sending the packets in same direction from the bus.

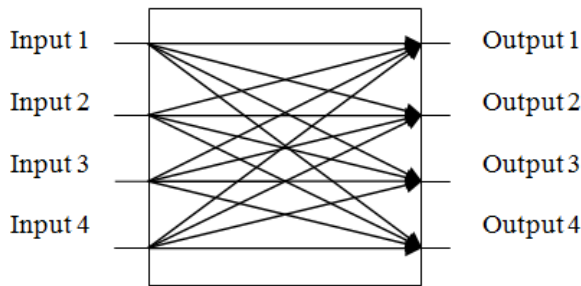


Fig. 6 4x4 Cross Bar Switch

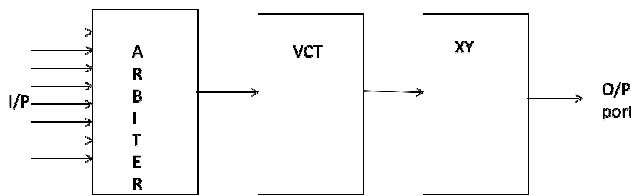


Fig. 7 Proposed NoC router technique.

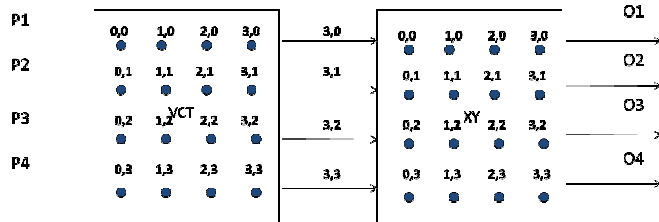


Fig. 8 Combination of VCT routing and X-Y routing.

Fig. 8 shows the combination of this both routing techniques. The example of 4 packets has been taken . It is also applicable for random number of packets.

5. Simulation Result

In the proposed architecture simulation result has been taken using the XIINX 13.1 simulator. Fig. 9 shows the output of the X-Y router. Synthesis report shows all the time related information of the X-Y router, like total delay, input time, output time, total path delay, source, destination, etc.

Fig. 10 shows the number of devices utilized by the X-Y router to perform its task. This will give the total energy utilized by the router.

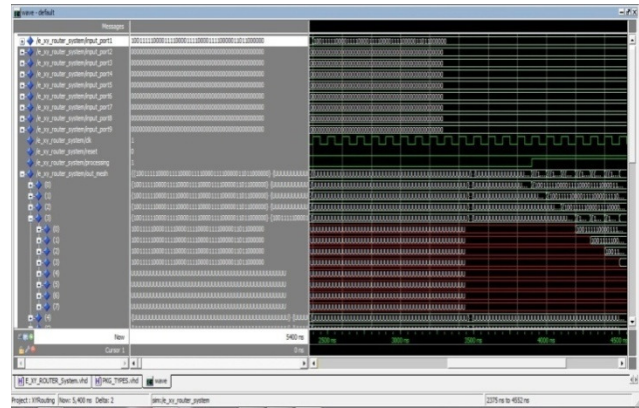


Fig. 9 Output of X-Y Router

Project File:	xyv.xise	Parser Errors:	No Errors
Module Name:	E_Router	Implementation State:	Synthesized
Target Device:	xc3e200-5F256	Errors:	No Errors
Product Version:	ISE 13.2	Warnings:	20855 Warnings (0 new)
Design Goal:	Balanced	Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:	
Environment:	System Settings	Final Timing Score:	

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	127	1920	6%
Number of Slice Flip Flops	53	3840	1%
Number of 4 input LUTs	230	3840	5%
Number of bonded IOBs	41	173	23%
Number of GCLKs	1	8	12%

Detailed Reports					
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sun Jan 13 19:53:52 2013	0	20855 Warnings (0 new)	1 Info (0 new)
Translation Report					

Fig. 10 Device utilization of X-Y router

Fig. 11 shows the output of the VCT router and Fig. 12 shows the number of devices utilized by the VCT router to perform its task. This will give the total energy utilized by the router.

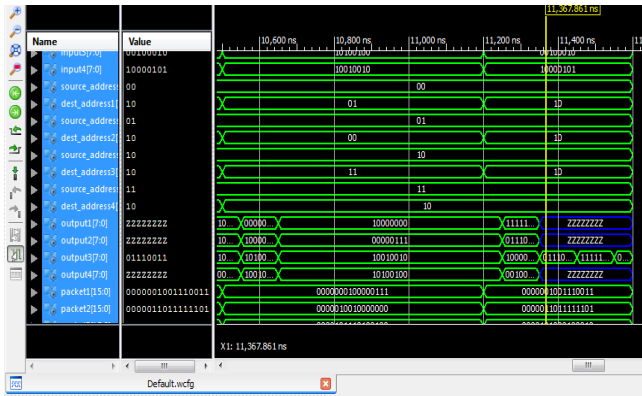


Fig.11 Output of VCT router

E_New_VCT_System Project Status			
Project File:	vct.xise	Parser Errors:	No Errors
Module Name:	E_New_VCT_System	Implementation State:	Synthesized
Target Device:	xc3e700a-4fg484	Errors:	No Errors
Product Version:	ISE 13.2	Warnings:	5 Warnings (1 new)
Design Goal:	Balanced	Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:	
Environment:	System Settings	Final Timing Score:	

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	148	5888	2%
Number of Slice Flip Flops	52	11776	0%
Number of 4 input LUTs	269	11776	2%
Number of bonded IOBs	81	372	21%
Number of GCLKs	1	24	4%

Detailed Reports				
Report Name	Status	Generated	Errors	Warnings
Synthesis Report	Current	Sun Dec 8 13:45:57 2013	0	5 Warnings (1 new)

Fig. 12 Device utilization of VCT router

Table 1 Comparison table of the results of VCT and XY router.

Parameters	Available			Used			Utilization (%)		
	XY	VC T	VC T + XY	XY	VCT	VC T + XY	XY	VC T	VC T + XY
Number of slices	1920	5888	14752	127	148	2407	6%	2%	16%
Number of slices flip flop	3840	11776	29504	53	52	329	1%	0%	1%
Number of 4 input LuTs	3840	11776	29504	230	267	4574	5%	2%	15%
Latency	----	----	----	20.34 1ns	5.642 ns	----	----	----	----

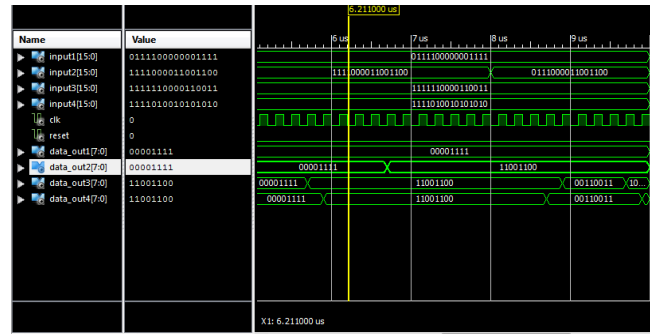


Fig. 13 Output of VCT and XY

E_HybridTB Project Status (07/12/2014 - 21:16:55)			
Project File:	Cdes.xise	Parser Errors:	No Errors
Module Name:	E_Hybrid_Wireless_Router	Implementation State:	Synthesized
Target Device:	xc3e1600e-5fg320	Errors:	No Errors
Product Version:	ISE 13.2	Warnings:	118 Warnings (0 new)
Design Goal:	Balanced	Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:	
Environment:	System Settings	Final Timing Score:	

XPS Reports			
Report Name	Generated	Errors	Warnings
Synthesis Report	Current	0	118 Warnings (0 new)

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	2407	14752	16%
Number of Slice Flip Flops	329	29504	1%
Number of 4 input LUTs	4574	29504	15%
Number of bonded IOBs	86	250	34%
Number of GCLKs	1	24	4%

Fig. 14 Device utilization of VCT with XY router

Fig. 13 shows the output of the VCT and XY router and Fig. 14 shows the number of devices utilized by the VCT and XY router to perform its task.

6. Conclusion

Table 1 shows the comparative results of the XY router, and VCT router. As shown in table 1, utilization of number of slices, number of slices flip flop and number of 4 input LuTs in XY is better than VCT. Hence from the above result we can say that XY router results are better than VCT router. The utilization of same parameters is further improved when the combination of XY and VCT router is used. Path selection can be done based on XY routing. We would be selecting a Virtual Channel to pass packets, thus the system would be having advantages of the VCT algorithm. "The authors declare that they have no competing interests."

References

- [1] V.Soteriou, R.S. Ramanujam, B. Lin, Li-Shiuan Peh. A High- Throughput Distributed Shared-Buffer NoC Router. IEEE Computer Architecture Letters, vol. 8, no. 1, pp. 21-24, Jan.-June 2009, doi:10.1109/LCA.2009.5.

- [2] A. Louri, J. Wang, Design of energy-efficient channel buffers with router bypassing for network-on-chips (NoCs)."
- [3] H. C. Freitas and C. A. P. S. Martins, "Didactic Architectures and Simulator for Network Processor Learning", Workshop on Computer Architecture Education, San Diego, CA, USA, 2003, pp.86-95
- [4] A. Kodi, A. Louri, J. Wang. Design of energy-efficient channel buffers with router bypassing for network-onchips (NoCs). Proceedings of International Symposium on Quality of Electronic Design (ISQED), pp.826-832, March 2009.
- [5] Khalid Latif, Moazzam Niazi, Hannu Tenhunen, Tiberiu Seceleanu, Sakir Sezer. Application development flow for on-chip distributed architectures. Proceedings of IEEE International SoC Conference (SOCC), Sept. 2008, pp. 163-168.