IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 1, Ver. II (Jan – Feb. 2016), PP 01-05
www.iosrjournals.org

Evaluation of NOC Using Tightly Coupled Router Architecture

Bharati B. Sayankar[1], Pankaj Agrawal[2]
[1]*Research Scholar GHRCE, Nagpur, India*
[2]*Professor GHRAET, Nagpur, India*

Abstract: *One of the most important role in many core system is played by Network on Chip (NoC). Researchers are recently focusing on the design and optimization of NoC. In this paper we describe the architecture of a tightly coupled NoC router. To improve the network performance the router uses the on-chip storage and to improve the use of on-chip resource and information, several optimizations are introduced. In theory this design can save 9.3% chip area. The experiment shows that the latency can be reduced to 75% by the process of optimization on the ejection process and energy consumption by 31.5% in heavy traffic load network. Under different buffer depth, it can also improve latency by 20% and energy consumption by 25%. The experimentally results also prove that this tightly coupled router architecture can achieve better performance in the large scale network.*

Keywords: *virtual cut through, wormhole ,latency, deadlock*

I. Introduction

NoC i.e. Network-on-chip connects a large number of cores in network style, it was first raised in 2001. It provides parallel communication to the cores. It has two characteristics; firstly, during transmission the data latency between any two cores is variable according to congestion conditions. In on-chip conditions it is unacceptable if some data is delayed for a long time. Second characteristic is that frequent communication between cores leads to heavy load or heavy wire load which brings high power dissipation. The basic function unit in on-chip network is router. A basic router consists of a buffer, routing logic, arbiter and crossbar. The received packet or flit are stored in buffer, it dependents on the flow control pattern. To reduce energy consumption and area the most direct method is to reduce buffer size. But for the most popular wormhole router, insufficient buffer leads to high latency. Therefore the mechanics of latency reduction should be carefully designed in wormhole router. In this paper we will discuss about different routing mechanisms and select the best one to make better use of the on-chip processes.

II. Proposed System

Wormhole Network

Wormhole routing is a system of simple flow control in NoC. It is based on fixed links, in the absence of blocking it takes message latency almost independent of the intermediate distance. In this network packets are broken down into small packets called flits. The header flit is the first flit, it consist of all the information about the packet`s route i.e. the destination address. It also sets up the routing behavior for all subsequent flits associated with the packet. The head flit i.e. the first flit is followed by more body flits that contain the actual pay load of data. To close the connections between two nodes, the final flit i.e. the tail flit performs the function of book keeping. The data flits are blocked behind blocking the header occupying all the channels and the

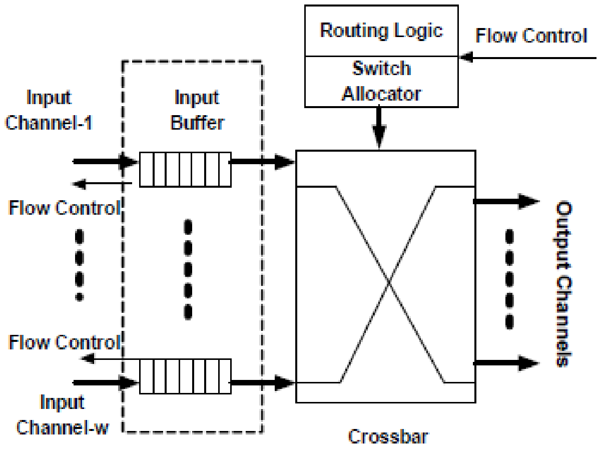
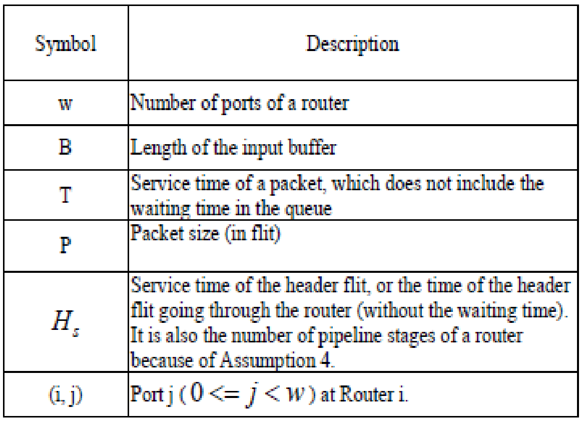


Figure 1. The structure of a router

buffer is already taken.

The process of transferring a message through a network path is defined by the routing algorithm. For any routing algorithm the key issue is deadlock free. For preventing deadlock prevention deterministic routing is widely employed. Figure 1. Shows the structure of router. According to this figure we are assuming that the wormhole router contains w ports and adopts a deterministic routing algorithm. The local port is regarded as same as others and each port is associated with a single input buffer. The abstraction of input buffers are done in groups of FIFO. To connect an input buffer of the router to any output channel, a crossbar switch is configured but under the situation that each input is connect to at least one output and vice-versa. The switch allocator resolves all the potential requests to the crossbar and other shared resources of the router. The flit level flow control is implemented in the router. The upstream router stops transmitting if the input queue of a router is full since input queue capacity is limited. The parameters of the above model is shown in the table below:

TABLE I.
PARAMETERS OF THE MODEL



Symbol	Description
w	Number of ports of a router
B	Length of the input buffer
T	Service time of a packet, which does not include the waiting time in the queue
P	Packet size (in flit)
H_s	Service time of the header flit, or the time of the header flit going through the router (without the waiting time). It is also the number of pipeline stages of a router because of Assumption 4.
(i, j)	Port j ($0 <= j < w$) at Router i.

Virtual Cut-Through

A message arriving at an intermediate node is immediately forwarded to the next node on the path without buffering, provided that a channel to the next node is available. This is referred to as the cut-through operation. When the header arrives with the destination information a cut through can be performed at the intermediate node. If cut-through are established between all intermediate nodes then a circuit is established to the destination. If the request outgoing channel is unavailable the message header is blocked at an intermediate node and the message is completely buffered at the current node. In this cut-through when a packet cannot cut-through an immediate node, the packet is buffered at the node instead of being kept in the network. This helps it to achieve high performance.

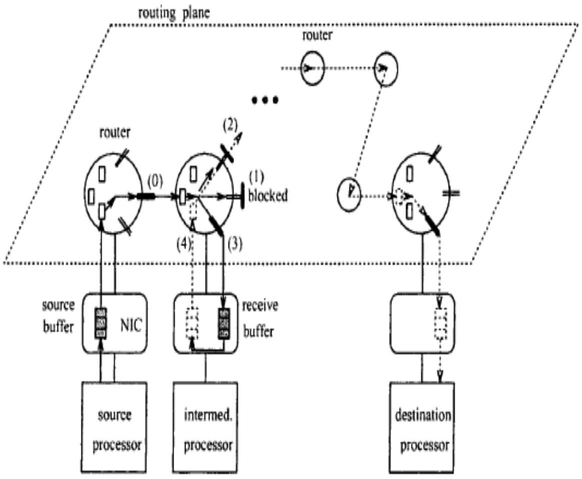


Figure 2: General switching model

The Figure 2 shows a general switching model. When an incoming channel receives the head of a packet, the destination node information is extracted from the header. A connection with the first choice outgoing channel is attempted. If the attempt is successful then a cut-through is established through the current node, it is denoted by (0) in the Figure 2. But when the requested outgoing channel (1) is blocked since it is used by another packet, then attempt for connection is made on the second-choice outgoing channel (2). The process is repeated until the connection is made or all the outgoing channels are exhausted. If all the channels are exhausted then the blocked packet remains in the network until a channel becomes available or till the permitted waiting time expires. If the time expires the packet is taken to the source buffer of the current node just as if the packet was originated there. With the help of a specially designed network interface controller the ejection and the reinjection latency into the network for blocked messages in the intermediate nodes can be alleviated.

Dynamic Priority Round Robin Algorithm

The round robin algorithm can be explained in the following steps:

Step 1: A queue of ready processes is maintained by the scheduler along with the list of blocked and swapped out processes.

Step 2: The process control block of newly created process is added to the end of ready queue. The process control block of the terminating structure is removed from the scheduling data structure.

Step 3: The selection of the process control block by the scheduler is done from the head of the ready queue.

Step 4: A running process is moved to end of ready queue when it finishes its time slice.

Step 5: The following actions are performed by the event handler:

a) When an input-output request is made by a process or it is swapped out then the process control block of this process is removed from the ready queue to blocked/swapped out list.

b) When I/O operations awaited by the process is swapped on its process control block or a process is finished, it is put into the end of ready queue.

But there are a few disadvantages of round robin algorithm like static tie quantum, larger waiting time and response time, larger number of context switches and low throughput.

III. Implementation

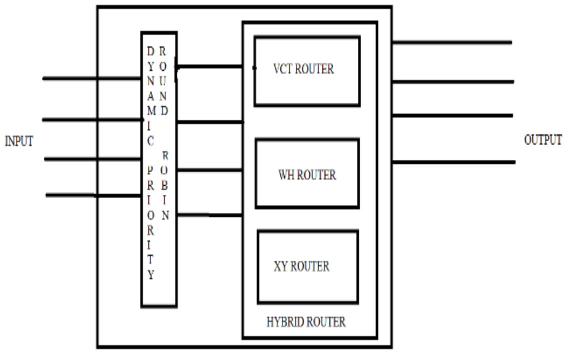


Figure 3: Implemented on chip router

The Figure 3 shows the on chip router that is to be designed. Here a hybrid router is created out of VCT, WH and XY router. The input to it is given using the Dynamic Priority Round Robin.

Here the input request is categorized in priority and is then sent to the routers accordingly. If A and B are two inputs and A has a higher priority than B then A is given first priority to use the resources. If such a case occurs that both the inputs have same priority then the round robin is applied.

Here we will be dividing the input stream into multiple super packets. Also we would be sending one super packet along with a single calculated path, and then we will make the path free for other transmissions, this path selection is done using the XY or YX algorithm.

Here a Virtual channel is selected to pass packets, that is the system will have the advantages of VCT algorithm. Since one upper packet is made up of multiple packets and we are transmitting one super packet in a single transaction the route is busy for that time, this scenario indicates the presence of the WH algorithm. Here the implemented result contains the hybrid of all the three algorithm. The generated algorithm reduces the latency and improves throughput, the area of operation is also reduced.

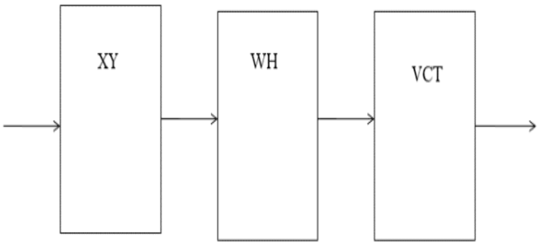


Figure 4: Routing Technique

As shown in Figure 4 the hybrid router is a mixture, therefore in it the VCT algorithm is used for finding the shortest path, The XY algorithm is used for sending the packets in same direction and the WH algorithm is used for sending maximum number of packets at a same time.

IV. Result

Table II & III shows the comparison of various parameters XY,WH,VCT with Dynamic Priority Round Robin Algorithm.

Table II

PARAMETERS	**AVAILABLE**	**XY**	**WH**	**VCT**	**VCT +XY**
Number of Slices	28800	4%	0%	1%	0%
Number of slice LUTs	28800	7%	1%	1%	1%
Number of LUT-FF Pairs	-----	---	15%	30%	20%
Latency	-----	3.920ns	2.525ns	2.826ns	5.600 ns
Route Latency	-----	1.294ns	1.871ns	0.286ns	3.174 ns
Power	-----	11.24 mW	15.65 mW	10.56 mW	12.66 mW
Throughput	-------	32.65 GbPS	50.693 GbPS	45.293 GbPS	22.85 GbPS

Table III

PARAMETERS	**AVAILABLE**	**VCT+WH+XY**	**VCT+WH+XY With DPRRRC**
Number of Slices	28800	0%	0%
Number of slice LUTs	28800	6%	2%
Number of LUT-FF Pairs	-----	4%	15%
Latency	-----	4.487ns	4.677ns
Route Latency	-----	3.447ns	3.006ns
Power	-----	11.44 mW	9.65 mW
Throughput	-----	28.526 GbPS	27.367 GbPS

V. Conclusion

The comparison of XY, WH, VCT was done and on comparison of the outputs VCT was found to be the better algorithm. The results using hybrid VCT algorithm were also evaluated. These results using hybrid VCT with DPRRRC were determined effectively. The results of hybrid VCT and hybrid VCT with DPRRRC were compared and route latency, power and throughput were found to be decreased.

References

[1] J. D. Davis, J. Laudon, and K. Olukotun, “Maximizing cmp throughput with mediocre cores,” in PACT ’05: Proceedings of the 14th International Conference on Parallel Architectures and Compilation Techniques. Washington, DC, USA: IEEEComputer Society, 2005, pp. 51–62.

[2] N. Kirman, M. Kirman, R. K. Dokania, J. F. Martinez, A. B. Apsel, M. A. Watkins, and D. H. Albonesi, “Leveraging optical technology in future bus-based chip multiprocessors,” in MICRO 39: Proceedings of the 39th Annual IEEE/ACMInternational Symposium on Microarchitecture. Washington,DC, USA: IEEE Computer Society, 2006, pp. 492–503.

[3] C. A. Nicopoulos, D. Park, J. Kim, N. Vijaykrishnan, M. S. Yousif, and C. R. Das, “Vichar: A dynamic virtual chan-nel regulator for network-on-chip routers,” in MICRO 39:Proceedings of the 39th Annual IEEE/ACM International Symposium on Microarchitecture. Washington, DC, USA:IEEE Computer Society, 2006, pp. 333–346.

[4] X. Chen and L.-S. Peh, “Leakage power modeling and optimization in interconnection networks,” in ISLPED ’03:Proceedings of the 2003 international symposium on Low power electronics and design. New York, NY, USA: ACM,2003, pp. 90–95.

[5] T. T. Ye, G. D. Micheli, and L. Benini, “Analysis of power consumption on switch fabrics in network routers,” in DAC’02: Proceedings of the 39th conference on Design automa-tion. New York, NY, USA: ACM, 2002, pp. 524–529.