Scientia Iranica, Vol. 15, No. 6, pp. 579–588

SCIENTIA
IRANICA

Clock Boosting Router: Increasing the Performance of an Adaptive Router in Network-on-Chip (NoC)

S.E. Lee[1] **and N. Bagherzadeh**[1,*]

In this paper, a simple and efficient clock boosting mechanism to increase the performance of an adaptive router in Network-on-Chip (NoC) is proposed. One of the most serious disadvantages of a fully adaptive wormhole router is performance degradation due to the routing decision time. The key idea to overcome this shortcoming is the use of different clocks in a head flit and body flits. The simulation results show that the proposed clock boosting mechanism enhances the performance of the original adaptive router by increasing the accepted load and decreasing the average latency in the region of effective bandwidth. The enhanced throughput of a router results in power saving by reducing the operating frequency of a router for certain communication bandwidth requirements.

Keywords: Network-on-Chip (NoC); Interconnection network; Wormhole flow control; Adaptive router; Dynamic Frequency Scaling (DFS); Low power design.

INTRODUCTION

Recently, an on-chip interconnection network, Network-on-Chip (NoC), using a packet switching technique has been proposed to accommodate the trend for System-on-Chip (SoC) integration [1,2]. An example of the NoC architecture is shown in Figure 1. The basic idea is the use of a packet switching technique that has been extensively used for computer networks. NoC is a promising solution to the communication challenges of on-chip interconnection, featuring the requirements of the next-generation multiprocessor systems.

In this paper, a simple and efficient mechanism to increase performance of an adaptive router in NoC is proposed. One of the most serious disadvantages of a fully adaptive wormhole router is its performance degradation, due to the route decision time. To overcome this weak point, the use of a faster clock during the service of the body flit is proposed [3].

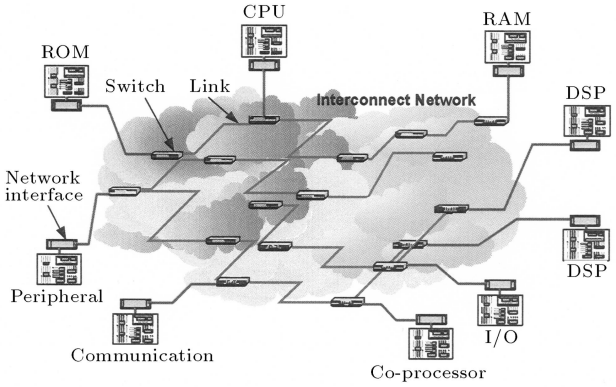

Figure 1. Example of network-on-chip architecture.

It will be shown that throughput can be enhanced, with respect to the conventional adaptive routers, while keeping the same functionality. In order to show the feasibility of the mechanism, a detailed hardware implementation will be presented.

The major contributions of this paper are:

1. Proposal of a simple and efficient mechanism to improve performance of fully adaptive wormhole routers.
2. Quantitative evaluation of the proposed mechanism

1. *Department of Electrical Engineering and Computer Engineering, University of California at Irvine, Irvine, CA, USA.*
*. *To whom correspondence should be addressed: nader@uci.edu*

showing that the proposed router can support higher throughput than a conventional one.

3. An evaluation of hardware overhead for the proposed router.
4. Proposal of power management schemes for an on-chip interconnection network.

In the following sections, first, the motivation for this research and an informal and intuitive description of the proposed mechanism are presented. Then, the baseline router that is used to evaluate the performance of the proposed mechanism is described in detail. After that, the implementation of the proposed mechanism and the experimental results are presented, respectively, and the previously published works are addressed. Finally, conclusions are drawn.

MOTIVATION

Recently, a number of research projects have shown how to develop network architectures appropriate for on-chip environments. Different routing algorithms, such as deterministic/oblivious and adaptive routing algorithms, have been proposed. Because of simplicity and ease of analysis, many researchers adopted deterministic/oblivious algorithms, such as DOR [4], ROMM [5] and O1TURN [6]. Some researchers have developed better performance routing algorithms even using adaptive routing algorithms [7-13]. Recently there have been several implementation related works using deterministic routing algorithms, as well as adaptive routing ones [14,15].

The design of a high performance router for on-chip interconnection has tight resource constraints, such as the router's area, power and speed. Adaptive routing has been proposed as a method of improving network utilization by using information about the network state to select a path among alternative paths to deliver a packet, potentially reducing network latency. However, while a good adaptive routing algorithm can balance network occupancy and enhance its maximum throughput, it also suffers from high costs in terms of additional sophisticated logic and performance degradation, due to the routing decision time.

Wormhole flow control has increasingly been advocated as a means of reducing latency. It reduces latency by routing a packet as soon as its head flit arrives at a node. The routing of a head flit enables the switches to establish the path, and body flits are simply forwarded along the path in a pipelined fashion. However, wormhole flow control has some disadvantages, for instance, a router stops a packet when its head flit is blocked. When the link requested by a head flit is busy, the head flit can not advance and the remainder of the flits is also stopped, holding the buffers and channels along the path that has already formed, leading to significant link congestion.

One of the most serious drawbacks of an adaptive wormhole router is the performance degradation, due to the routing decision time, because routing flexibility requires additional resources. The goal is to design a fast fully adaptive router, which provides more throughput than current wormhole routers.

In this paper, the use of multiple clocks to implement an adaptive wormhole router is investigated and it is shown how this technique reduces latency, greatly increases throughput of an adaptive router, and reduces power consumption.

CLOCK BOOSTING ROUTER

In this section, a mechanism to enhance the throughput of an adaptive wormhole router is presented, and an informal and intuitive description of this design is described.

Mechanism

In an adaptive wormhole router, the routing decision time for the head flit is the critical path of an adaptive router, restricting the operating frequency of the overall router. While the head flit requires support of complex logic that can impact the decision path, body flits can continue advancing along the reserved path that is already established by the head flit.

The key idea of this paper is the use of different clocks in a head flit and body flits, because the body flits can be forwarded immediately without any computation and the FIFO usually operates much faster than route decision logic in an adaptive router. Figure 2 explains the mechanism for the proposed router. The conventional wormhole flow control uses the same clock to advance the head flit and body flits. Generally, the operating frequency is defined by the routing decision logic for the head flit, therefore, this technique uses a faster clock to advance body flits. It reduces the average latency of a packet, as well as the contention, by compacting the effective length of body flits in the time domain.

Analysis of the Mechanism

The performance of an interconnection network can be described by latency versus offered traffic curves. Although latency versus offered traffic curves give the most accurate view of the ultimate performance of an interconnection network, they do not have simple, closed form expressions and are generally evaluated by discrete-event simulation [16].

Zero-load latency gives a lower bound for the average latency of a packet through the network by

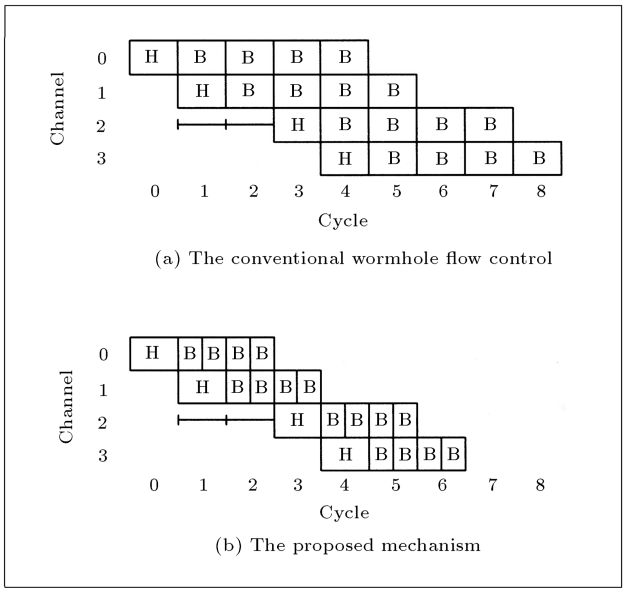

Figure 2. Time-space diagram showing wormhole flow control sending a 5-flit packet over 4 channels. (a) The conventional wormhole flow control; (b) The proposed mechanism.

assuming that a packet never contends for network resources with other packets. The zero-load latency of a packet using wormhole routing is:

$$T_0 = Ht_r + L/b, \tag{1}$$

ignoring wire latency. The first term reflects the time required for the head flit to traverse the network, with an average hop count of H and a delay of t_r, through a single router. The second term is serialization latency, the time for a packet of length L to cross a channel with bandwidth b. The proposed mechanism reduces the serialization latency by boosting clock frequency during the body flit transfer. The boosting of clock frequency results in increasing the bandwidth and reduces the zero-load latency. Average latency versus offered traffic curves are obtained by the discrete event simulation in the experimental results sections, demonstrating performance improvement with the proposed mechanism.

BASELINE ADAPTIVE ROUTER

This technique uses a wormhole switching with a deadlock- and livelock- free algorithm for 2D-mesh topology. The routing algorithm was compared with DOR [4], ROMM [17] and O1TURN [18] algorithms in [19]. The router models were written in SystemC and simulations were executed with different traffic patterns. The routing algorithm showed the same or better performance for all traffic patterns in 4×4 mesh topology. For every traffic pattern, it sustained the highest offered traffic amount with the lowest average latency. Though ours had a slightly lower performance than O1TURN at 8×8 mesh topology, it still showed competitive performance. However, at the given amount of offered traffic before the saturation point, it demonstrated the best performance, with respect to average latency. Moreover, the bandwidth and the total area overhead of the router enabled the router as a feasible alternative to existing routers for NoC. In this paper, the adaptive router [20] is used as a baseline router and the throughput is improved by adopting the proposed clock boosting mechanism.

Overview

The network topology in 2D-mesh is assumed, which has $N \times M$ routers. The overall block diagram of a single router is shown in Figure 3. There is an input FIFO queue per each input channel and each output port has an associated arbiter to choose the proper packet among the given incoming packets from each candidate input port. A router with seven interfaces, suitable for a 2D-mesh is considered, which includes interface to an integrated Processing Element (PE). It is assumed that a packet coming through an input port does not loop back, thus, each input port is connected to four output ports.

The router is composed of three architectural blocks: Right Router, Left Router and Internal Router. The Right Router serves the port set {W-in, N1, E-out, S1}. On the contrary, the Left Router serves the port set {E-in, N2, W-out, S2}. The Internal Router supports the additional interface to an integrated processing element. The separated routing paths for a vertical direction and the unidirectional path for a horizontal direction allow the network to avoid cycles in its channel-dependency-graph, resulting in a dead lock-free operation [7,8,21]. Also, by choosing the shortest path in routing, a livelock free operation is guaranteed [16].

Packet Format

Figure 4 shows the packet format. Each packet has the DestPE_addr field to indicate the destination node in the head flit. The address of the destination node is represented by the relative distance of the horizontal and vertical direction. A positive value represents the southern and eastern direction in a vertical and horizontal direction, respectively. Each relative distance is signed magnitude value, i.e. the MSB of each X-dir and Y-dir field represents its sign and the rest of the bits represents its magnitude. For instance, if destPE_addr has $0x91$ in hexadecimal format, it represents that the destination node is located at western 1 hop and southern 1 hop from the current node. Its vectorized representation in $X - Y$ coordinate is (-1, 1). The

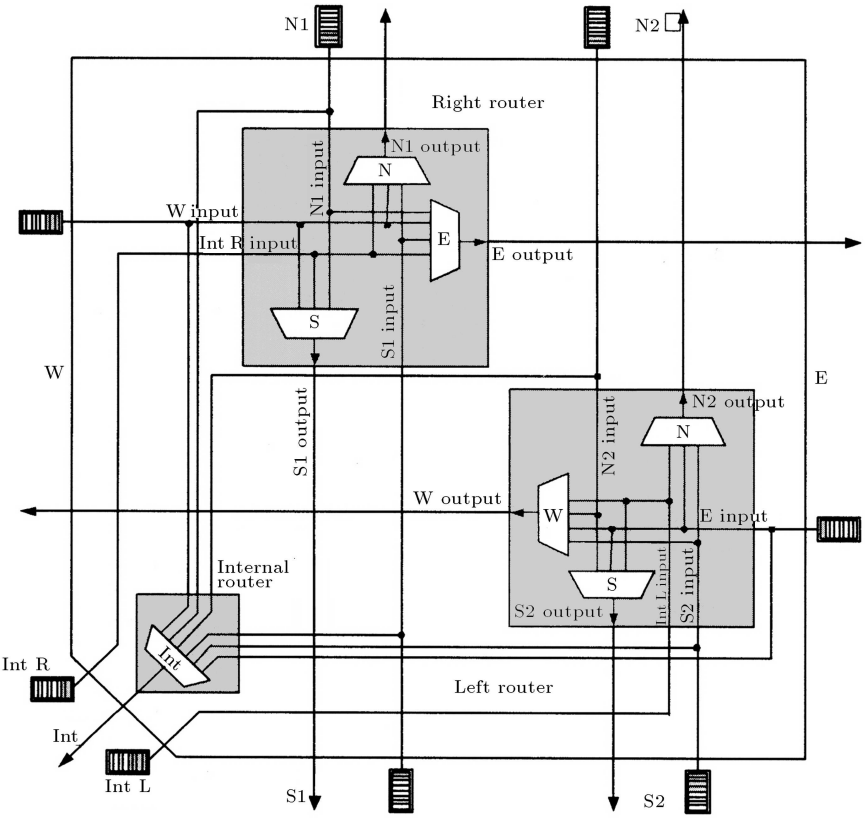

Figure 3. Adaptive router architecture.

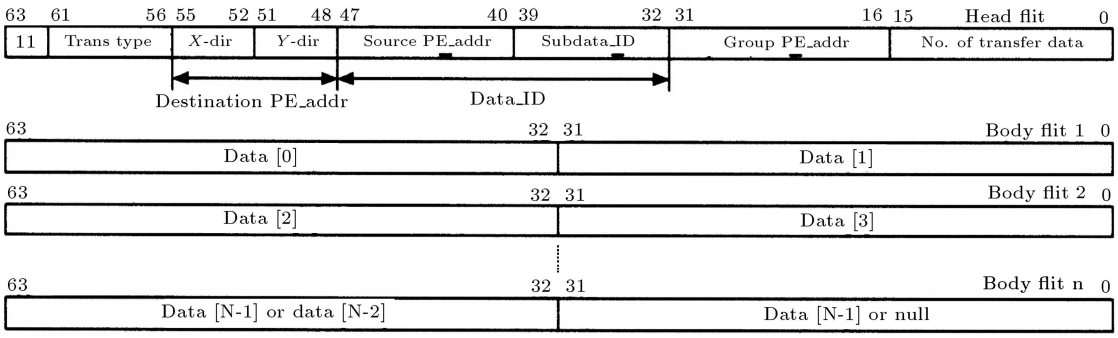

Figure 4. Packet format.

head flit also has the No_of_data field to represent the number of body flits followed by the head flit.

Priority

For outgoing channel allocation, the router applies a fixed priority scheme to the incoming packets that have reached the corresponding node simultaneously. For each outgoing channel, the possible incoming channels have a descending order of priority in a clock-wise direction. The incoming packets and output port of an internal router have the lowest priority. Table 1 summarizes the priority for outgoing channel allocation. Similarly, each output port has a descending order of priority in a clock-wise direction from N1.

Router Architecture

The detailed block diagram of the Right Router is shown in Figure 5. The routers for each output port are placed according to their priority level from the highest router (N1) to the lowest router (S1). Each incoming packet is directed to the Header Parsing Unit (HPU) per each output port. The HPU generates a set of possible incoming packets, which could be routed to the corresponding output port, in order of the input

Table 1. Priority for outgoing channel allocation.

Router	Outgoing Port	Order of Priority on Incoming Ports
Right	N1-out	S1-in, W-in, IntR-in
	E-out	S1-in, W-in, N1-in, IntR-in
	S1-out	W-in, N1-in, IntR-in
Left	N2-out	E-in, S2-in, IntL-in
	S2-out	N2-in, E-in, IntL-in
	W-out	N2-in, E-in, S2-in, IntL-in
Internal	Int-out	N1-in, N2-in, E-in, S1-in, S2-in, W-in

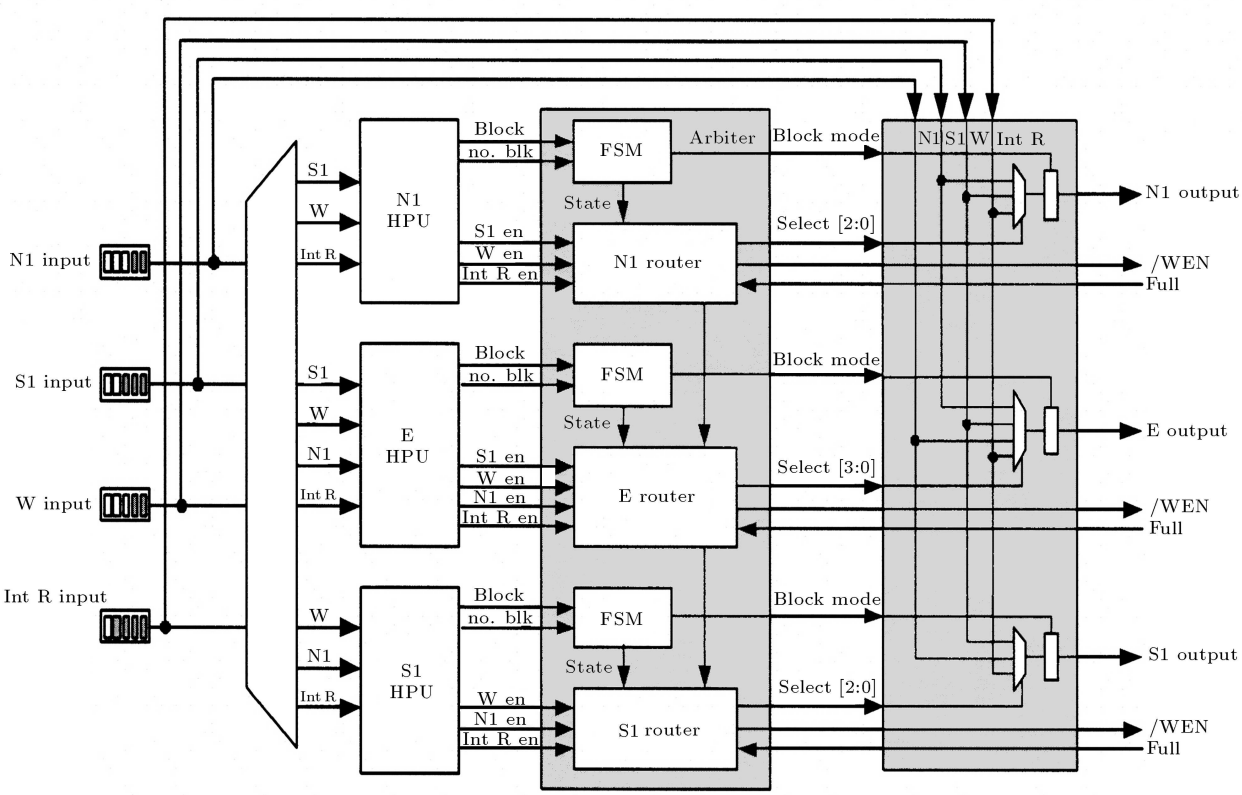

Figure 5. Micro-architecture of the right/left router.

priority level, by looking up the destination address in the head flit.

When the output port is available (by referring FULL signal), the router chooses the input packet for the corresponding output port among the set of routable incoming entries provided by the HPU. If two or more packets arrive simultaneously, the arbiter will choose a packet according to this priority.

If one output port is granted to one head flit, it reserves the output port until all body flits are forwarded. To support this feature, the router has a Finite State Machine (FSM) in each output port, which stores the state of the corresponding router. Also, it counts the number of forwarded flits, in order to record the remaining number of the body flits that need to be forwarded. Therefore, each incoming packet has its own path, in order to forward the body flits to its selected output port. The body flits arriving along them will reach the router output without passing through the Arbiter unit.

After completing the path decision for the highest router, the lower prioritized router refers the decision result of the highest router and disables the incoming port, which was served by the former router, from the set of possible incoming packets for the router. In this manner, the arbitration is done by propagating the routing decision from the highest router to the

lowest router. Therefore, all incoming packets could be advanced in their output port at once.

Finally, the multiplexer unit maps corresponding incoming packet to the output port by referring the SELECT signal. It also updates the destination address in the head flit in order to complete proper transmission;

1. Head flit: Decrease the corresponding destination address,
2. Body flits: Bypass the incoming flits.

In order to achieve high performance, all routing decisions are made within one clock cycle. So, an incoming flit can advance in one clock cycle if the corresponding output port is free.

CLOCK BOOSTING ADAPTIVE ROUTER DESIGN

Figure 6 shows the block diagram of the improved router. The hardware cost of the proposed mechanism is a multiplexer in each channel for switching the clock domain and the use of a FIFO supporting dual clock operation. The clock multiplexer is in charge of switching clock domain, according to the service flit, and could also drive the corresponding load with appropriate fan out. A multiple clock scheme is used to boost clock speed for the body flits, in order to simplify the implementation. As a test case, a two times faster clock (2x) and four times faster clock (4x) are used, since the original router and the FIFO can operate at 423 MHz and 1.8 GHz, respectively.

The clock is the most important and sensitive signal in a system and the glitches between clock domain transitions make the system unstable causing erroneous signals. To ensure consistency of clock phase during clock domain changes between original clock (1x) and boosting clock (2x and 4x), a constraint in body flit size is added. That is, in a case of using a 2x boosting clock, the length of body flits is limited to a multiple of two and in the case of using a 4x boosting clock, the length of body flits is limited to a multiple of four.

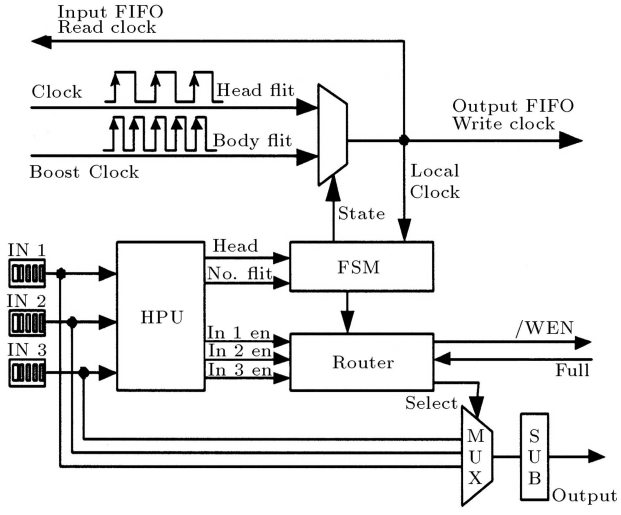

Figure 6. Micro-architecture of the modified router.

A logic description of the router's component has been obtained using synthesis tools from the Synposys™, using TSMC™ 90 nm technology. While the results are obtained by simulation, it provides one with the physical design characteristics of the proposed routers. Table 2 summarizes the physical characteristics of the routers. The additional area cost in the clock boosting router is basically due to the clock multiplexer for each channel. Table 3 describes the physical characteristics of the original FIFO and modified FIFO that support dual clock operation. In the SoC layout, wire lengths are very non-uniform, due to the irregular floor plan. While the wire length can be very long, leading to timing and clocking issues, this system assumes a regular mesh network and a well segmented layout of PEs. Therefore, this physical characteristic of the single router is well suited for NoC.

The overall router, including the input FIFOs with size 8, occupies an area of approximately 0.157 mm^2 (Router Area + FIFO Area × 8) using a 90 nm technology. The ARM11 MPCore™ and PowerPC™ that provide multi CPU designs, occupies 1.8 mm^2 and 2.0 mm^2 in 90 nm technology, respectively [22,23]. If the router was integrated as an on-interconnection network, the area overhead imposed by the network would be reasonable, showing the feasibility.

Table 2. Physical characteristics of the router.

	Router 1x	**Router 2x**	**Router 4x**
Voltage	1.0 V	1.0 V	1.0 V
Frequency	423 MHz	421 MHz	421 MHz
Area	17,537 μm^2	17,821 μm^2	17,830 μm^2
Dynamic Power	2.47 mW	3.30 mW	5.73 mW
Leakage Power	171.94 μW	175.43 μW	175.69 μW

Table 3. Physical characteristics of the FIFO with depth 8.

	FIFO Single Clock	**FIFO Dual Clock**
Voltage	1.0 V	1.0 V
Frequency	1.81 GHz	2.32 GHz/write 1.68 GHz/read
Area	17,428 μm^2	14,412 μm^2
Dynamic Power	10.12 mW	10.11 mW
Leakage Power	160.77 μW	160.34 μW

EXPERIMENTAL RESULTS

Evaluation Methodology

In order to evaluate the performance of the proposed clock boosting mechanism, the router has been developed written in Verilog™ HDL. For the measurement of throughput and adjusting incoming traffic, a standard interconnection network measurement setup [16] has been adopted, where the packet generation is placed in front of an infinite depth source queue and the input timing of each packet is measured whenever it is generated. In this simulation, the number of body flits in a packet was fixed to 8, even though the defined packet format supports a various number of flits. Also, the depth of a FIFO between each link is fixed to 8 in this simulation. In order to achieve fair comparison between the original router and the proposed router, it is assumed that the operating frequency of each processing element is the same as the operating frequency of the original router. Thus, the input packet is injected in each cycle of the processing element's operation frequency, during the simulation of the boosting router, featuring the same traffic load as the original router.

Recently, several specific communication patterns between pairs of nodes have been used to evaluate the performance of interconnection networks. These communication patterns take into account the permutations that are usually performed in parallel numerical algorithms [16,24]. In this paper, the simulation is completed using four different traffic patterns, such as uniform random, matrix transpose, bit-complement and bit-reverse traffics in 4×4 and 8×8 mesh networks. Even though these traffic patterns cannot realistically reflect the type of real traffic that will traverse the network, they are generally used to evaluate the performance of a network, providing a reasonable measurement of network performance.

Performance on Throughput/Latency

The simulation is completed using four different traffic patterns in 4×4, and 8×8 mesh networks (see Figure 7). Each graph represents offered traffic (flit/node/cycle) in a X-axis and average latency (cycles) in a Y-axis.

For the 4×4 network, the adaptive router, with a 4x boosting clock, makes the phase transition region shift from 0.26 flit/cycle to 0.7 flit/cycle with uniform random traffic, from 0.36 flit/cycle to 0.8 flit/cycle with bit-complement traffic, from 0.6 flit/cycle to 1.0 flit/cycle with matrix transpose traffic, and from 0.45 flit/cycle to 1.0 flit/cycle with bit-reverse traffic. Similar improvements have been obtained for the 8×8 mesh network. In this case, the phase transition region shifts from 0.12 flit/cycle to 0.32 flit/cycle with uniform random traffic, from 0.14 flit/cycle to 0.36 flit/cycle with bit complement traffic, from 0.28 flit/cycle to 0.64 flit/cycle with matrix transpose traffic, and from 0.18 flit/cycle to 0.54 flit/cycle with bit-reverse traffic.

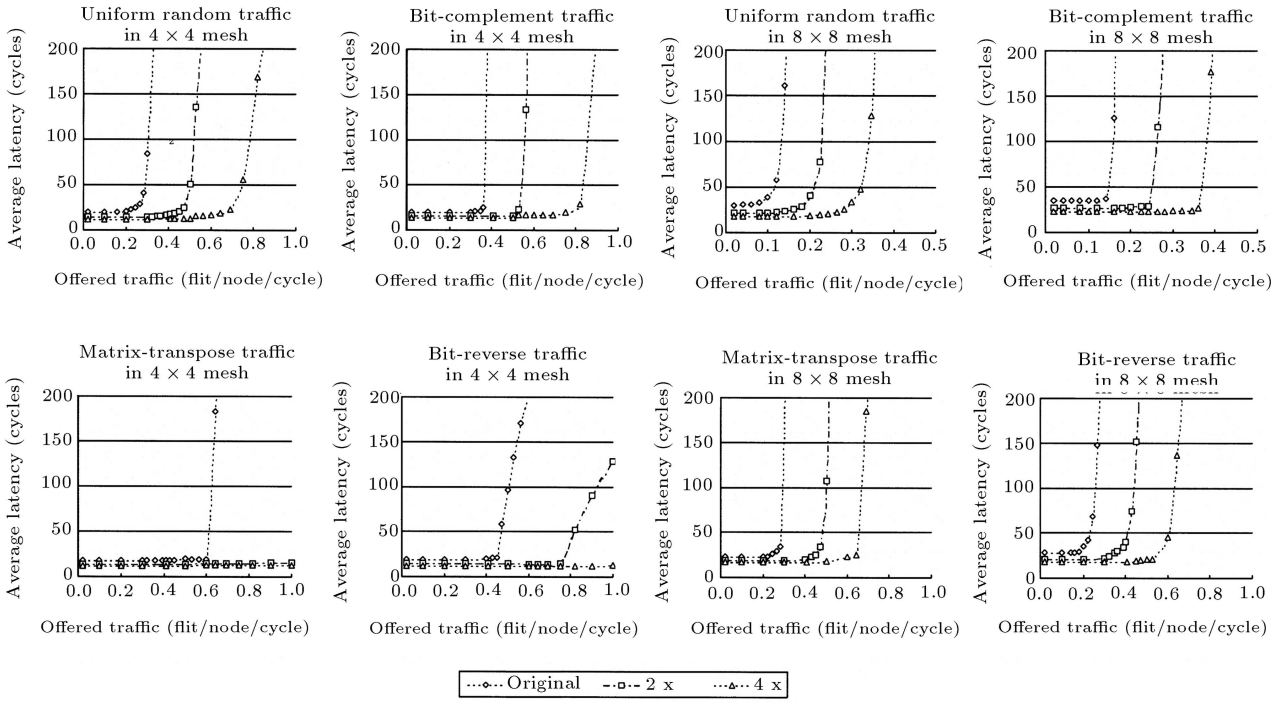

Figure 7. Average latency versus offered traffic curve.

The critical traffic load grows with the boosting clock frequency, thus, utilizing the link bandwidth that is wasted in the original router.

Table 4 describes the zero-load latency calculated by Equation 1 and average latency with 0.02 flit/cycle offered traffic obtained by simulation in the case of a uniform random traffic pattern. The difference between ideal zero-load latency and simulation results is because of contention in the network and the input injection assumptions. While the flits were injected with a 0.02 flit/cycle, it could result in a contention situation. In this simulation, it is assumed that the operating frequency of each processing element is the same as the operating frequency of the router (1x). So, in the case of a boosting operation, the input body flit could not be injected into the incoming port, resulting in added latency. Although there exists a small difference between them, the experimental results show the superiority of the proposed mechanism in the range of a low data rate.

In summary, the simulation results show that the proposed clock boosting mechanism enhances performance of the original adaptive router by increasing throughput and decreasing the average latency in the region of the effective bandwidth.

Power Saving with Clock Boosting Router

The power consumption of the interconnection network was extracted using Synposys^{TM} PrimePower^{TM}. In this analysis, the power consumption under the uniform random traffic pattern is investigated, because the uniform random traffic pattern shows worst performance among the four traffic patterns (see Figure 7).

The enhanced throughput of the clock boosting router can reduce the power consumption in an interconnection network. Under the given traffic requirements, that is usually defined by the application, the operating frequency of the router could be reduced in the clock boosting router, decreasing power consumption. For instance, if an application generates a communication traffic with maximum 4 Gbps for each PE node, the original router should operate with more than 364 MHz. However, the clock boosting router could support the traffic with 200 MHz and 145 MHz operations in a 2x and 4x clock boosting router, respectively, reducing about 30% of the power consumption. Table 5 summarizes the power consumption in a 4×4 mesh network, when each PE generated 4 Gbps traffic.

Table 4. Zero-load latency and average latency with 2% flit/cycle offered traffic (cycles) with uniform random traffic.

Boosting Router	4×4 Mesh Network		8×8 Mesh Network	
	Zero	2%	Zero	2%
Original (1x)	15.55	19.54	20.13	30.33
Boost (2x)	11.55	14.72	16.13	21.71
Boost (4x)	9.55	12.28	14.13	17.44

Clock Boosting Router for DFS Links

The link utilization in an interconnection network has wide variance and fine tuning the network bandwidth can lead to considerable power saving by applying low operating frequency for under utilized links, still maintaining the required performance. Dynamic Voltage Scaling (DVS) has been used to reduce the power consumption of on-chip interconnection network [25,26] varying the link frequency and voltage in run time. However, current DVS techniques require thousands of clock cycles to shift between the voltage levels, limiting their ability to respond to high frequency changes in network bandwidth demands [27-29]. Moreover, the DVS links have an additional frequency synthesizer and adaptive power supply regulator in links, adding to hardware complexity.

Dynamic Frequency Scaling (DFS) only adapts the system clock frequency by setting all links in the network to the same voltage. The clock boosting router provides a variable frequency link that is applicable for DFS, with negligible hardware cost and fast response time to the frequency changes. Table 6 describes the DFS link characteristics of a 4×4 mesh network

Table 5. Power consumption with 4 Gbps traffic for each PE in a 4×4 interconnection network.

Boosting Router	Normal Freq.	Boosting Freq.	Throughput in Each PE	Power
Original (1x)	364 MHz	364 MHz	4.01 Gbps	1.03 mW
Boosting (2x)	200 MHz	400 MHz	4.04 Gbps	0.72 mW
Boosting (4x)	145 MHz	518 MHz	4.02 Gbps	0.79 mW

Table 6. DFS link characteristics with 100 MHz operation in a 4×4 interconnection network.

Boosting Router	Boosting Freq.	Offered Traffic	Throughput in Each PE	Power
Original (1x)	100 MHz	22.5%	1.10 Gbps	0.30 mW
Boosting (2x)	200 MHz	45 %	2.02 Gbps	0.37 mW
Boosting (4x)	400 MHz	69.2 %	2.96 Gbps	0.55 mW

that adopts the clock boosting router with 100 MHz operating frequency under the random traffic pattern. The use of a 2x and 4x boosting clock provides more bandwidth in each PE at the expense of more power consumption. The DFS link, using a clock boosting router, supports three clock frequency levels. However, Lu et al. [30] showed that four discrete frequencies are sufficient to achieve nearly maximum energy saving. Hsu [31] saved 30% power consumption in the MPEG core by applying a DFS power management mechanism, using only three frequency levels (25, 50, and 100 MHz), supporting the authors claim that the clock boosting mechanism provides a DFS link for an on-chip interconnection network with simple and easy implementation.

RELATED WORKS

Improving network performance has power saving potential for an NoC. For instance, Express Cube [32] lowers network latency by reducing average hop counts. The main idea is to add extra channels between non-adjacent nodes, so that packets spanning long source-destination distances can shorten their network delay by travelling mainly along these express channels, reducing the average hop count. Besides its performance benefit, an Express Cube can also reduce network power, since it reduces hop count effectively, resulting in complete removal of intermediate router energy [33]. The enhanced throughput of a switch can result in power saving by reducing the operating frequency of a switch for certain communication bandwidth requirements that are usually defined by an application. A speculative virtual-channel router [34] optimistically arbitrates the crossbar switch operation, in parallel with allocating an output virtual channel. This speculative architecture largely eliminates the latency penalty of using virtual-channel flow control, having the same per-hop router latency as a wormhole router, while improving the throughput of a router. Express virtual channels [35], which use virtual lanes in the network to allow packets to bypass nodes along their path in a non-speculative fashion, reducing delay and energy consumption.

CONCLUSIONS

In this paper, a simple and efficient mechanism to increase the performance of an adaptive router in NoC is introduced. One of the most serious disadvantages of fully adaptive wormhole routers is their performance degradation, due to the route decision time. The key idea to overcome this shortcoming is the use of different clocks for a head flit and body flits, because the body flits can be forwarded immediately and the FIFO usually operates faster than route decision logic in an adaptive router.

This clock boosting mechanism was implemented for the adaptive wormhole router and the performance evaluation was completed using simulation. The simulation results demonstrated the performance enhancements in terms of throughput, average latency in the range of accepted traffic load, and power consumption. Moreover, the proposed router was synthesized using 90 nm technology and proved the feasibility of this mechanism for NoC design. This mechanism could be applicable for conventional adaptive routers with wormhole flow control.

ACKNOWLEDGMENT

This research was supported by a fellowship from the Center for Pervasive Computing and Communication (CPCC).

REFERENCES

1. Dally, W. and Towles, B. "Route packets, not wires: On-chip interconnection networks", in *Design Automation Conference*, pp. 684-689 (2001).
2. Benini, L. and Micheli, G.D. "Networks on chips: A new SoC paradigm", *IEEE Computer*, **35**(1), pp. 70-78 (2002).
3. Lee, S.E. and Bagherzadeh, N. "Increasing the throughput of an adaptive router in network-on-chip (noc)", in *CODES+ISSS '06: Proceedings of the 4th International Conference on Hardware/Software /Codesign and System Synthesis*, New York, NY, USA: ACM, pp. 82-87 (2006).
4. Sullivan, H. and Bashkow, T.R. "A large scale, homogeneous, fully distributed parallel machine, i", in *ISCA '77: Proceedings of the 4th Annual symposium on Computer Architecture*, pp. 105-117 (1977).
5. Nesson, T. and Johnsson, S.L. "ROMM routing on mesh and torus networks", in *Proc. 7th Annual ACM Symposium on Parallel Algorithms and Architectures SPAA'95*, pp. 275-287 (1995).
6. Seo, D., Ali, A., Lim, W.-T., Rafique, N. and Thottethodi, M. "Near-optimal worst-case throughput routing for two-dimensional mesh networks", in *ISCA '05: Proceedings of the 32nd Annual International Symposium on Computer Architecture*, pp. 432-443 (2005).
7. Dally, W. and Seitz, C.L. "Deadlock-free message routing in multiprocessor interconnection networks", *IEEE Trans. Computs.*, **C-36**(5), pp. 547-553 (1987).
8. Duato, J. "A new theory of deadlock-free adaptive routing in wormhole networks", *IEEE Trans. Parallel Distrib. Syst.*, **4**(12), pp. 1320-1331 (1993).
9. Glass, C.J. and Ni, L.M. "Adaptive routing in mesh-connected networks", in *International Conference on Distributed Computing Systems*, pp. 12-19 (1992).

10. Glass, C.J. and Ni, L.M., "The turn model for adaptive routing", *J. ACM (JACM)*, **41**(5), pp. 874-902 (1994).

11. Puente, V., Beivide, R., Gregorio, J.A., Prellezo, J.M., Duato, J. and Izu, C. "Adaptive bubble router: A design to improve performance in torus networks", in *International Conference on Parallel Processing*, pp. 58-67 (1999).

12. Chiu, G.-M. "The odd-even turn model for adaptive routing", *IEEE Trans. Parallel Distrib. Syst*, **11**(7), pp. 729-738 (2000).

13. Hu, J. and Marculescu, R. "DyAD: smart routing for networks-on-chip", in *Proceedings of the 41st Annual Conference on Design Automation (DAC-04)*, pp. 260-263 (7-11 Jun. 2004).

14. Lee, S.-J., Lee, K. and Yoo, H.-J. "Analysis and implementation of practical, cost-effective networks on chips", *IEEE Des. Test*, **22**(5), pp. 422-433 (2005).

15. Goossens, J.D.K. and Radulescu, A. "Aethereal network on chip: Concepts, architectures, and implementations", *IEEE Des. Test*, **22**(5), pp. 414-421 (2005).

16. Dally, W. and Towles, B., *Principles and Practices of Interconnection Networks*, Morgan Kaufman (2004).

17. Nesson, T. and Johnsson, S.L. "Romm routing on mesh and torus networks", in *SPAA '95: Proceedings of the Seventh Annual ACM Symposium on Parallel Algorithms and Architectures*, pp. 275-287 (1995).

18. Seo, D., Ali, A., Lim, W.-T., Rafique, N. and Thottethodi, M. "Near-optimal worst-case throughput routing for two-dimensional mesh networks", in *ISCA '05: Proceedings of the 32nd Annual International Symposium on Computer Architecture*, pp. 432-443 (2005).

19. Bahn, J.H., Lee, S.E. and Bagherzadeh, N. "On design and analysis of a feasible network-on-chip (noc) architecture", in *Proceedings of International Conference on Information Technology (ITNG'07)*, pp. 1033-1038 (2007).

20. Lee, S.E., Bahn, J.H. and Bagherzadeh, N. "Design of a feasible on-chip interconnection network for a chip multiprocessor (cmp)", *19th International Symposium on Computer Architecture and High Performance Computing*, SBAC-PAD 2007, pp. 211-218 (24-27 Oct. 2007).

21. Lin, X., McKinley, P.K. and Ni, L.M. "The message flow model for routing in wormhole-routed networks", *IEEE Transactions on Parallel and Distributed Systems*, **06**(7), pp. 755-760 (1995).

22. ARM, Arm11 mpcore core, available at: http://www.arm.com.

23. IBM, Ibm powerpc 405 embedded core, available at: http://www.ibm.com.

24. Duato, J., Yalamanchili, S. and Lionel, N., *Interconnection Networks: An Engineering Approach*, Morgan Kaufmann Publishers Inc. (2002).

25. Worm, F., Ienne, P., Thiran, P. and Micheli, G.D. "An adaptive low-power transmission scheme for on-chip networks", in *ISSS '02: Proceedings of the 15th International Symposium on System Synthesis*, pp. 92-100 (2002).

26. Li Shang, Li Shiuan Peh, and Jha, N.K. "Dynamic voltage scaling with links for power optimization of interconnection networks", *High Performance Computer Architecture, 2003. HPCA-9 2003*, pp. 91-102 (Feb. 8-12, 2003).

27. Wei, G.Y., Kim, J., Liu, D., Sidiropoulos, S. and Horowitz, M.A. "A variable-frequency parallel I/O interface with adaptive power-supply regulation", *IEEE Journal of Solid-State Circuits*, **35**(11), pp. 1600-1610 (Nov. 2000).

28. Kim, J. and Horowitz, M.A. "Adaptive supply serial links with sub-1V operation and per-pin clock recovery", *IEEE International Solid-State Circuit Conference, Digest of Technical Papers, ISSCC*, **2**, pp 216-480 (2002).

29. Stine, J.M., Carter, N.P. and Flich, J. "Comparing adaptive routing and dynamic voltage scaling for link power reduction", *IEEE Comput. Archit. Lett.*, **3**(1), p 4 (2004).

30. Lu, Y.-H., Benini, L. and De Micheli, G. "Dynamic frequency scaling with buffer insertion for mixed workloads", *IEEE Trans. Computer Aided Design of Integrated Circuits and Systems*, **21**(11), pp. 1284-1305 (2002).

31. Hsu, C.-L., Wang, W.-T. and Hong, Y.-F. "Frequency-scaling approach for managing power consumption in nocs", *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, **E88-A**(12), pp. 3580-3583 (2005).

32. Dally, W. "Express cubes: improving the performance of k-ary n-cube interconnection networks", *Computers, IEEE Transactions on*, **40**(9), pp. 1016-1023 (Sept. 1991).

33. Wang, H., Peh, L.-S. and Malik, S. "Power-driven design of router microarchitectures in on-chip networks", in *MICRO 36: Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture*, Washington, DC, USA: IEEE Computer Society, p 105 (2003).

34. Peh, L.-S. and Dally, W.J. "A delay model and speculative architecture for pipelined routers", in *HPCA '01: Proceedings of the 7th International Symposium on High-Performance Computer Architecture*, Washington, DC, USA: IEEE Computer Society, p 255 (2001).

35. Kumar, A., Peh, L.-S., Kundu, P. and Jha, N.K. "Express virtual channels: towards the ideal interconnection fabric", in *ISCA '07: Proceedings of the 34th Annual International Symposium on Computer Architecture*, New York, NY, USA: ACM, pp. 150-161 (2007).