HANDSHAKE AND CIRCULATION FLOW CONTROL IN NANOPHOTONIC INTERCONNECTS

A Thesis

by

JAGADISH CHANDAR JAYABALAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

August 2012

Major Subject: Computer Engineering

HANDSHAKE AND CIRCULATION FLOW CONTROL IN NANOPHOTONIC INTERCONNECTS

A Thesis

by

JAGADISH CHANDAR JAYABALAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Co-Chairs of Committee,	Eun Jung Kim
	Paul V. Gratz
Committee Member,	Lawrence Rauchwerger
Head of Department,	Duncan M. (Hank) Walker

August 2012

Major Subject: Computer Engineering

ABSTRACT

Handshake and Circulation Flow Control in Nanaphotonic Interconnects. (August 2012)

Jagadish Chandar Jayabalan, B. Tech., National Institute of Technology, Tiruchirappalli

Co-Chairs of Advisory Committee, Dr. Eun Jung Kim
Dr. Paul V. Gratz

Nanophotonics has been proposed to design low latency and high bandwidth Network-On-Chip (NOC) for future Chip Multi-Processors (CMPs). Recent nanophotonic NOC designs adopt the token-based arbitration coupled with credit-based flow control, which leads to low bandwidth utilization. This thesis proposes two handshake schemes for nanophotonic interconnects in CMPs, Global Handshake (GHS) and Distributed Handshake (DHS), which get rid of the traditional credit-based flow control, reduce the average token waiting time, and finally improve the network throughput. Furthermore, we enhance the basic handshake schemes with setaside buffer and circulation techniques to overcome the Head-Of-Line (HOL) blocking. The evaluations show that the proposed handshake schemes improve network throughput by up to 11× under synthetic workloads. With the extracted trace traffic from real applications, the handshake schemes can reduce the communication delay by up to 55%. The basic handshake schemes add only 0.4% hardware overhead for optical components and negligible power consumption. In addition, the performance of the handshake schemes is independent of on-chip buffer space, which makes them feasible in a large scale nanophotonic interconnect design.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Performance of a system depends on the utilization of the computational components of the system. If the computational components of the system can be kept busy all the time then maximum performance can be reached. Since multiple computational components need to work together, providing efficient communication between them is an important aspect in achieving maximum utilization.

1.1 Computation

Performance of uniprocessors is not increasing at the same rate as before. One of the popular methods used to increase uniprocessor performance is employing various superscalar techniques like out of order issue and large instruction windows, to execute multiple instructions in parallel. But the instruction-level parallelism is very limited in most applications. As the feature size keeps on decreasing, more and more transistors are being integrated on the chip. A many-core era with thousands of cores in a single die has been expected to exploit the increasing number of transistors and due to the drive to create more applications that exhibit more thread-level parallelism. To keep the computational components fully utilized an efficient communication system is required on the chip. One of the critical factors in high performance Chip Multi-Processors (CMPs) is architecting efficient communication on a single chip [1].

This thesis follows the style of *IEEE Transactions on Parallel and Distributed Systems*.

1.2 Communication

1.2.1 Interconnects

Network-On-Chip (NOC) is a promising candidate for orchestrating chip-wide communications in the many-core era. Electrical interconnects have been used to provide global communications between multiple components. For long wires the time taken to transmit the data is high. As the features size keeps on decreasing with each generation, the number of clock cycles taken to transmit data through the global wires keeps on increasing. This increase in latency might keep the computational components waiting and hence communication becomes a bottleneck for overall performance.

Silicon nanophotonics is an emerging alternative to electrical interconnects because of its high speed communication over long distance. It provides a new area for design exploration and the challenges and opportunities are different from that of electrical internconnects. Light travels through silicon waveguides provided on chip. In nanophotonics the power consumption occurs only at the ends of the silicon waveguides. So power is independent of how long the data is transmitted. Nanophotonic interconnects are becoming a viable low latency alternative to electrical interconnects.

1.2.2 Flow Control

Since on-chip buffers are limited resources, flow control becomes a critical factor in the NOC design. In electrical NOC with hop-by-hop transmission, credit-based flow control is preferred since the most recent credit information is instantly available due to the short communication delay between neighbors. In addition, to get the best

throughput with credit-based flow control, it is necessary to keep enough number of buffers to help cover the credit round-trip delay. The short transmission delay between neighbors helps reduce the buffer requirement. On the other hand, in ring-based optical interconnects, where each node is attached to the shared ring, the traffic logically becomes one-hop communication. The one-hop delay between source and destination depends on the ring size, and is normally multiple cycles, which makes credit-based flow control become inefficient in the ring-based optical interconnect. This work proposes alternate flow control mechanisms for nanophotonic interconnects to overcome the disadvantages of credit-based flow control.

1.3 Thesis Contributions

1.3.1 Global Handshake and Distributed Handshake

This work proposes two handshake schemes for nanophotonic interconnects, Global Handshake (GHS) and Distributed Handshake (DHS). Instead of using the credit-based flow control, the proposed handshake schemes rely on acknowledgments between senders and receivers. A sender begins to transmit packets right after winning the channel arbitration without knowing the buffer status at the receiver side. A receiver sends ACK or NACK messages as a feedback. Packet dropping and retransmission may occur if there is not enough buffer space at the receiver.

1.3.2 Setaside Buffer and Circulation

Furthermore, this work proposes setaside buffer and circulation techniques to overcome the Head-Of-Line (HOL) blocking in the basic handshake schemes. Packets are moved to the setaside buffer after transmission, yielding the head position to subsequent packets. The circulation technique gets rid of the extra setaside buffer by keeping packets circulating in the network until receivers have enough buffer space.

Our evaluation shows that the proposed handshake schemes improve network throughput by up to 11× under synthetic workloads with the packet dropping and retransmission rates below 1%. With the extracted trace traffic from real applications, the handshake schemes can reduce the communication latency by up to 55%. The basic handshake schemes add only 0.4% hardware overhead for optical components and negligible power consumption. In addition, the performance of the handshake schemes are independent of on-chip buffer space, which makes them feasible in a large scale nanophotonic interconnect design.

1.4 Thesis Structure

The rest of this thesis is organized as follows. Section 2 provides a background on silicon nanophotonic technology and presents a motivating case study to highlight the inefficiency of existing optical flow control schemes. A summary of related work is presented in Section 3. Section 4 presents a detailed description of the optical handshake schemes. Section 5 describes the evaluation methodology and summarizes the simulation results. Finally, Section 6 concludes the thesis work.

2. BACKGROUND

Silicon nanophotonic interconnects have low latency, low losses and high bandwidth. This section provides an overview of the components needed to provide optical communications and the architecture of the optical interconnect. Last part of the section provides a case study where the disadvantages of the traditional flow control mechanism in optical interconnects are highlighted.

2.1 Optical Communication Components

2.1.1 Laser

All light is generated from an off-chip multi-wavelength laser. With dense-wavelength-division-multiplexing (DWDM), up to 128 wavelengths can be generated and carried by the waveguides [11]. Light is constantly generated by the laser at all times and sent to the chip. The light is generated by a Si/III-V evanescent laser.

2.1.2 Waveguides

Waveguides are provided on chip to confine and guide the light signals throughout the chip. Light from the laser source travels unidirectionally through the waveguides. Multiple wavelengths can use the same waveguide with no interference. Losses in light signals can occur when they propagate through the waveguides. There are ongoing efforts to reduce the losses in the future. Two optical materials are used for the waveguides. A *core* made of high refraction index material like crystalline silicon which has a refractive index of 3.5 and a *cladding* made of low refraction index material like

silicon oxide which has a refractive index of 1.45. A typical waveguide has a cross-sectional length of 500 nm. Figure 1 shows a conceptual optical link.

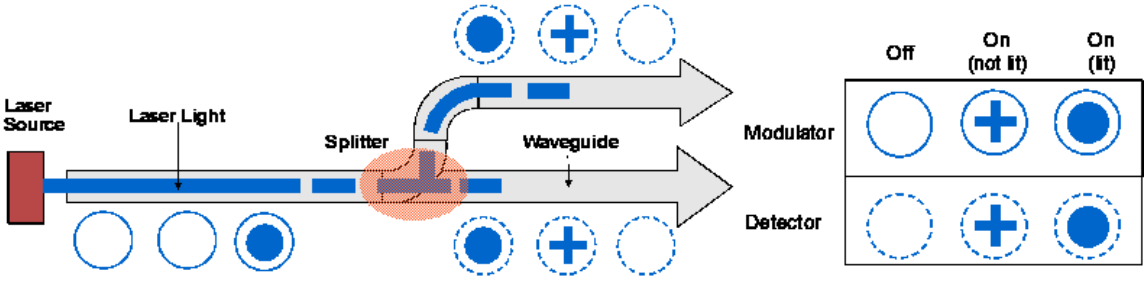

Fig. 1. A Conceptual Optical Link.

2.1.3 Ring Resonators

Ring resonators or micro-rings are used to modulate or demodulate the optical signals. Micro-rings are placed next to a waveguide. They can be tuned between resonance "on" and "off" states by means of an electrical signal.

Diverter or modulator is a micro-ring which is used to convert electrical signals to optical signals. When the micro-ring is on-resonance then it suppresses the light in the waveguide. This suppressed light gets dissipated. When the micro-ring is off-resonance, the light continues through the waveguide. Thus the electrical signals control the resonance of the ring which in turn controls the light in the waveguide. Figure 2 depicts the behavior.

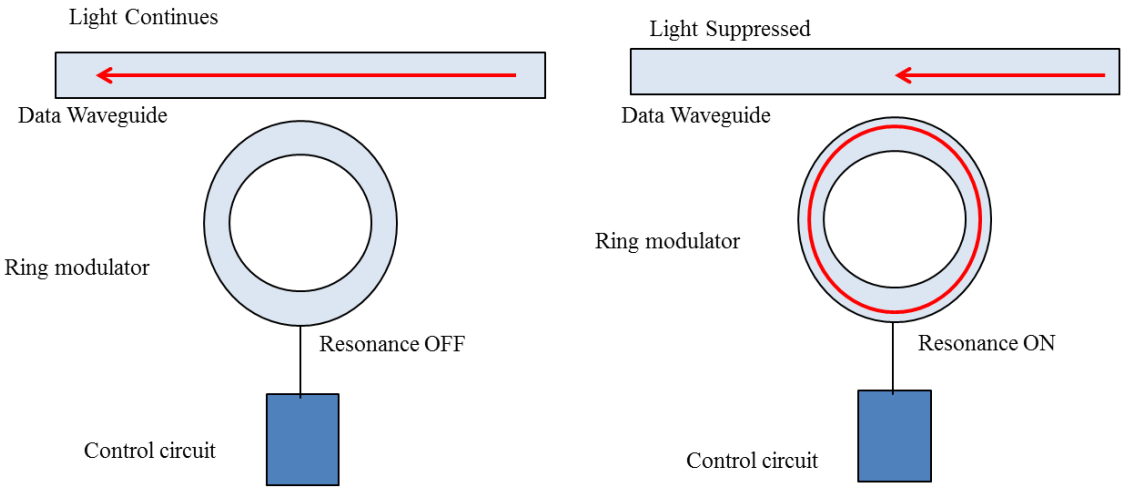

Fig. 2. Electrical to Optical.

Detector micro-rings are used to convert optical signals to electrical signals. During detection the rings are always in resonance on state. If there is light in the waveguide then it is suppressed and dissipated. If there is no light then no dissipation happens in the ring. The main difference in detectors is that it is doped with Germanium which generates a photo electric current when light is dissipated in the ring. Conversely no current is generated when there is no light. Thus optical signals are converted to electrical signals. Ring detection is destructive, which means that an active ring detector removes all the light during the process of detection. Thus, any downstream detectors will not be able to detect the light. Figure 3 depicts demodulation of optical signals.

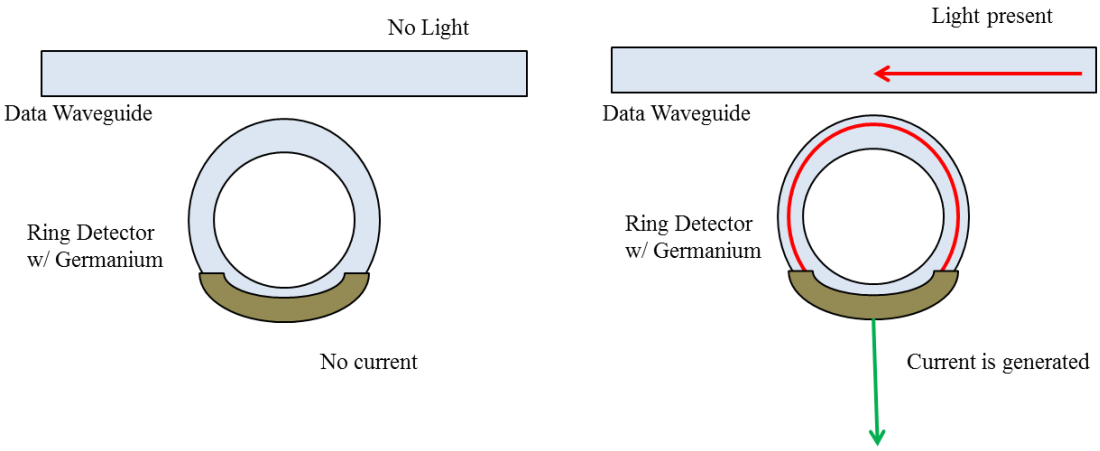

Fig. 3. Optical to Electrical.

Injector is another use of micro-rings where the optical signals can be injected from one waveguide to another. This happens if there are two parallel waveguides with a ring between them and when the ring is tuned to resonance on state. Functioning ring resonators are described in [12].

This thesis adopts single wavelength selective ring resonators and 64 DWDM. A ring can be tuned for only one wavelength at a time. Each ring is sized according to the wavelength it is designated. A ring resonates when its circumference is an integral number of its wavelength.

2.1.4 Splitters

A splitter is used for splitting a fraction of light to another waveguide. Light across all wavelengths are sent to the new waveguide. This is primarily how the light from the off chip laser is made available in multiple waveguides on chip.

The optical components can be on the same die as the processors or on a separate die. Monolithic integration has the advantage of less interfacing overhead but the optical components occupy some die area that could have been used for transistors. In 3D stack integration, the optical components are on a separate layer and the processor cores or a separate layer. While there is some interfacing overhead, the layers are independent of each other and can be optimized separately.

2.2 Optical Tokens

Data waveguides are a shared resource among devices that need to transmit data. Tokens are used to grant exclusive access to this shared resource. An optical token is a pulse of light travelling in a token waveguide. The presence of the optical token means that the data waveguide is free and any sender node can get access to the waveguide by capturing the token. If there is no token pulse then the waveguide is not available. Thus an optical token grants the senders exclusive access to the communication channels. Multiple tokens can traverse the same waveguide at the same time. One token follows another and the total number of tokens depends on how long the waveguide is.

A ring can inject a token into the token waveguide by acting as an *injector*. By a quick resonance on-off transition, a pulse of light is injected into the token waveguide from a power waveguide. A node can detect a token by having a *detector* ring in resonance on condition. Since detection removes the light from the waveguide, by default the detector rings are always in resonance off condition. Only if a sender wants to transmit data, the ring is brought into resonance on condition and detects the token.

2.3 Topology

In traditional electrical interconnects, each node is connected to its neighboring nodes using separate electrical links, such as a 2D Mesh network, while in optical interconnects nodes are normally attached to a single communication media forming a ring-based network as shown in Figure 4. 64 nodes, each of which contains 4 cores, are connected through unidirectional optical rings.

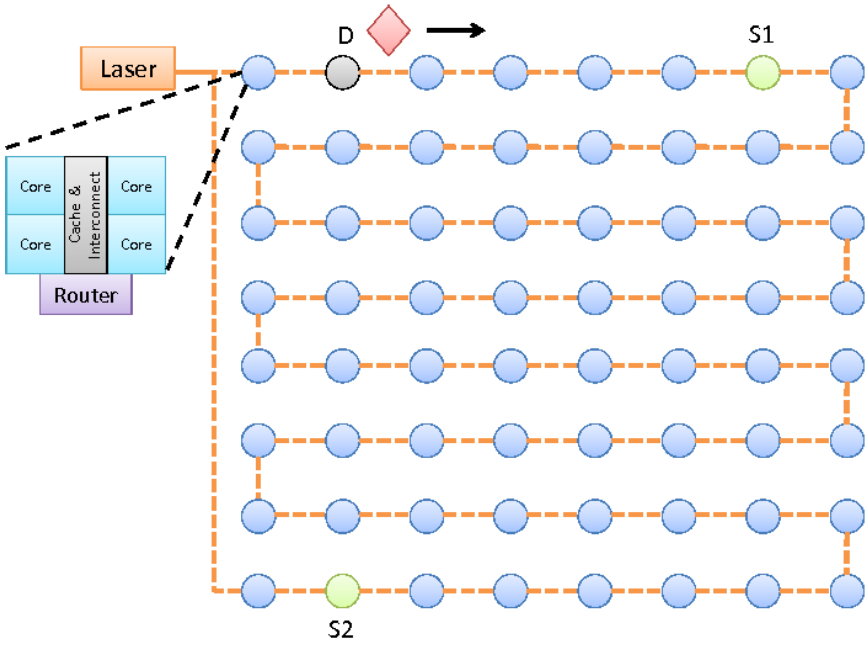

Fig. 4. Ring-Based Network Architecture.

The ring-based optical interconnect falls into two categories: Multiple Write Single Read (MWSR) such as Corona [5], or Single Write Multiple Read (SWMR) such as Firefly [6]. Figure 5 shows these two interconnects. In MWSR, a node can write to all the channels except one specific channel from which the node can read, while in SWMR a node can write to a specific channel from which any other nodes can read. MWSR

needs arbitration in the sender side, since a destination node can only receive one light signal at a time. The advantage of SWMR is that it does not require any arbitration in the sender, but introduces extra communication complexity. Considering multiple nodes can read from one given channel in SWMR, a reader should activate its detector. Since ring detection is destructive, we cannot allow all the nodes to keep their detectors activated all the time. Only the destination node is allowed to open its detector. To handle this situation, before sending data signals, the sender must notify the receiver of the future communication to activate the receiver's detector, which costs extra bandwidth and needs relatively expensive broadcast waveguides. Although the proposed handshake schemes can be applied to both MWSR and SWMR, MWSR is the interconnect pattern of choice for its simplicity and low cost.

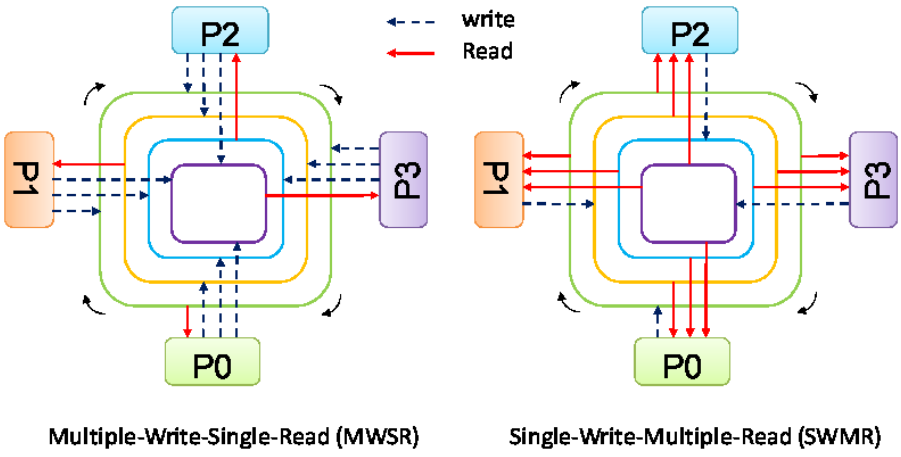

Fig. 5. MWSR and SWMR.

2.4 Arbitration

With limited on-chip channel resource, arbitration is one of the most critical factors in the NOC design. In nanophotonic interconnects, optical packets traverse through optical channels in a pipelined manner, which allows a single optical channel to be divided into several segments, and each segment is similar to a single-cycle bus. For example, on a 576 mm^2 chip with 64 nodes and a 5 GHz clock, the round trip time for an optical channel is 8 cycles [5], so it can be divided into 8 segments. Considering the specific characteristics of optical channels, the arbitration of a shared optical channel can take two methods: global arbitration or distributed arbitration.

2.4.1 Global Arbitration

A token grants permission to use the channel. Global arbitration or token channel arbitration [13] is where there is one global token circulating through the token waveguide. The global token is generated by the reader or *home node* of the waveguide. Whichever sender detects the token gets permission to use the data waveguide. It gets control of the whole channel. It can send multiple packets when it has the full control of the channel. No other node can use the channel at this time. After finishing sending out the packets, the token is reinjected into the token waveguide by the sender. The downstream nodes can acquire the token once the sender releases the token. Then it takes a whole round trip time for the sender to reacquire the token. Distributed arbitration reduces this token waiting time.

2.4.2 Distributed Arbitration

Distributed arbitration or token slot [13] is where the data waveguide is split into multiple slots. A token gives permission to use one slot of the data waveguide. So in distributed arbitration multiple tokens traverse the token waveguide. So the control of using the data waveguide is distributed among multiple tokens. Any node that wants to send a packet acquires the token and sends a packet for the slot the token corresponds to. In distributed arbitration the sender need not reinject the token back into the token waveguide. Any other nodes can acquire token corresponding to other slots. Since tokens travel continuously through the waveguide, the waiting time to reacquire the token is considerably reduced when compared with global arbitration. In global arbitration every node needs the resources to reinject the token back into the token waveguide. Since tokens are not reinjected in the distributed arbitration, these resources are not needed. The *home node* generates a token every clock.

2.5 Fairness

One major problem of token-related protocol is fairness. Considering that a home node acts as a global controller to generate tokens for every sender, nodes close to the home node have higher priority over farther downstream nodes in obtaining tokens. This can starve the farther downstream nodes. A similar problem has been addressed in [13], which proposes Fair Token Channel and Fair Slot with well served nodes sitting on their hands for a while and yielding the chance to other nodes. In this work, we adopt the same methods proposed in [13].

The starved node sends a signal to the home or destination node. Then fairness is made possible by providing a hunger waveguide through which tokens are sent out to the starving nodes. Other nodes that are not facing starvation will not detect tokens in the hunger waveguide. So the starved nodes get permission to use the channel or slot by acquiring tokens in the hunger waveguide. No tokens are sent through the normal token waveguide when interconnect is in hunger mode.

2.6 Motivation

Traditional electrical on-chip interconnects hire credit-based flow control, in which upstream routers keep a count of free buffers that are present in downstream routers. When a router sends a flit to the next hop and frees a buffer, it sends a credit backward to its upstream router. Inherited from credit-based flow control, all the above token-based arbitration schemes integrate the credit information into the arbitration token [13].

Traditional credit-based flow control benefits from the short and fixed transmission delay (normally one cycle) between neighboring nodes. However, in optical interconnects, the transmission latency between neighboring nodes is not always one cycle, which delays the synchronization of the credit information between the sender and the receiver. Figure 4 shows such a situation. We assume the round trip time for the ring is 8 cycles. Nodes S_1, S_2 and D are connected in a ring, as shown in Figure 4. Nodes S_1 and S_2 want to send packets to Node D. Before sending a packet, S_1 and S_2 need to get a token from Node D, which also carries the credit information of Node D, indicated by

T_c in Figure 6. In cycle 0, Node D sends out the token, and its local credit (shown as D_c) becomes zero. In cycle 1, the token arrives at Node S_1, which consumes all the credits. When Node S_1 releases the token, there are no credits left in the token, which means Node S_2 cannot send a packet when the token arrives at Node S_2. Node S_2 should wait until the token returns to Node D and gets reimbursed. As shown in Figure 6, in cycle 4 Node D has newly freed buffer space (D_c becomes 1). However, the token cannot get this information immediately since it is in the middle of transmission. Finally, it takes 17 cycles before Node S_2 has a chance to send a packet.

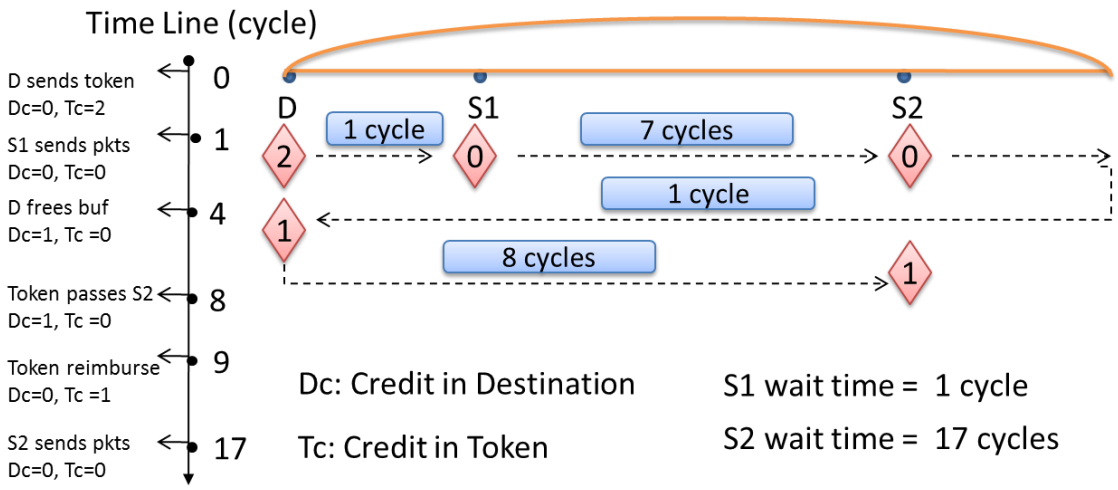

Fig. 6. Coupled Arbitration and Flow Control.

Token slot and token stream try to solve the above problem by adopting multiple tokens. Instead of piggybacking all the credits in a single token, token slot and token stream represents one credit with one token. The number of tokens depends on the number of credits at destination nodes. Destination nodes stop generating tokens if no

more credits are available, making the network performance rely on the size of on-chip buffer space as shown in Figure 7. The detailed simulation configuration is explained in Section 5. We observe that a certain amount of on-chip buffers should be provided to avoid performance degradation. Therefore, credit-based flow control coupled with token-based arbitration is inefficient in the ring-based nanophotonic interconnect design.

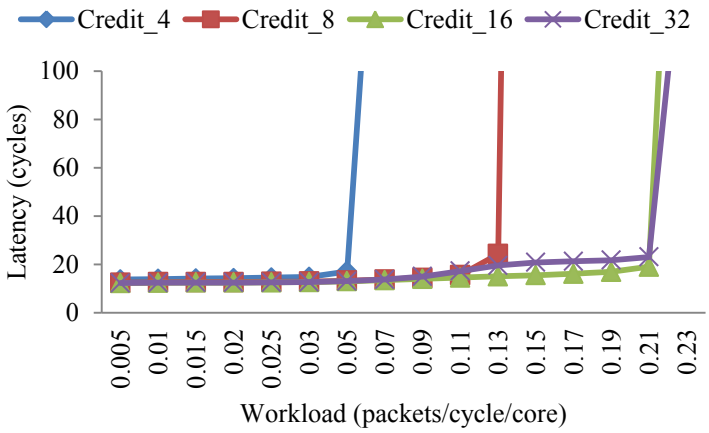

Fig. 7. Performance of Token Slot.

3. RELATED WORK

Architects have explored alternative technologies including electrical transmission lines [3], radio frequency (RF) signaling [4], and nanophotonics [5, 6, 7]. While electrical transmission lines and RF suffer from low bandwidth density and relatively large components, nanophotonics provides high bandwidth density, low latency, and distance-independent power consumption, which makes it a promising candidate for future NOC designs. Traditional electrical interconnects may not be able to meet scalability and high bandwidth while maintaining acceptable performance within power and area budgets [2].

Optical interconnects have been leveraged to build various on-chip networks. Kirman et al. [7] propose to use optical components to build on-chip buses. They explore snoop based coherence protocol. Before the request is sent the receiver needs to make sure that there is sufficient buffering available. This is similar to credit based protocol. A request sent by the receiver resembles the availability of a credit at the receiver.

Shacham et al. [9] propose a circuit-switching photonic interconnect for data packets in parallel with an electric network. In this first an electrical control packet sets up the switches needed for optical transmission of the data. Once the path setup is complete then the data is sent to the destination. An acknowledgment is sent back to ensure guaranteed delivery. Gu et al. [10] also use electrical path setup and optical transmission scheme that are similar to [9]. These methods require an extra electrical network on top of the optical network. This work does not have the overhead of an extra electrical network.

Nanophotonic switching is explored in the Phastlane [8]. Pre-decoded source routing is made use of to transmit packets across multiple hops in a mesh network. If there is a contention for the channel, then packets are demodulated and buffered in electrical networks. If there are insufficient buffers then the packet is dropped and the message is sent to the sender through a separate high speed drop network. There are many overheads in this method like the drop network and need for demodulation of optical packets at intermediate nodes that might happen multiple times for a single transmission.

Firefly [6] uses partitioned nanophotonic crossbars to connect clusters of electrically connected mesh networks. This scheme makes use of SWMR topology. Credits are sent to the upstream nodes by piggybacking on the packet. So the credit information is maintained at the nodes. We saw earlier how credit based flow control is inefficient for optical networks.

Joshi et al. [17] build a nanophotonics clos network, which provides uniform latency and throughput with low power. They use wormhole flow control so the receiver buffer is first reserved before the data is sent to the next hop. Ha et al. [18] and Kodi et al. [19] advocate token-based protocols to arbitrate for optical off-chip interconnects. An optical arbiter can be found in [20].

FlexiShare [14] reduces the number of channels across the network and proposes single-pass and two-pass token stream arbitration. The credit information is embedded in the tokens and the sender can transmit according to the credit information.

Vantrease et al. [13] propose token channel and token slot for optical on-chip interconnects, which piggyback flow control information on the arbitration tokens. Token channel is derived from [21]. This method also embeds the credit information in the tokens. We saw in Section 2.6 how credit based flow control is not suited for optical interconnects. To summarize since the links are long in the optical interconnects the relaying of credit information is very inefficient and so the waiting time for sending out the data becomes large. This work moves away from using credit based flow control and uses handshake messages. In this work the data is sent irrespective of credit information and so the waiting time for the data is reduced.

Handshake has been widely used in the Internet. TCP/IP protocol adopts three-way handshake for reliable data transfer (RDT) [15]. In TCP/IP, a receiver sends an acknowledgment to the sender located thousands of miles away as a feedback after receiving a message. This acknowledgment does not provide flow control between senders and receivers. On-chip interconnect, which is considered to be reliable, also hires acknowledgment-based transmissions. In a circuit switching network, to set up a transmission circuit from source to destination, a routing probe is injected and traversing to the destination, which will send back an acknowledgment to notify the successful circuit set-up. SCARAB [16] introduces an optimized NACK network to provide retransmission in a bufferless network.

4. HANDSHAKE AND CIRCULATION FLOW CONTROL

In this section, we propose two handshake schemes, Global Handshake (GHS) and Distributed Handshake (DHS). Instead of using the credit-based flow control, the proposed handshake schemes rely on acknowledgments between senders and receivers. GHS uses global arbitration, while DHS adopts distributed arbitration. Because of DWDM and the speed of light, nanophotonics is capable of high bandwidth densities. So a large data packet can fit in a single flit. With a multi-flit packet, the header information can be added to each flit. In this work, we assume each packet contains a single flit.

4.1 Global Handshake

With global arbitration, GHS has a single token relayed among different senders. Since there are multiple writers but only a single reader in MWSR, the reader or the destination node is responsible for sending out the arbitration token. We define the single reader or destination node as a home node. When a node detects and removes the token, it gets exclusive access to the data channel and starts to send packets in the next cycle. If there are no more packets to be sent, the token will be released to the other nodes and finally return to the home node. It will take multiple cycles for a packet to arrive at the home node. Since senders have no information about the buffer status of the home node, after the packet is sent, it cannot be removed from the sender side. When the packet arrives, the home node checks its buffer status. If there is free buffer space, the packet is stored into the buffer and an ACK message is sent back as a handshake message to the source node. Otherwise the packet is dropped and a NACK message is

sent. When the source node receives an ACK message, the packet is removed from its input buffer and the following packets are ready for transmission. If a NACK message is received, the packet is waiting for retransmission.

Figures 8 to 11 show the operation of Global Handshake. In this example, Node P_0 is set as the home node, and the other nodes try to send packets to P_0. This work assumes that it takes one cycle for the token to traverse between two neighboring nodes.

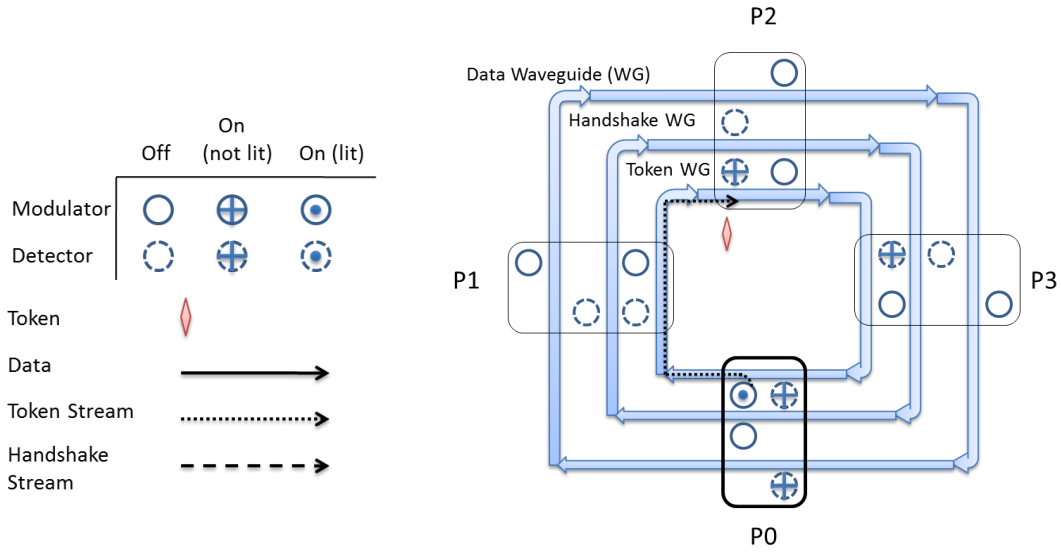

Fig. 8. GHS Cycles 0 and 1.

In Cycle 0, Figure 8, Node P_0 sends out the arbitration token, which will keep circulating in the token channel. Since Node P_1 has no request, the token passes Node P_1 and arrives at Node P_2 in Cycle 1. In Cycle 2, Figure 9, Node P_2 begins to send a data packet. Because Node P_2 has no more packets to send, it releases the token.

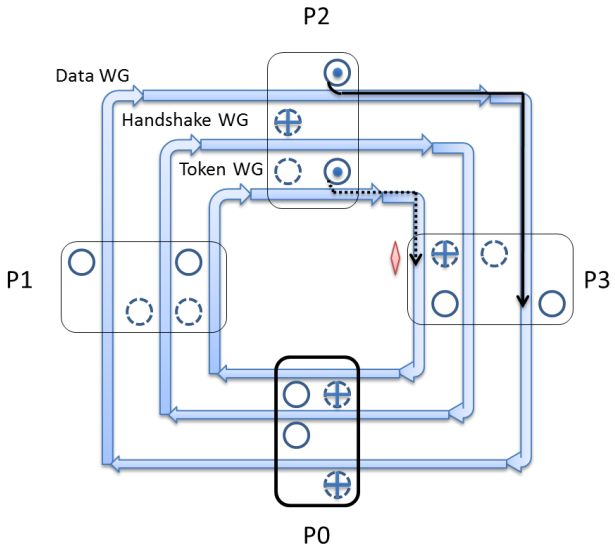

Fig. 9. GHS Cycle 2.

In Cycle 3, Figure 10, Node P_3 gets the token and sends its data packet, which follows the packet from Node P_2 in a wave-pipelined manner. The token stays in Node P_3, since it has more packets to send.

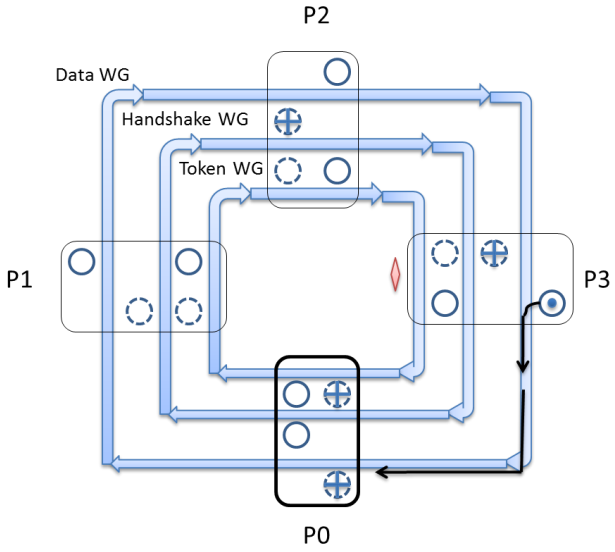

Fig. 10. GHS Cycle 3.

In Cycle 4, Figure 11, the packet from Node P_2 arrives at the home node, which has free buffer slots. An ACK message is sent to Node P_2 through the handshake waveguide.

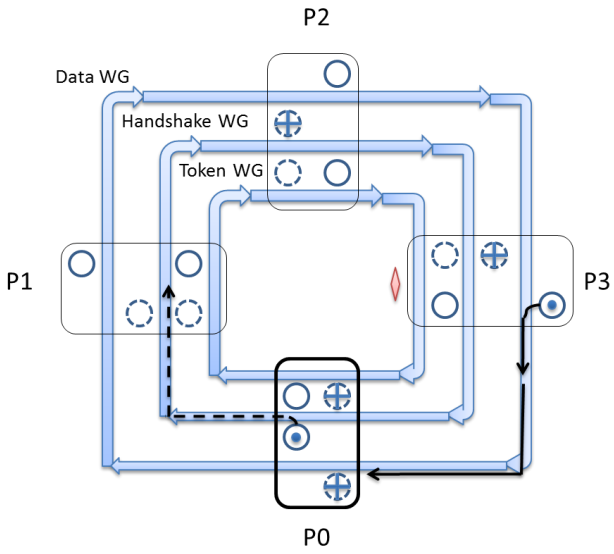

Fig. 11. GHS Cycle 4.

Global Handshake gets rid of the traditional credit-based flow control. Senders can send a packet without knowing the buffer status at the home node even though there could be no credits available at the home node in the current cycle. If the home node frees a buffer slot one cycle before the packet arrival, the packet can be successfully delivered. With limited buffer space, packet dropping and retransmission may occur. Based on our evaluation, packet dropping and retransmission rate is less than 1% even in high workloads. Decoupled with flow control, GHS shortens the average waiting time and therefore improves the network throughput. Figure 12 shows the same example as Figure 6 with GHS, where the waiting time for Node S_2 is reduced from 17 cycles to 8

cycles. Global Handshake has only one token circulating around the channel. After releasing the token, it takes a whole round trip time for a node to get the token again, even though other nodes have no packets to send. This situation becomes worse in a large network, in which the token round trip time can be tens of cycles. To solve this problem, multiple tokens should be provided, which introduces Distributed Handshake (DHS).

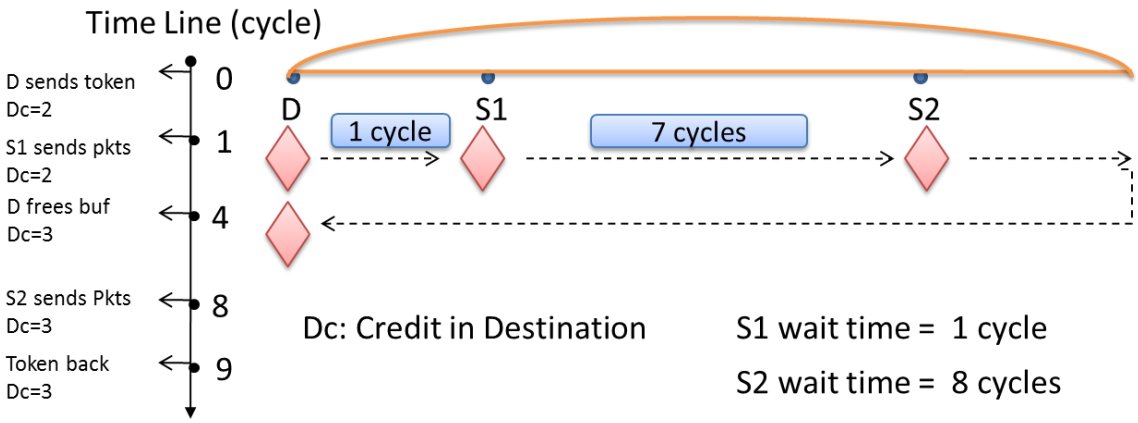

Fig. 12. Global Handshake in a Token-Ring Network.

4.2 Distributed Handshake

DHS considers the wave-pipelined manner of packet transmission in optical links. Home nodes keep generating a token every cycle. Multiple tokens divide the channel into multiple slots that have a fixed size and are back-to-back. In a cycle, only a portion of the network nodes are able to detect the token. If the token is taken by a node, there is no releasing operation for the token and other nodes cannot detect it forever. A

sender can only send one flit after getting a token. Like GHS, packets cannot be removed from the sender side until an ACK message is received.

Figures 13 to 17 show the operation of DHS. Home Node P_0 keeps generating a token every cycle. In Cycle 0, Figure 13, a token arrives at Node P_1, which removes the token.

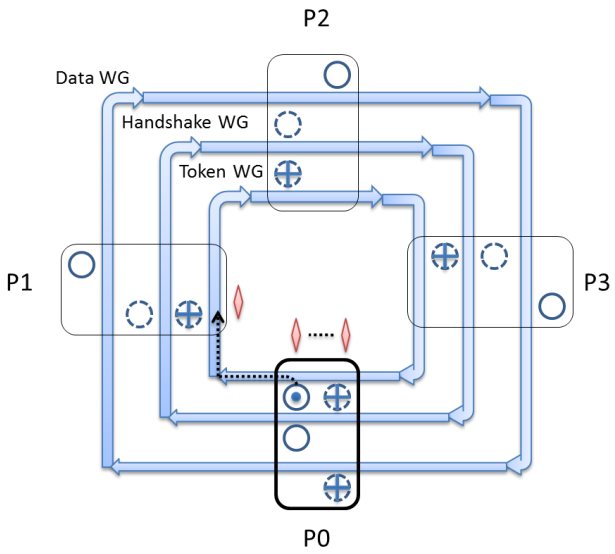

Fig. 13. DHS Cycle 0.

In Cycle 1, Figure 14, Node P_1 starts to send a packet, and turns on the detector in Handshake Channel. Meanwhile, a new token from the home node is generated and arrives at Node P_1 again. However, since there is no new request from Node P_1, the token will keep traversing to Node P_2.

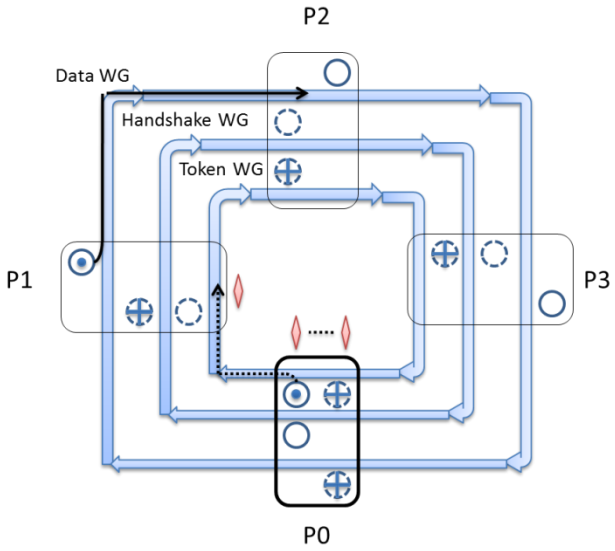

Fig. 14. DHS Cycle 1.

In Cycle 2, Figure 15, the data packet from Node P_1 passes Node P_2, and the token arrives at Node P_2. Node P_2 takes the token, and starts its transmission in the next cycle.

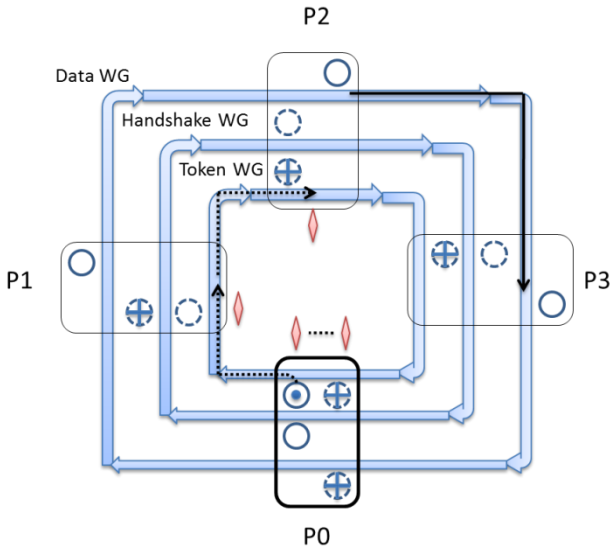

Fig. 15. DHS Cycle 2.

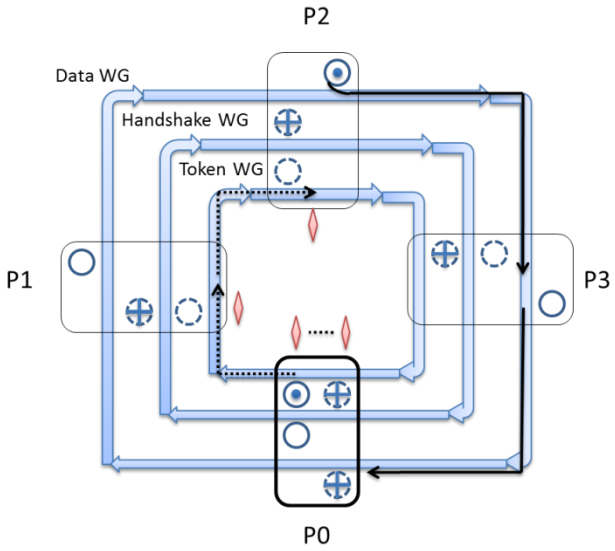

Fig. 16. DHS Cycle 3.

In Cycle 3, Figure 16, Node P_2 sends a data packet which follows the previous data packet from Node P_1, which arrives at the home node. After checking the buffer status, the home node, P_0, sends a handshake message to Node P_1 in Cycle 4, Figure 17.

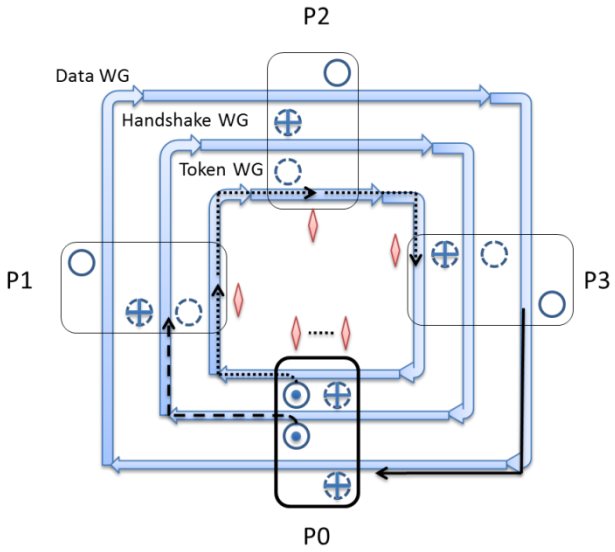

Fig. 17. DHS Cycle 4.

GHS and DHS allow senders to send packets without knowing the buffer status of destination nodes, decouple the channel arbitration with flow control, and consequently reduce the credit synchronizing time ideally to zero. However, basic GHS and DHS suffer from the Head-Of-Line (HOL) blocking problem. Before receiving an ACK message, senders cannot drop the packet that was sent, which makes the packet stay in the head of the input queue for at least a round-trip time. This time is equal to the traversal time of the data packet from source to destination plus the traversal time of the handshake message back from destination to source. The pending packet will block the following packets in the same input buffer. To overcome this, we adopt a setaside buffer technique in GHS and DHS. Setaside buffers are small number of buffer slots that are collocated with input queues. Pending packets are removed from the input buffer and wait for the handshake message in the setaside buffer. Therefore, the next packet becomes the head of the queue and is ready for transmission. The size of setaside buffers may affect the network performance, which is discussed in Section 5.

4.3 Distributed Handshake with Circulation

While the setaside buffer technique tackles the HOL problem with additional buffer space, we propose another technique called circulation to remove this extra buffer overhead. The basic idea is that instead of packet dropping, receivers reinject packets into the same data channels if they run out of buffer space. The reinjected packet will circulate in the optical ring until the buffer is available at the receiver. Without packet dropping in the receiver side, circulation enables senders to remove packets from the

head of input buffers immediately after sending them out, which gets rid of the HOL blocking problem. Considering no packet retransmission, there is no need for receivers to send acknowledgments, and thus handshake waveguides can be removed.

To integrate the circulation technique, basic DHS should be modified. To avoid channel collision, when a home node needs to reinject packets into the data channel, tokens are not generated in the same cycle. It looks like that the home node virtually consumes a token and gets the permission to use the channel. Figures 18 to 22, describes the operation of DHS with the circulation technique.

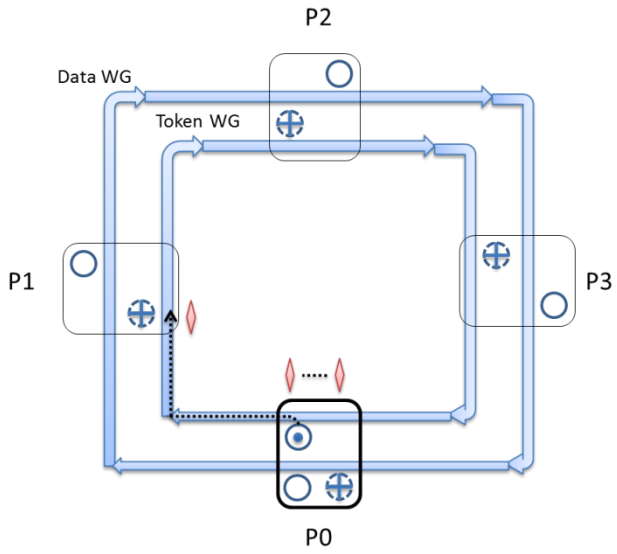

Fig. 18. DHS with Circulation Cycle 0.

Home Node P_0 keeps generating a token every cycle. In Cycle 0, Figure 18, a token arrives at Node P_1, which removes the token. In Cycle 1, Figure 19, Node P_1 starts to send a packet, and removes the packet from its input buffer. Meanwhile, a new token from the home node is generated and arrives at Node P_1 again. However, since there is

no new request from Node P_1, the token will keep traversing to Node P_2. In Cycle 2, Figure 20, the data packet from Node P_1 passes Node P_2, and the token arrives at Node P_2. Node P_2 takes the token, and starts its transmission in the next cycle.

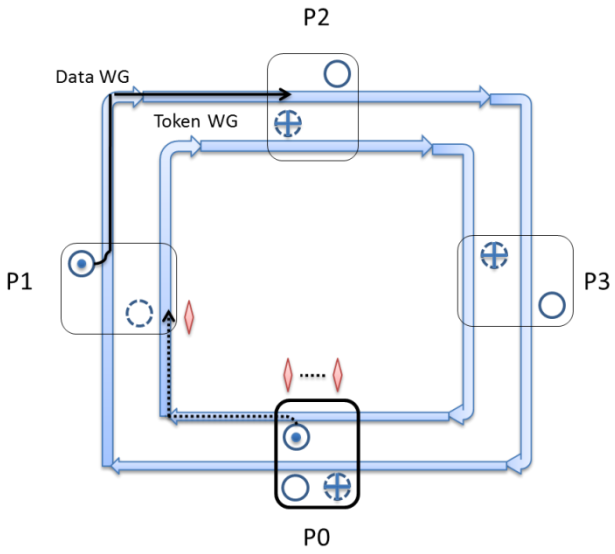

Fig. 19. DHS with Circulation Cycle 1.

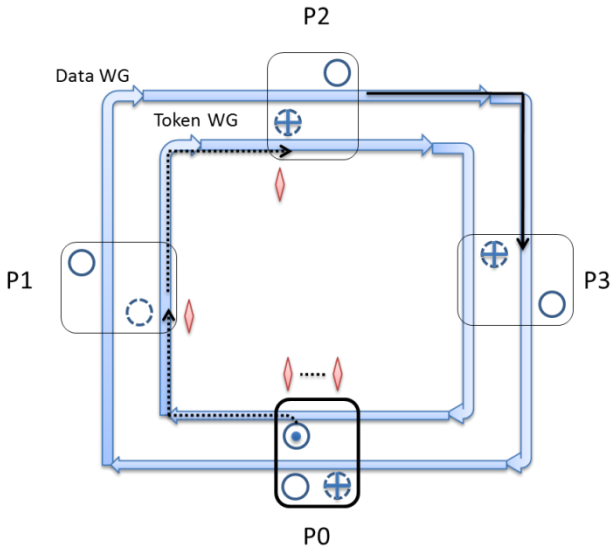

Fig. 20. DHS with Circulation Cycle 2.

In Cycle 3, Figure 21, Node P_2 sends a data packet which follows the previous data packet from Node P_1, which arrives at the home node. Let us assume for this example, there are no free buffer slots at this time at the home node.

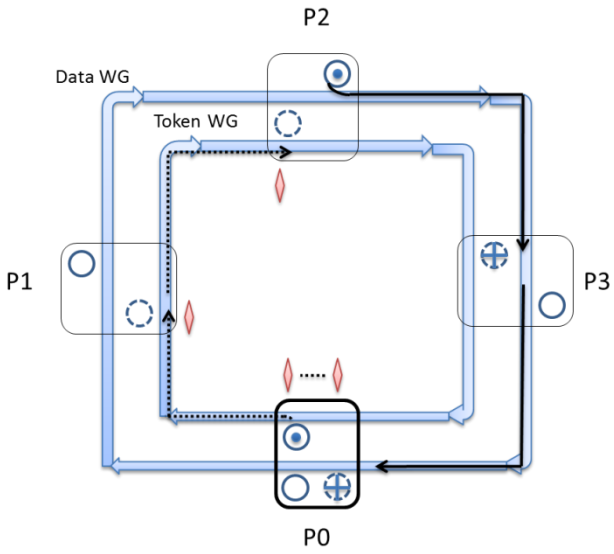

Fig. 21. DHS with Circulation Cycle 3.

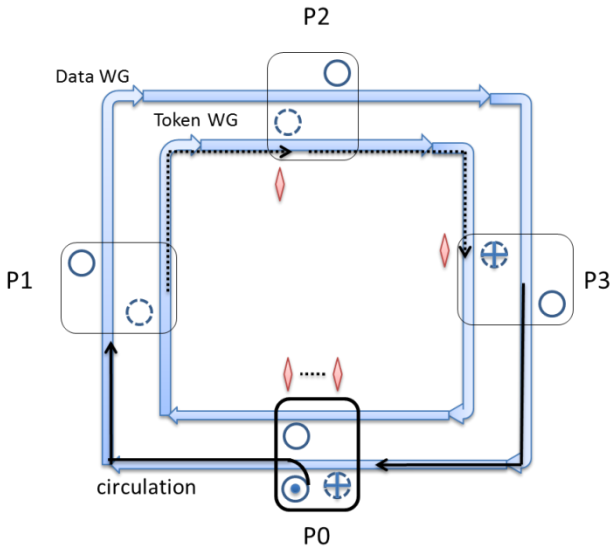

Fig. 22. DHS with Circulation Cycle 4.

In Cycle 4, Figure 22, the home node reinjects the packet into the data channel. Meanwhile, the home node does not generate a token for that cycle. Unlike DHS, the circulation technique cannot be applied to GHS. Note that GHS generates only one channel arbitration token that is relayed among senders. Before the token returns to the home node, the home node cannot grant itself the permission of using the channel and thus no packets are allowed to be reinjected from the home node.

4.4 Network Architecture

4.4.1 Optical Network Architecture

Figure 23 shows the architecture of an optical network with the handshake schemes. Each router is attached to global optical rings, which are composed of different channels, including data channels, token channels and handshake channels. A channel can consist of multiple waveguides, each of which carries 64 wavelengths. To support handshake schemes, extra components are added to the conventional virtual channel (VC) router, which are labeled as Output and Input modules. In the Output module, an output queue, designed as VOQ, is used to buffer the packets before Electronic/Optical (E/O) conversion. To avoid the HOL blocking problem, setaside buffers are added in parallel with the output buffer. Each setaside buffer slot is only one flit long, and connected to an output mux. A handshake receiver processes ACK or NACK messages, and selects a flit to enter E/O conversion. In the *Input* module, the detector checks the status of the global optical ring. If any flit arrives, after Optical/Electronic (O/E) conversion, the flit will be stored into the router input buffer. In basic GHS and DHS

schemes, if there are no empty slots in the input buffer, flits will be dropped. However, with the circulation technique, router buffer status is recorded in the circulation controller, which controls packet reinjection.

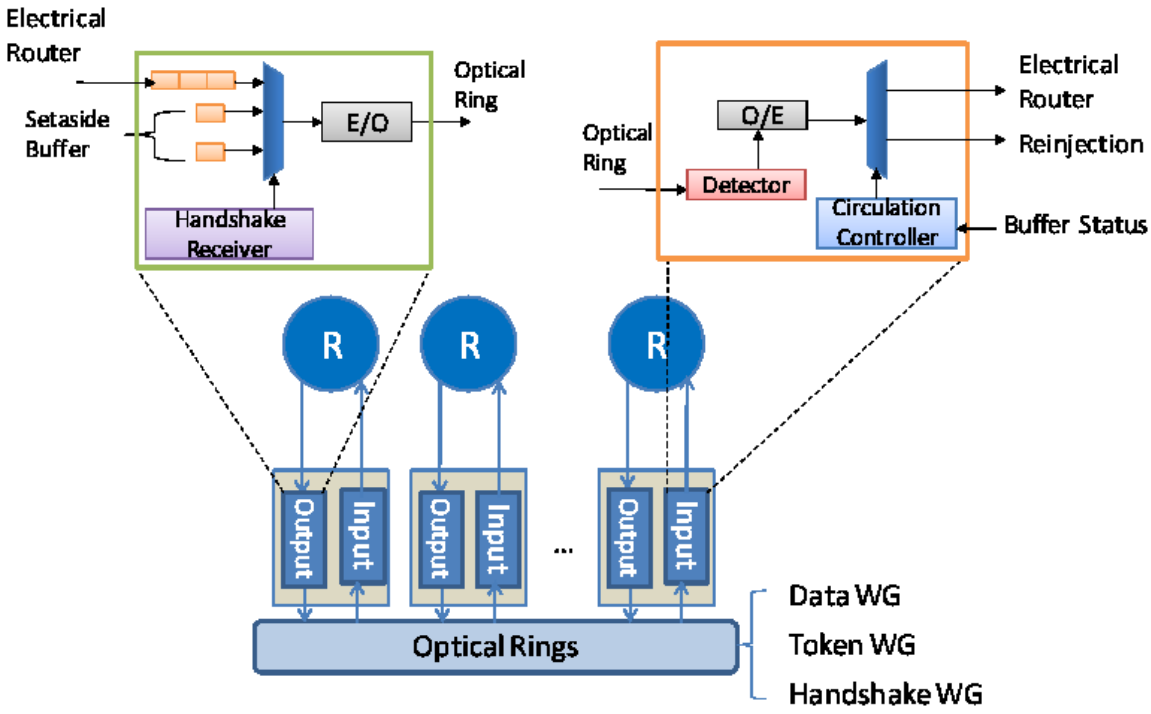

Fig. 23. The Optical Network Architecture with the Handshake Schemes.

4.4.2 Router Pipeline

A conventional electrical router processes packets in four pipeline stages, which are routing computation (RC), VC allocation (VA), switch allocation (SA), and switch traversal (ST). In optical on-chip networks, every router is attached to the global ring making any two routers become neighboring routers, which increases the overhead of recording the VC status for every neighboring router. Note that optical links can provide a wide link width, which is advisable for a single-flit packet design. There is no concern

about flit interleaving in a network with only single-flit packets. Therefore, the VA stage can be removed from the traditional router pipeline, simplifying the electrical router logic. In this work, we adopt a two-stage electrical router, with RC and SA in one stage and ST in the other.

4.4.3 Hardware Overhead

The handshake schemes add handshake messages (ACK and NACK) into normal optical communication, which incurs extra hardware overhead. We analyze the hardware overhead in a network with 256 cores connected as 64 nodes. We advocate using a single bit for a handshake message. Note that in a segment of the channel only one node can get the arbitration token every cycle, and the round trip time for an optical ring is fixed. After sending a packet, the source node will receive a handshake message in a fixed amount of time. For example, if we assume the round trip time for the optical ring is 8 cycles, then a source node will receive the handshake message in 9 cycles. A source node only needs to turn on its handshake detector 9 cycles after sending a packet, while at other times it keeps the detector off and passes the handshake messages for other source nodes. That is why using a single bit, which indicate whether it is an ACK or a NACK, for handshake message is feasible. If we use one wavelength, modulated as 1 bit, for the handshake message of a node, 64 wavelengths are required in a 64-node network. Note that an optical waveguide can carry 64 wavelengths. Thus, only one waveguide is added to support the handshake schemes in a 64-node network. Since each wavelength requires 64 micro-rings to function as modulators or detectors, this extra

waveguide needs total 4K (0.4%) micro-rings. In basic GHS and DHS, a home node only reads packets from its dedicated data channels, while with the circulation technique the home node needs to reinject packets, adding extra 16K (1.5%) micro-rings in the whole network. Table 1 lists the budget of optical components for each handshake scheme.

TABLE 1

Component Budgets for the Handshake Schemes in a 64-node Network

Optical Schemes	**Data WG**	**Handshake WG**	**Token WG**	**Micro-rings**
Token Slot [13]	256	0	1	1024K
GHS	256	1	1	1028K
DHS	256	1	1	1028K
DHS with Circulation	256	0	1	1040K

5. EXPERIMENTAL EVALUATION

In this section, we first describe our evaluation methodology. Then, the performance of the proposed handshake schemes is analyzed, followed by comparison with previous designs. Based on the power model in [22, 17], we estimate the power consumption in the handshake schemes. Finally, we explore the schemes' sensitivity to a variety of network design points.

5.1 Methodology

Our evaluation methodology consists of two parts. First, we use Simics [23], a full system simulator. It is configured to be a SunFire multiprocessor system made up of UltraSPARCIII+ processors. The operating system is Solaris 9. This system is used to extract trace information from real applications. We use a customized timing-model interface modeling out-of-order cores with 4 MSHRs per each processing core to implement a self-throttling CMP network [24]. The CMP system contains 128 out-of-order processing cores and 128 L2 cache banks in a single chip, connected as 64 nodes with 4-way concentration, modeling static non-uniform cache architecture (S-NUCA) [25].

Next, we evaluate performance and power consumption using a cycle-accurate on-chip network simulator that models a 2 stage pipelined router architecture. The total latency of E/O or O/E conversion is around 75 ps [26] and is included in the latency of the nanophotonic link traversal time. When assuming a die size of 400 mm^2 with a 5 GHz clock, the nanophotonic link traversal time varies between 1 to 8 cycles which is

based on how far the sender is from the receiver. The workloads for our evaluation consist of synthetic workloads and traces from real applications.

Three different synthetic traffic patterns, Uniform Random (UR), Bit Complement (BC) and Tornado (TOR), are used. The real applications considered in this work are fma3d, equake, and mgrid from SPEComp2001 [27]; blackscholes, freqmine, streamcluster, and swaptions from PARSEC [28]; FFT, LU, and radix from SPLASH-2 [29]; NAS parallel benchmarks [30] and SPECjbb2000 [31]. Table 2 shows the simulation configuration.

TABLE 2

Simulation Configuration

# Cores	128 out-of-order	Concentration	4
L1I Cache	1-way 32 KB	Router Pipeline Stage	2
L1D Cache	4-way 32 KB	Optical Link Latency	1 - 8 cycles
# L2 Banks	128 512 KB/Bank	Data Channel Width/Flit Size	256 bits
Cache Block Size	64 B	Clock Frequency	5 GHz

5.2 Performance

Given that GHS and token channel use global arbitration while DHS and token slot adopt distributed arbitration, we separate the performance evaluation into two groups. GHS related schemes are compared with token channel, and DHS related schemes are compared with token slot.

5.2.1 Synthetic Workloads

We first evaluate the average packet latency and the throughput saturation bandwidth with synthetic workloads. The total amount of credits or buffer slots provided by each destination is four. The trend from the three traffic patterns is consistent. The handshake schemes achieve approximately 4-6× throughput improvement in UR and 2-11× in BC and TOR. Because token channel [13] suffers from the long token waiting time, especially after senders consume all the credits stored in the token, GHS produces better performance than token channel.

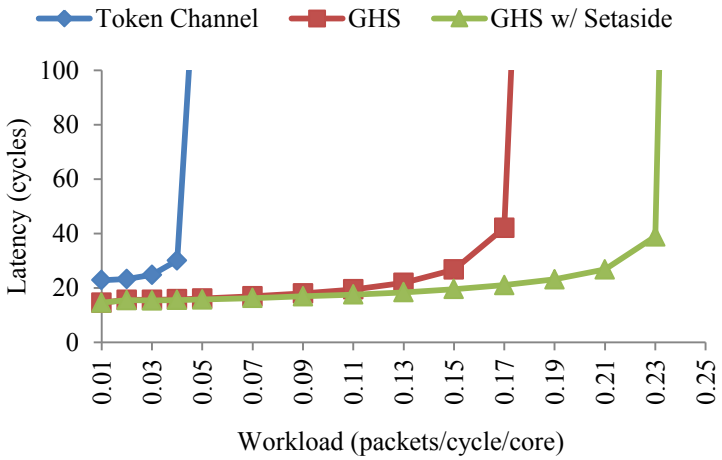

Fig. 24. Performance of GHS in UR.

Figures 24, 25 and 26 show the results of the schemes using global arbitration, in which only one arbitration token is circulating for each destination. In Figure 24, we compare the GHS schemes against token channel for Uniform Random traffic. GHS with Setaside Buffer improves the network throughput by 6× when compared with token channel.

In Figure 25, we compare the GHS schemes against token channel for Bit Complement traffic. GHS with Setaside Buffer improves the network throughput by 11× when compared with token channel.

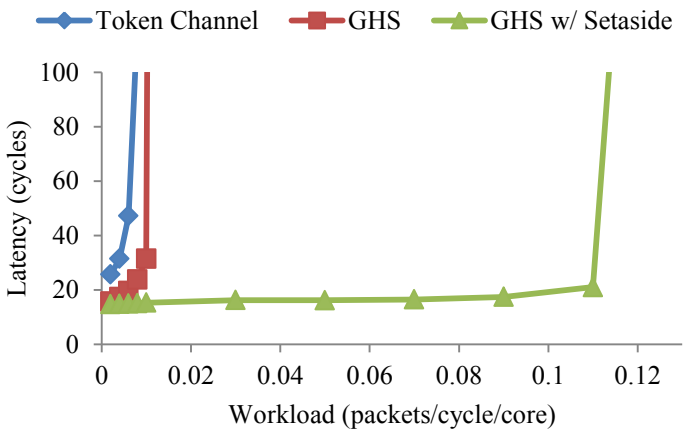

Fig. 25. Performance of GHS in BC.

In Figure 26, we compare the GHS schemes against token channel for Tornado traffic. GHS with Setaside Buffer improves the network throughput by 10× when compared with token channel.

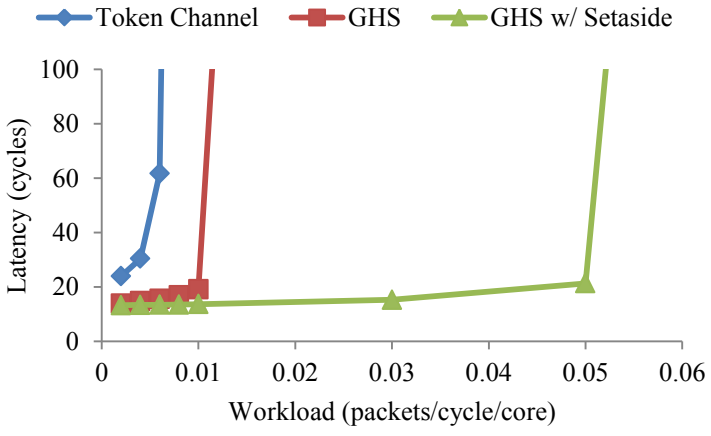

Fig. 26. Performance of GHS in TOR.

In Figures 27, 28 and 29, we evaluate the average token waiting time of different schemes. Since the three traffic patterns have different saturation points, we select different evaluation injection rates for the three traffic patterns in our experiments. Compared with token channel, GHS reduces the average token waiting time dramatically.

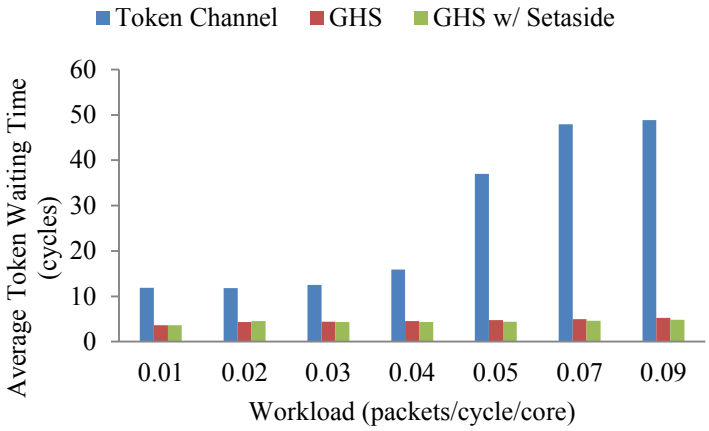

Fig. 27. Token Waiting Time of GHS in UR.

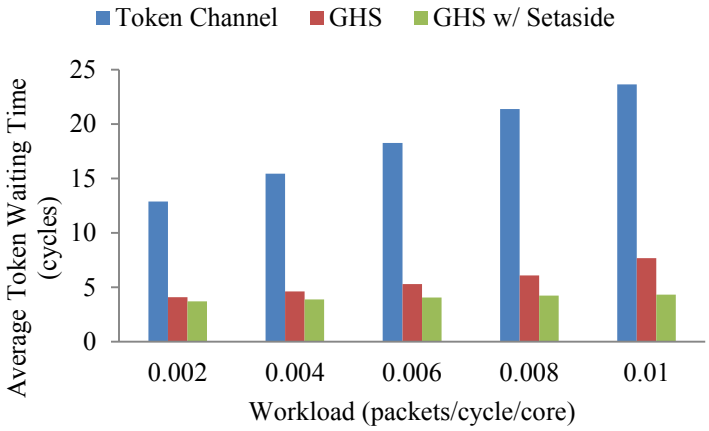

Fig. 28. Token Waiting Time of GHS in BC.

In Figure 27, the token waiting time of GHS schemes are compared against token channel for Uniform Random traffic. We see a clear decrease in average token waiting time. In Figure 28, the same comparison is done with Bit Complement traffic. In Figure 29, the traffic is changed to Tornado traffic. In all traffic patterns, we observe a reduction in average token waiting time.

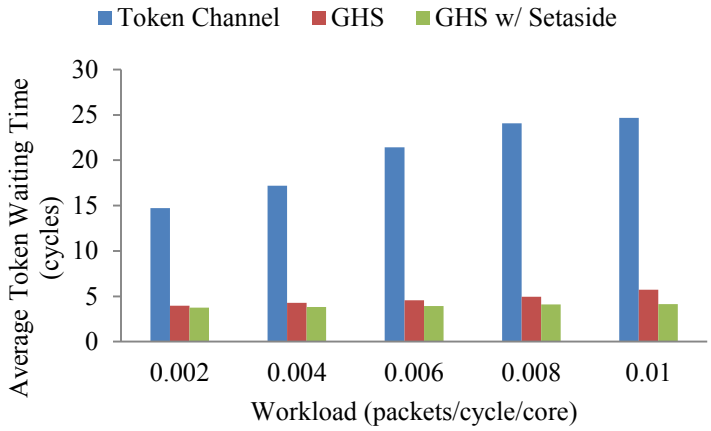

Fig. 29. Token Waiting Time of GHS in TOR.

Compared with token channel, token slot [13] produces better performance. However, since the number of tokens depends on the number of credits at the destination, the destination node with full buffers will stop generating new tokens until a free buffer slot is available. Limited buffer space restrains the performance of token slot. Different from token slot, there is no credit-based flow control in our handshake schemes. Tokens are generated every cycle maximizing the transmission opportunity for senders, which is more efficient than token slot especially when destinations get free buffer space while packets are already in the middle of traversal.

Figures 30, 31 and 32, show the results of the schemes using distributed arbitration. The HOL blocking problem affects the performance of basic GHS and DHS. Although there are free tokens, the following flits cannot seize a token because the flit in the head of the queue is waiting for the acknowledgment. This situation becomes more obvious in the peer-to-peer communication patterns such as BC. From Figure 31, we can see that token slot outperforms basic DHS. With the setaside buffer technique, flits can wait for the acknowledgments in the setaside buffer, yielding the chances to following flits, which brings significant throughput improvement. The setaside buffer and circulation techniques have almost the same effect on relieving the HOL blocking. However, compared with the setaside buffer technique, the circulation does not require additional buffer space, and is a more promising design.

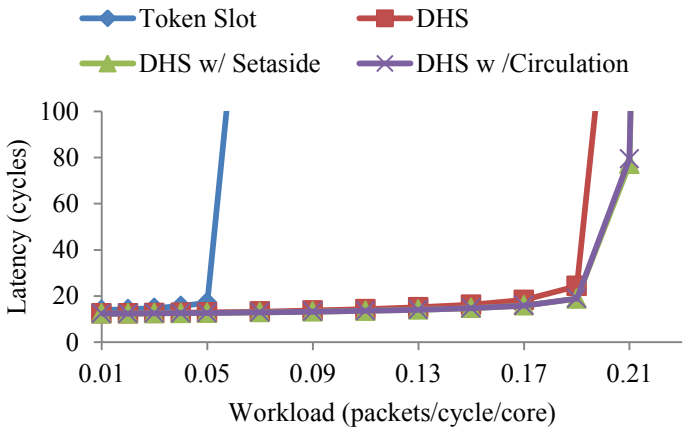

Fig. 30. Performance of DHS in UR.

In Figure 30, we compare the DHS schemes against token slot for Uniform Random traffic. DHS with Circulation improves the network throughput by 4× when compared with token slot.

In Figure 31, the DHS schemes are compared against token slot for Bit Complement traffic. DHS with Circulation improves the network throughput by 2× when compared with token slot.

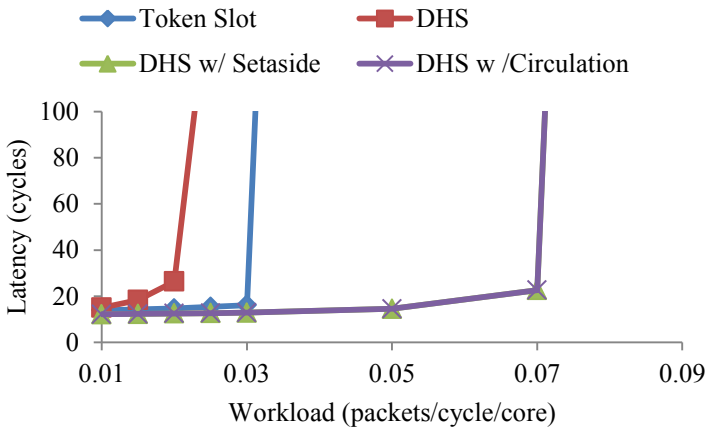

Fig. 31. Performance of DHS in BC.

In Figure 32, the Tornado traffic is used to compare DHS schemes against token slot. DHS with Circulation improves the network throughput by 3× when compared with token slot.

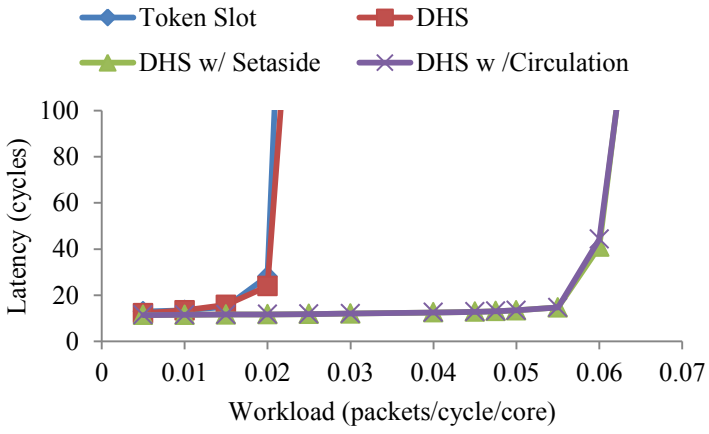

Fig. 32. Performance of DHS in TOR.

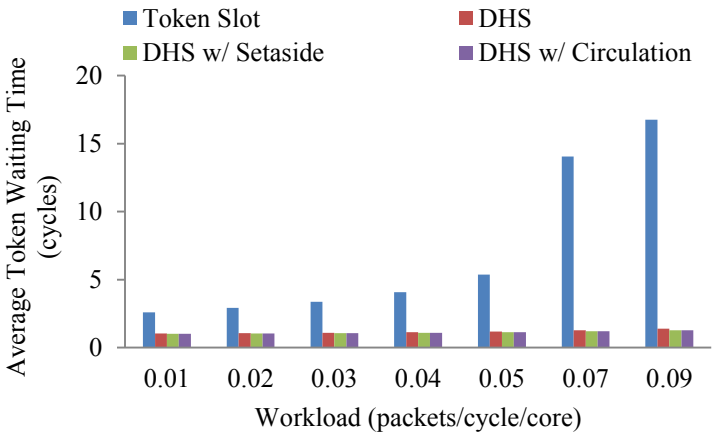

Fig. 33. Token Waiting Time of DHS in UR.

Figures 33, 34 and 35, show that in all the three traffic patterns the average token waiting time is reduced. With multiple arbitration tokens, distributed arbitration shortens the token waiting time. Figure 33 evaluates performance token waiting time in Uniform Random traffic. In Figure 34, Bit Complement traffic is used. Tornado traffic is used I Figure 35. We see reduced average token waiting times for our schemes for all traffic patterns.

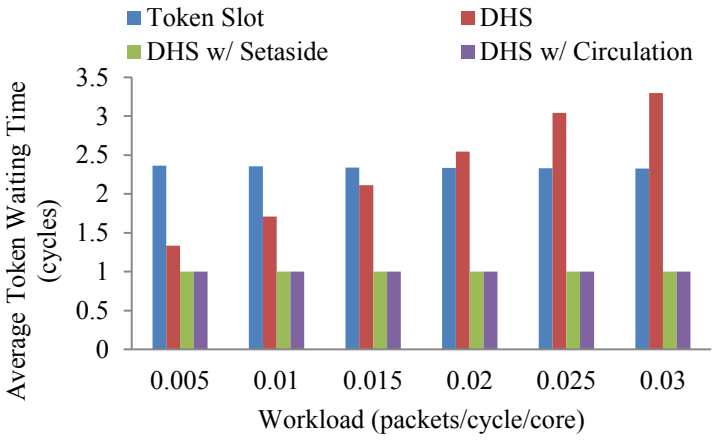

Fig. 34. Token Waiting Time of DHS in BC.

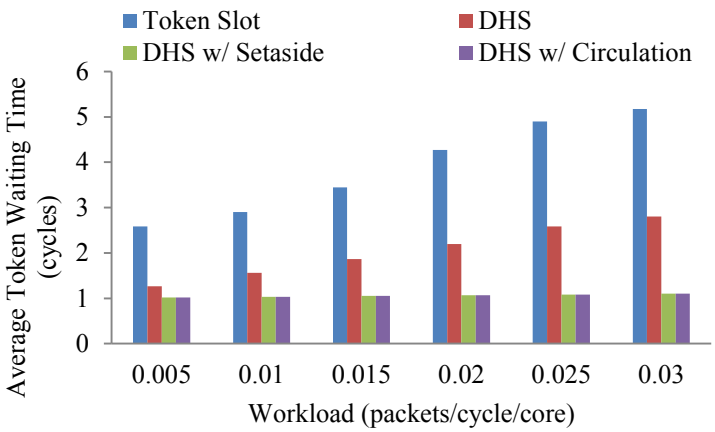

Fig. 35. Token Waiting Time of DHS in TOR.

5.2.1 Real Applications

Figures 36 and 37 show the performance results with real applications. It is clear that the handshake schemes produce obvious performance improvement, especially in NAS parallel benchmarks. Compared with token channel, GHS reduces communication latency by an average of 55%, while DHS achieves an average of 17% latency reduction over token slot. Suffering from the HOL blocking problem, basic GHS and DHS cannot perform as well as GHS and DHS with the setaside buffer and circulation techniques. However, in most of the selected benchmarks, basic GHS and DHS outperform the previous schemes.

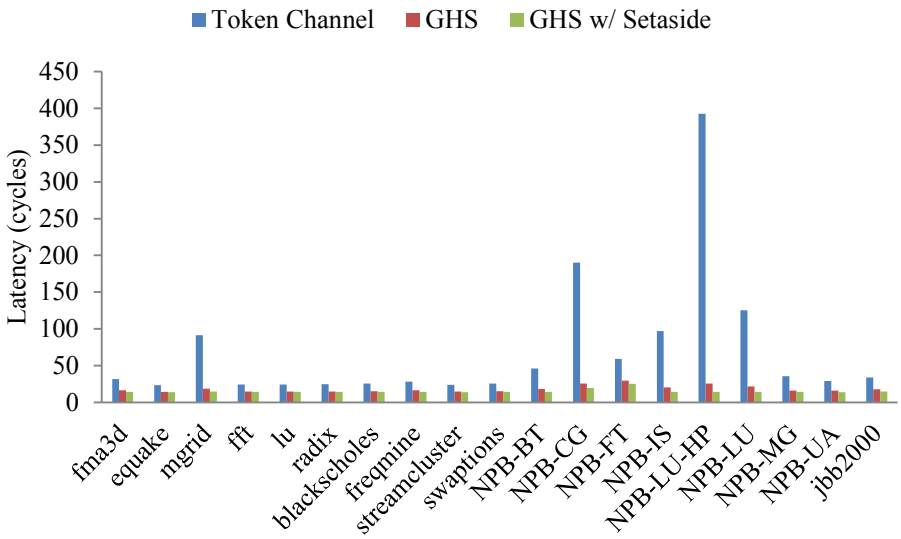

Fig. 36. Performance of GHS in Real Applications.

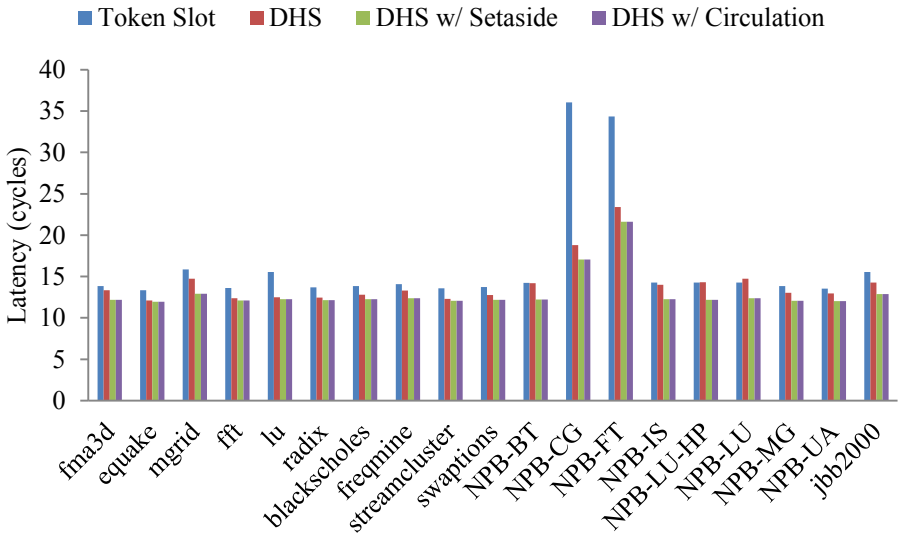

Fig. 37. Performance of DHS in Real Applications.

To study the effect of the handshake schemes on the system performance, we evaluate CPI as depicted in Figures 38 and 39. In this experiment, we select the handshake schemes with the setaside buffer technique to compare with token channel

and token slot separately. GHS improves the CPI by an average of 13% compared with token channel, while DHS gets 1.3% CPI improvement over token slot.

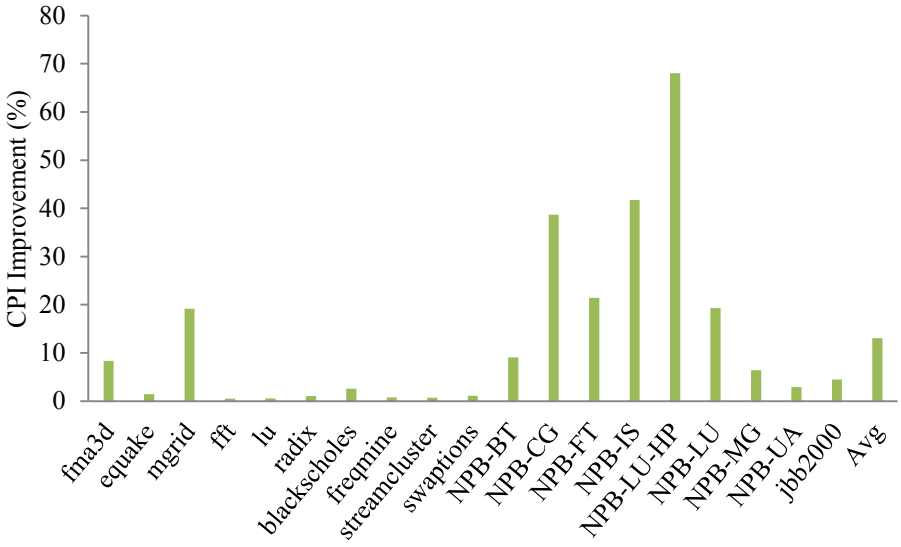

Fig. 38. CPI Improvement using GHS.

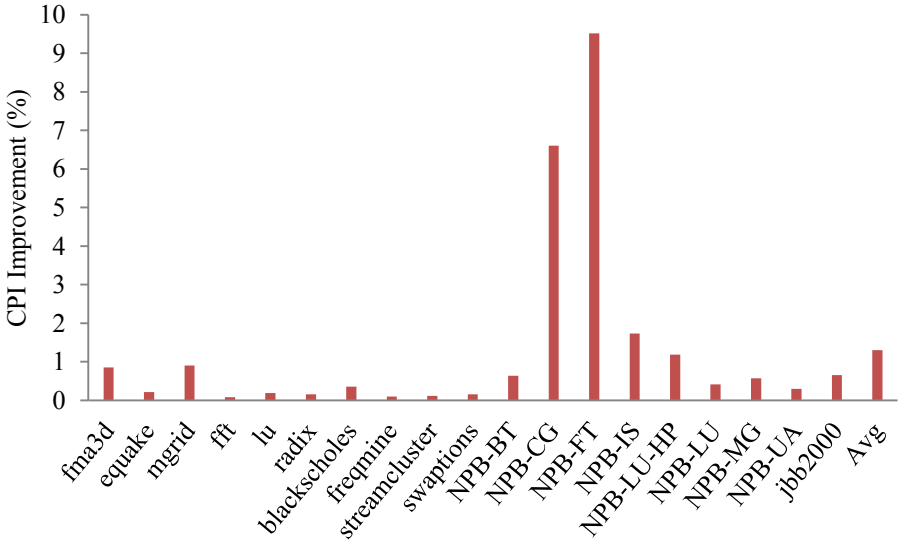

Fig. 39. CPI Improvement using DHS.

5.3 Power

Different from conventional electrical network designs, in which buffers and switches tend to dominate the total power consumption [32], the power dissipated in nanophotonic on-chip networks is composed of electrical router power, modulation/demodulation power, laser power and ring tuning power. Laser power and ring tuning power are also known as static power which dominates the overall power consumption. Modulation/demodulation power is determined by the number of E/O and O/E conversions. Table 3 shows the energy costs of electrical back-end for optical links (modulator drives, receivers, and clocking), and we use 158 fJ/b as the energy cost for each signal conversion. To calculate the laser power, we consider the E/O conversion losses as well as transmission losses in the waveguide.

TABLE 3

Estimated Energy of Electrical Back-End for Optical Links [22]

Component	**Energy (fJ/b)**
Serializer	1.5
Pre-Driver	19
Push-Pull Modulator	70
Analog Receiver Front End	40
Flip-Flop Sampling & Monitoring	12
Deserialiser	1.5
Optical Clocking Source	2
Clock Phase Control	12
Total	**158**

Table 4 lists various optical losses in the optical laser power and the electrical laser power (30% conversion efficiency [17]). Along the optical critical path coupler loss, modulation insertion loss, and filter drop loss are not dependent on the layout, size and topology of the network. Waveguide loss is length-dependent. 30 mW at 1 dB loss is assumed in our evaluations for waveguides. In Corona [5], waveguide and off-resonance ring losses form the majority of power consumption, due to the long waveguides (9 cm) and large number of rings (4096 rings per data waveguide). We assume 10 μW for the sensitivity of photodetectors [14]. All rings in the system must be tuned thermally, under on-die temperature variations, to maintain their resonance. We assume 1 μW tuning power per ring per K, and a temperature range of 20 K [17]. We use Orion 2.0 [33] power model to estimate the power consumption of an electrical router.

TABLE 4

Optical Losses [17]

Component	**Loss**
Coupler	1.0 dB
Splitter	0.2 dB
Non-linearity	1.0 dB
Modulator Insertion	0.001 dB
Waveguide Loss	1.0 dB/cm
Waveguide Crossing	0.05 dB
Ring Through Loss	0.001 dB/ring
Filter Drop	1.5 dB
Photo Detector	0.1 dB

Figures 40 and 41 show the power comparison among different schemes. As expected, laser power and ring heating power are dominant in all the schemes. Because the schemes with global arbitration, such as token channel and GHS, have only one shared token circulated in the network, which incurs more optical loss, they consume more laser power than the schemes with distributed arbitration such as token slot and DHS. Given that the token in GHS does not carry credit information, GHS has less laser power consumption than token channel. Among all the schemes, token slot has the lowest power consumption because the handshake schemes add additional handshake waveguides. However, the power overhead introduced by additional handshake waveguides is negligible as shown in Figure 40.

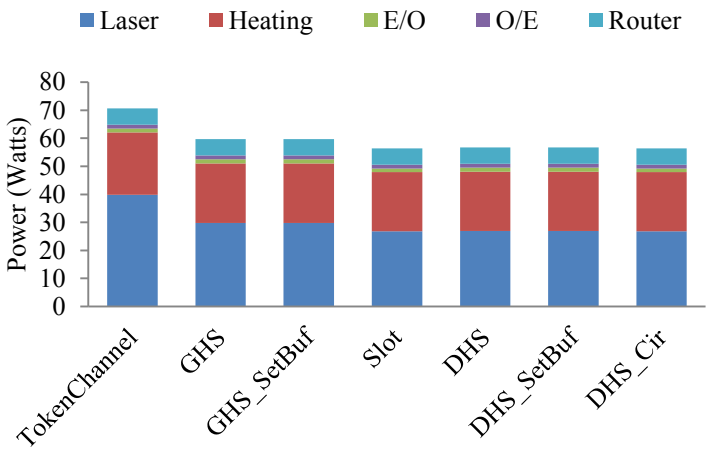

Fig. 40. Total Power Breakdown.

Figure 41 indicates the average energy consumption for delivering a packet. In nanophotonics, the modulation is done just by allowing or suppressing light. During modulation there is no need to produce light. So the circulation technique has nearly no energy overhead for delivering a packet.

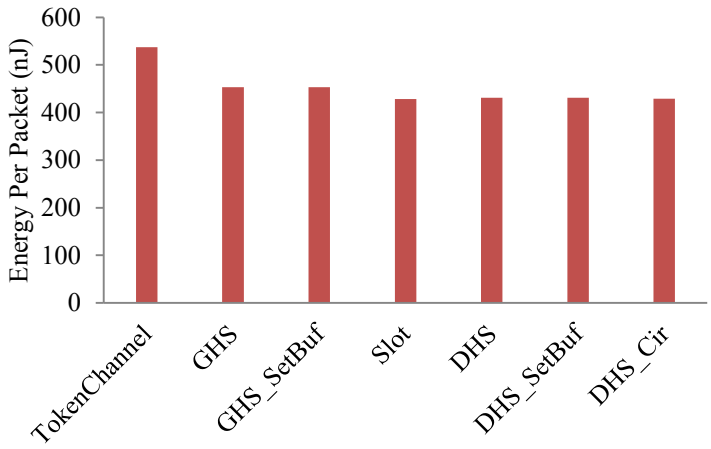

Fig. 41. Energy Consumption per Packet.

5.4 Sensitivity Study

In this section, we present variations that provide insight into the performance of the handshake schemes in different environments. We select Uniform Random as our traffic pattern and set the injection rate as 0.11. First, we evaluate the handshake schemes with various numbers of credits. Because in the handshake schemes, a token is used only for channel arbitration and no credit information is piggybacked, the performance of the handshake schemes is virtually independent of the number of credits,

as shown in Figures 42 to 46. Next, we analyze the performance of the handshake schemes with different sizes of the setaside buffer.

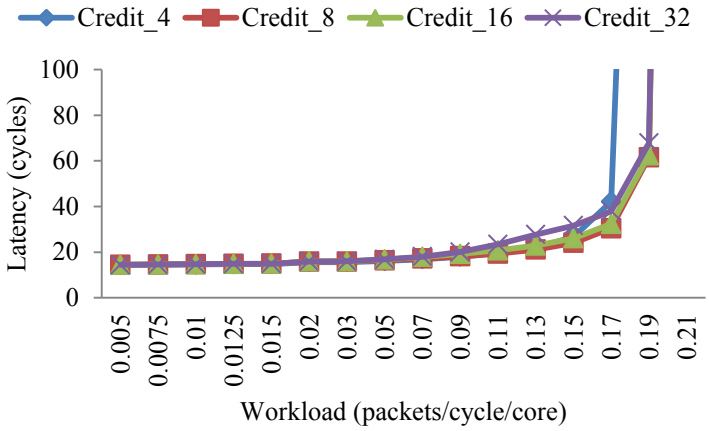

Fig. 42. Sensitivity Study of GHS.

In GHS, Figure 42, the throughput remains constant for 8, 16 and 32 credits. We see small improvement in throughput from 4 to 8. In GHS with setaside buffer, Figure - 43, the throughput remains constant for all four credit sizes.

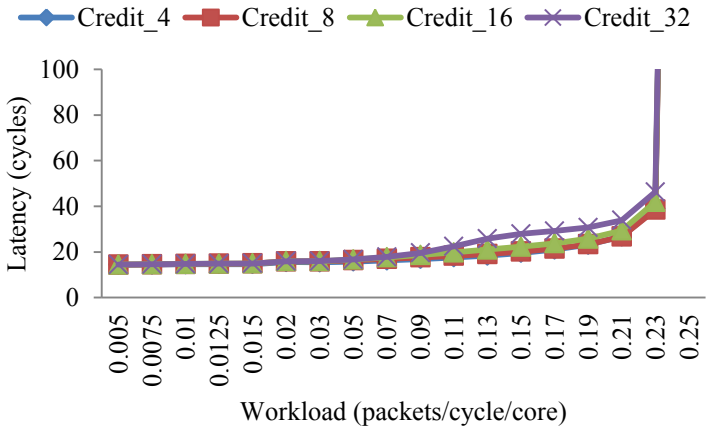

Fig. 43. Sensitivity Study of GHS with Setaside Buffer.

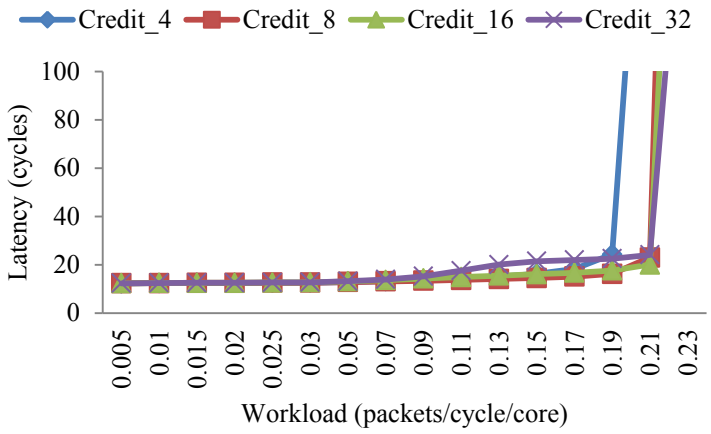

Fig. 44. Sensitivity Study of DHS.

In DHS, Figure 44, the throughput results are similar to GHS. Small increase and then constant throughput for remaning credit sizes. Figure 45 and 46, show results for DHS with setaside buffer and DHS with circulation respectively. The throughput remains constant in both schemes for all credit sizes.

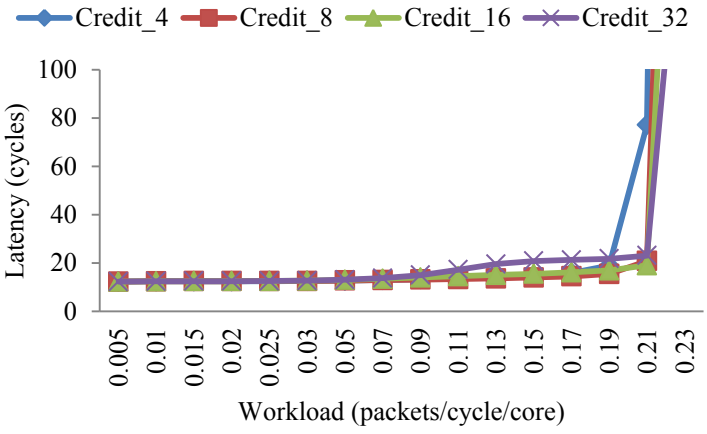

Fig. 45. Sensitivity Study of DHS with Setaside Buffer.

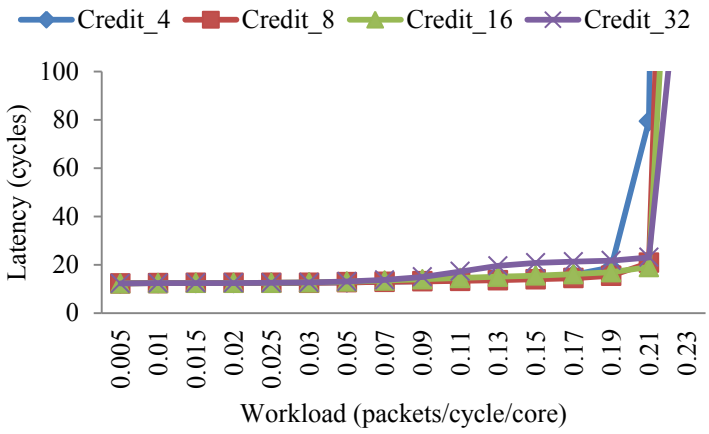

Fig. 46. Sensitivity Study of DHS with Circulation.

Figure 47 shows that the handshake schemes can produce comparable performance with only a small size of the setaside buffer. This shows almost constant latency across varying setaside buffer sizes in Uniform Random traffic.

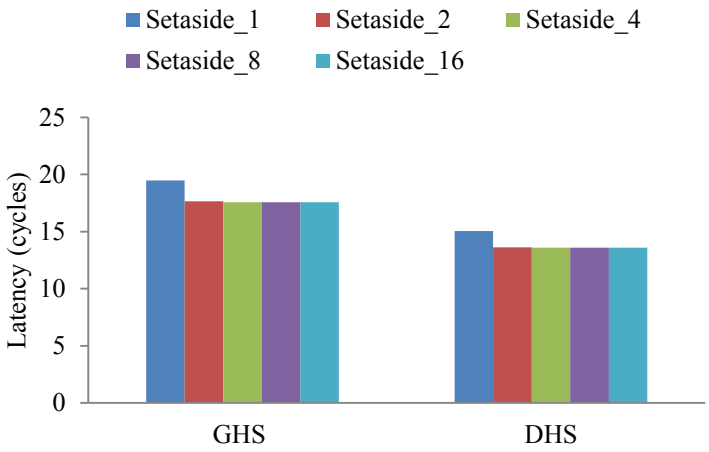

Fig. 47. Sensitivity Study of Setaside Buffer Size.

6. CONCLUSIONS

As the on-chip network size continues to increase, the bandwidth needed to conduct concurrent computation on all cores also increases hugely. Optical interconnects have been leveraged to build various on-chip networks. In this proposal, we propose handshake schemes for nanophotonic interconnects, Global Handshake (GHS) and Distributed Handshake (DHS). By getting rid of the traditional credit-based flow control, GHS and DHS reduce the average token waiting time and improve the network throughput. To remove the HOL blocking problem existing in the basic handshake schemes, we propose the setaside buffer and circulation techniques, which improve the channel utilization further. Our evaluation shows that the proposed handshake schemes improve network throughput by up to 11× under synthetic workloads. For real applications, the handshake schemes can reduce the communication latency by up to 55%. The handshake schemes add only 0.4% hardware overhead for optical components and negligible power consumption. In addition, the performance of the handshake schemes are independent of on-chip buffer space, which makes them feasible in a large scale nanophotonic interconnect design.

REFERENCES

[1] J. D. Owens, W. J. Dally, R. Ho, D. N. Jayasimha, S. W. Keckler, and L.-S. Peh, "Research Challenges for On-Chip Interconnection Networks," *IEEE Micro*, vol. 27, no. 5, pp. 96-108, 2007.

[2] R. Kumar, V. V. Zyuban, and D. M. Tullsen, "Interconnections in Multi-Core Architectures: Understanding Mechanisms, Overheads and Scaling," *Proc. 32nd Int'l Symp. Computer Architecture (ISCA)*, pp. 408-419, June 2005.

[3] A. Jose and K. Shepard, "Distributed Loss-Compensation Techniques for Energy-Efficient Low-Latency On-Chip Communication," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 42, no. 6, pp. 1415-1424, June 2007.

[4] M. F. Chang, J. Cong, A. Kaplan, M. Naik, G. Reinman, E. Socher, and S.-W. Tam, "CMP Network-On-Chip Overlaid with Multi-Band RF-Interconnect," *Proc. IEEE 14th Int'l Symp. High Performance Computer Architecture (HPCA)*, pp. 191-202, February 2008.

[5] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. P. Jouppi, M. Fiorentino, A. Davis, N. L. Binkert, R. G. Beausoleil, and J. H. Ahn, "Corona: System Implications of Emerging Nanophotonic Technology," *Proc. 35th Int'l Symp. Computer Architecture (ISCA)*, pp. 153-164, June 2008.

[6] Y. Pan, P. Kumar, J. Kim, G. Memik, Y. Zhang, and A. N. Choudhary, "Firefly: Illuminating Future Network-On-Chip with Nanophotonics," *Proc. 36th Int'l Symp. Computer Architecture (ISCA)*, pp. 429-440, June 2009.

[7] 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," *Proc. 39th IEEE/ACM Int'l Symp. Microarchitecture (MICRO)*, pp. 492-503, December 2006.

[8] M. J. Cianchetti, J. C. Kerekes, and D. H. Albonesi, "Phastlane: A Rapid Transit Optical Routing Network," *Proc. 36th Int'l Symp. Computer Architecture (ISCA)*, pp. 441-450, June 2009.

[9] A. Shacham, K. Bergman, and L. P. Carloni, "On the Design of a Photonic Network-On-Chip," *Proc. 1st Int'l Symp. Networks-on-Chip (NOCS)*, pp. 53-64, May 2007.

[10] H. Gu, J. Xu, and Z. Wang, "ODOR: A Microresonator-Based High-Performance Low-Cost Router for Optical Networks-On-Chip," *Proc. 6th IEEE/ACM/IFIP Int'l Conf. Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, pp. 203-208, October 2008.

[11] X. Zhang and A. Louri, "A Multilayer Nanophotonic Interconnection Network for On-Chip Many-Core Communications," *Proc. 47th ACM/IEEE Design Automation Conference (DAC)*, pp. 156-161, June 2010.

[12] L. Zhang, M. Song, T. Wu, L. Zou, R. G. Beausoleil, and A. E. Willner, "Embedded Ring Resonators for Microphotonic Applications," *Optics Letters*, vol. 33, no. 17, pp. 1978-1980, 2008.

[13] D. Vantrease, N. L. Binkert, R. Schreiber, and M. H. Lipasti, "Light Speed Arbitration and Flow Control for Nanophotonic Interconnects," *Proc. 42nd IEEE/ACM Int'l Symp. Microarchitecture (MICRO)*, pp. 304-315, December 2009.

[14] Y. Pan, J. Kim, and G. Memik, "FlexiShare: Channel Sharing for an Energy-Efficient Nanophotonic Crossbar," *Proc. IEEE 16th Int'l Symp. High Performance Computer Architecture (HPCA)*, pp. 1-12, January 2010.

[15] V. G. Cerf and R. E. Kahn, "A Protocol for Packet Network Intercommunication," *IEEE Transactions on Communications*, vol. 22, no. 5, pp. 637-648, May 1974.

[16] M. Hayenga, N. D. E. Jerger, and M. H. Lipasti, "SCARAB: A Single Cycle Adaptive Routing and Bufferless Network," *Proc. 42nd IEEE/ACM Int'l Symp. Microarchitecture (MICRO)*, pp. 244-254, December 2009.

[17] A. Joshi, C. Batten, Y.-J. Kwon, S. Beamer, I. Shamim, K. Asanovic, and V. Stojanovic, "Silicon-Photonic Clos Networks for Global On-Chip Communication," *Proc. 3rd ACM/IEEE Int'l Symp. Networks-on-Chip (NOCS)*, pp. 124-133, May 2009.

[18] J.-H. Ha and T. M. Pinkston, "A New Token-Based Channel Access Protocol for Wavelength Division Multiplexed Multiprocessor Interconnects," *Journal of Parallel and Distributed Computing (JPDC)*, vol. 60, no. 2, February 2000.

[19] A. K. Kodi and A. Louri, "A Scalable Architecture for Distributed Shared Memory Multiprocessors Using Optical Interconnects," *Proc. 18th Int'l Parallel and Distributed Processing Symp. (IPDPS)*, pp. 11, April 2004.

[20] C. Qiao and R. G. Melhem, "Time-division Optical Communications in Multiprocessor Arrays," *Proc. ACM/IEEE Conf. Supercomputing*, pp. 644-653, November 1991.

[21] ANSI/IEEE, "Token Ring Access Method and Physical Layer Specifications," *IEEE Standard 802.5-1989*, 1989.

[22] C. Batten, A. Joshi, J. Orcutt, A. Khilo, B. Moss, C. Holzwarth, M. Popovic, H. Li, H. I. Smith, J. L. Hoyt, F. X. Kartner, R. J. Ram, V. Stojanovic, and K. Asanovic, "Building Manycore Processor-to-DRAM Networks with Monolithic Silicon Photonics," *Proc. 6th IEEE Symp. High Performance Interconnects (HOTI)*, pp. 21-30, August 2008.

[23] P. S. Magnusson, M. Christensson, J. Eskilson, D. Forsgren, G. Hallberg, J. Hogberg, F. Larsson, A. Moestedt, and B. Werner, "Simics: A Full System Simulation Platform," *Computer*, vol. 35, no. 2, pp. 50-58, February 2002.

[24] D. Kroft, "Lockup-Free Instruction Fetch/Prefetch Cache Organization," *Proc. 8th Int'l Symp. Computer Architecture (ISCA)*, pp. 81-87, May 1981.

[25] C. Kim, D. Burger, and S. W. Keckler, "An Adaptive, Non-Uniform Cache Structure for Wire-Delay Dominated On-Chip Caches," *Proc. 10th Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 211-222, October 2002.

[26] P. Kapur and K. C. Saraswat, "Comparisons between Electrical and Optical Interconnects for On-Chip Signaling," *Proc. IEEE Int'l Interconnect Technology Conference (IITC)*, pp. 89-91, June 2002.

[27] Specomp 2001 benchmark suite. http://www.spec.org/omp/.

[28] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The PARSEC Benchmark Suite: Characterization and Architectural Implications," *Proc. 17th Int'l Conf. Parallel Architectures and Compilation Techniques (PACT)*, pp. 72-81, October 2008.

[29] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 Programs: Characterization and Methodological Considerations," *Proc. 22nd Int'l Symp. Computer Architecture (ISCA)*, pp. 24-36, June 1995.

[30] Nas Parallel Benchmarks. http://www.nas.nasa.giv/Resources/Software/npb.html.

[31] Specjbb 2000 benchmark. http://www.spec.org/jbb2000/.

[32] H.-S. Wang, L.-S. Peh, and S. Malik, "A Power Model for Routers: Modeling Alpha 21364 and InfiniBand Routers," *Proc. 10th Symp. High Performance Interconnects (HOTI)*, pp. 21-27, August 2002.

[33] A. B. Kahng, B. Li, L.-S. Peh, and K. Samadi, "ORION 2.0: A Fast and Accurate NoC Power and Area Model for Early-Stage Design Space Exploration," *Proc. Conf. Design, Automation & Test in Europe (DATE)*, pp. 423-428, April 2009.

VITA

Name: Jagadish Chandar Jayabalan

Address: Texas A&M University,
H.R. Bright Building, Room 335,
College Station, TX 77843

Email Address: jagadishchandar@gmail.com

Education: B. Tech., National Institute of Technology, Tiruchirappalli, 2007
M. S., Texas A&M University, College Station, 2012