FLY-OVER: A LIGHT-WEIGHT DISTRIBUTED ROUTER POWER-GATING MECHANISM FOR ENERGY-EFFICIENT INTERCONNECTS

A Thesis

by

NINGYUAN WANG

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

MASTER OF SCIENCE

Chair of Committee,	Eun Jung Kim
Co-Chair of Committee,	Peng Li
Committee Member,	Seong Gwan Choi
Head of Department,	Miroslav M. Begovic

May 2015

Major Subject: Computer Engineering

ABSTRACT

Scalable Networks-on-chip (NoCs) have become the de facto interconnection mechanism in large scale Chip Multiprocessors. Not only are NoCs devouring a large fraction of the on-chip power budget but static NoC power consumption is becoming the dominant component as technology scales down. Hence reducing static NoC power consumption is critical for energy-efficient computing. Previous research has proposed to power-gate routers attached to inactive cores so as to save static power, but they either required centralized decision making and global network knowledge or a non-scalable escape ring network. In this paper, we propose Fly-Over (FLOV), a light-weight distributed mechanism for power gating routers, which encompasses FLOV router microarchitecture and a partition-based dynamic routing algorithm to maintain network functionality. With simple modifications to the baseline router microarchitecture, FLOV can facilitate fly-over links over power-gated routers. The proposed routing algorithm provides best-effort minimal path routing without the necessity for global network information. We evaluate our scheme using both unicast and multicast synthetic workloads as well as real workloads from PARSEC 2.1 benchmark suite. The results show that FLOV can achieve 19.2% latency reduction and 16.9% total power savings.

To my parents and friends for their encouragement and support.

ACKNOWLEDGEMENTS

I wish to express my profound gratitude to my advisor Dr. Eun Jung Kim for the support and encouragement throughout the research. I am grateful to her for having provided me an opportunity to work with her in the field of Networks-On-Chip.

I thank my graduate committee members, Dr. Seong Gwan Choi and Dr. Alex Sprintson for their willingness to serve on my committee.

I also need to thank for Rahul, Jiayi Huang, and Vivek Kumar, who are my lab mates in the High Performance Com- puting Lab (HPCL). They helped me with simulations, ideas, writing and suggestions.

Finally, I thank my parents, family and friends for their loving support and words of encouragement.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

Chip Multiprocessors (CMPs), scaled to 100s and 1000s of cores, are being touted as the future solution for extracting huge performance gains using parallel programming paradigms. This is possible, as stated by Moore's law [23], because of shrinking transistor sizes, allowing for denser on-chip packaging. However the failure of Dennard Scaling [11], supply voltage not scaling down with the transistor size, means that all the components on the chip cannot be run simultaneously without breaking the power and thermal constraints. Thus future CMP designs will have to work under stricter power envelops. Scalable Networks-on-chip (NoC), like 2D meshes, have become the de facto interconnection mechanism on these large CMPs. Recent studies [29, 15, 14] have shown that NoCs consume a significant portion, 10 to 36%, of the total on-chip power budget. Hence power-efficient NoC designs are of the highest priority for the power-constrained future CMPs.

Static power consumption of the on-chip circuitry is increasing at an alarming rate with the scaling down of feature sizes and chip operating voltages towards near-threshold levels. Previous studies [7, 2, 4, 28, 25] have shown that the percentage of static power in the total NoC power consumption increases from 17.9% at 65nm, to 35.4% at 45nm, to 47.7% at 32nm and to 74% at 22nm. According to this trend as we reach towards sub-10nm feature sizes, static power will become the major portion of the NoC power consumption.

Power-gating, cutting off supply current to idle chip components, is an effective circuit-level technique which can be used to mitigate the worsening impact of on-chip static power consumption. Due to the low average core utilization in most modern workloads [3, 8], significant number of studies have proposed efficient mechanisms for power-gating cores with marginal impact on performance [1, 20, 21]. Recent studies [24] have proposed power-gating selected router components in a fine-grained fashion using topology reconfiguration. However limited research [26, 4, 6, 5] has been done regarding mechanisms for power-gating routers, which will reduce the NoC static power consumption.

On the other hand, for cache-coherent protocols and and other programming models need one-to-many communications, it is neccessary to have a efficient multicast routing scheme in NoCs. Previous work [30, 17] have provided solutions for multicast routing on NoCs.

However, currently there is no studies proposing multicast scheme for NoCs with power-gated routers. In this paper, we will show that our Fly-Over scheme can provide efficient support for multicast workload because this scheme make decision based on section of the destinations and therefore the replication time can be delayed.

Router Parking[26] and NoRD [4] have been proposed to seamlessly power-gate routers on mesh topology while maintaining network connectivity. But these mechanisms either require a centralized decision making mechanism[26] or necessitate an escape ring network for routing functionality[4]. Both these approaches are not scalable to large network sizes, while centralized decision making creates a huge synchronization overhead and becomes a single point of failure if the centralized fabric manager goes down.

We propose Fly-Over (FLOV), a light-weight distributed mechanism which eliminates the need for a centralized decision making mechanism to power-gate routers. Such a distributed power-gating mechanism may create interconnect partitions without communication paths. In order to maintain the network connectivity, the FLOV router design provides fly-over links in power-gated routers which enable incoming packets to travel straight through. Since a power-gated router does not have routing functionality and incoming packets can only travel in the same direction, without prior information about such power-gated routers in a packet's path, localized routing decisions cannot ensure the packet's deliverability to the destination. Therefore it is necessary to develop a routing algorithm which ensures a routing path between any source and destination pairs irrespective of the configuration of power-gated routers in the network.

For this, we propose a dynamic routing algorithm which ensures network routing functionality without the need for any global NoC information. The routing algorithm dynamically decides the output direction based on the destination and the status of the neighboring routers.

We evaluate our FLOV scheme using an in house cycle-accurate interconnect simulator and compare against the Router Parking (RP) mechanism. Our evaluations using nine PARSEC benchmark network traces show that we have on average 19.2 % further latency reduction and further total interconnect power consumption reduction of around 16.9 % compared to RP.

The rest of this paper is organized as follows. We briefly summarize the related work in section 2. We describe the baseline NoC router microarchitecture and FLOV router in section 4. In section 5, we explain the routing algorithm framework. We evaluate our design in section 6. Finally, we draw conclusions in section 7.

2. RELATED WORK

In recent years significant research has been performed in applying the power gating technique in NoCs for power savings [19] [1]. Kim et al. [18] proposed a dynamic link shutdown (DLS) technique together with dynamic voltage scaling to save link energy. Soteriou et al. [27] proposed a power-aware network which reduces static power consumption by monitoring the link utilization and power gating the underutilized links. Matsutani et al. [22] proposed a component-based power gating technique that individually controls the power supply to different components in an ultra fine-grained way. These approaches work well to reduce the static power consumption. However, they only power gate certain components of the Router. In addition, energy consumption and latency to wake up the components and the routers due to the node-router dependence can be an overhead, which may hide the advantages of power gating and also degrade the performance.

Chen et al. [4] proposed a node-router decoupling (NoRD) approach to leverage the independence of power-gating the core and the attached router. To this order they provide a decoupling bypass route which connects the ejection and injection channels to form a bypass link to the router. The decoupling bypass links ensure network connectivity even for the extreme cases (all routers are turned off) by using an escape ring network. However, a bypass ring is not scalable to large network sizes. Chen et al. [6] introduced performance-aware, non-blocking power-gating scheme which wakes up powered-off routers along the path of packet in advance thereby preventing packet from suffering router wakeup latency. Samih et al.[26] proposed Router Parking (RP) to power-gate as many routers as possible when their attached cores are sleeping while maintaining network connectivity. Router parking dynamically parks (power-gates) the routers based on the network traffic to maintain a balanced trade-off between power saving and performance. However this scheme requires dedicated channel to communicate with Fabric Manager (FM) and typically takes long time to reconfigure the network which may suspend the network operation in between. Parikh et al.[24] came up with power-aware routing and topology reconfiguration to minimize detours while selected components in routers are power-gated. This feedback-based mechanism is slow and reconfiguration takes place only on per epoch basis. Power-gating components inside the router in a fine-grained fashion requires additional circuitry.

3. NOC ARCHITECTURE

3.1 NoC Overview

As the number of cores increases rapidly, a high-bandwidth and scalable coomunication fabric to connect them becomes critically important[16]. Therefore, people on-chip networks are replacing traditional crossbars and buses. The most popular topology for on-chip network is Mesh. In Mesh topology, every Processing Element is connected with a Network Interface and a Router. The basic unit of the traffic is called flit. Usually one flit has the same size with the link width between routers. A packet may has multiple flits. The first flit in a packet is called head flit, which usually records the source, destination, and number of flits in the packet.

3.2 Baseline NoC Router Architecture

The Baseline microarchitecture is based on the state-of-the-art 4-stage wormhole-switched router. Figure 3.1 shows the main building blocks of the baseline NoC router input buffers, routing computation logic, VC allocator, switch allocator and crossbar. In a mesh topology, there are usually 5 input and 5 output ports for a single router. They are East, West, North, South input/output ports connected with neighbor routers and injection/ejection ports connected wth Processing Element (PE). For a single input port, there are buffers served as input buffer. To provide efficiency and deadlock freedom in the routing process, there are usually several Virtual Channels associated with one input port[9]. The virtual channels are buffers providing the decoupling capacity for multiple packets within the same input port. The processing inside a router is pipelined into 4 stages: Routing Computation (RC), VC Allocation (VA), Switch Allocation (SA) and finally Switch Traversal (ST). The output port to which a packet should go is computed in the RC stage based on the destination information in the header flit. In the VA stage, an available VC in the downstream router is allocated to this packet based on the credit information. The SA stage is the arbitration between the inputs and outputs of the crossbar. The flits successfully granted in the SA stage traverse the crossbar in the ST stage. Link traversal (LT) is external to the router pipeline and is also assumed to take one clock cycle. Wormhole switching is used along with credit-based flow control.

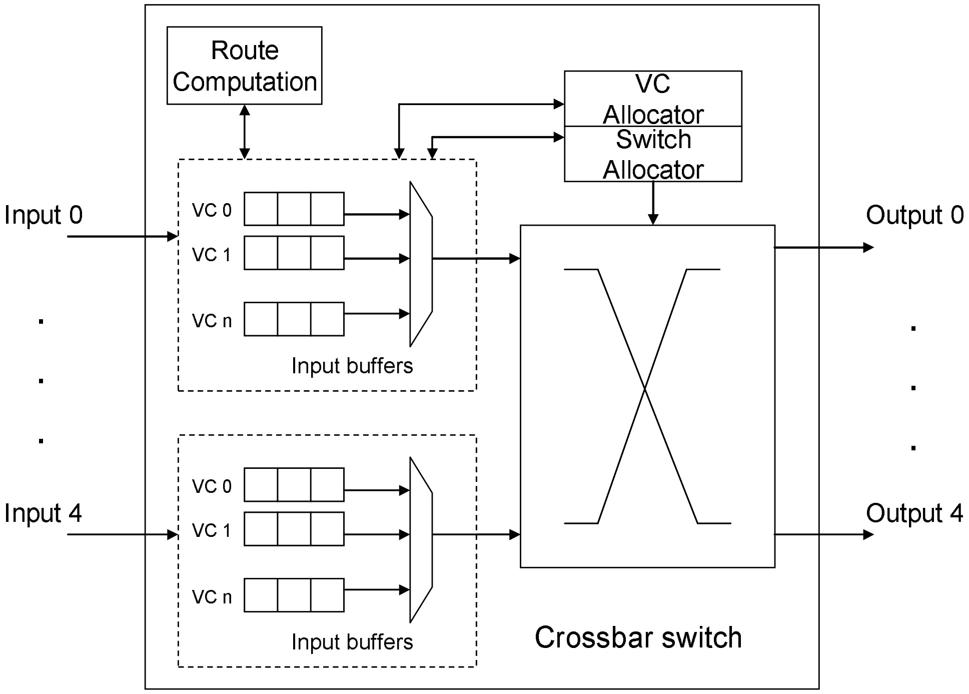


Figure 3.1: Baseline NoC Router Architecture.

4. FLOV ROUTER ARCHITECTURE AND MECHANISM

This section proposes the FLOV router and describes the distributed power-gating mechanism.

4.1 FLOV Router Architecture

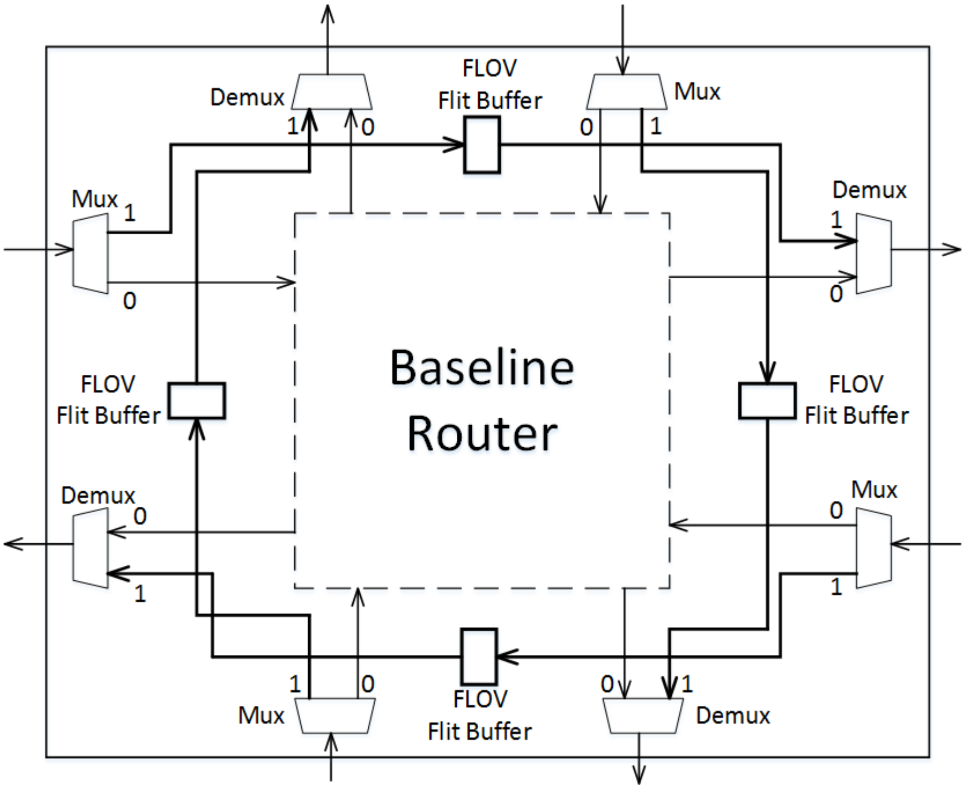


Figure 4.1: FLOV Router Architecture.

As shown in Figure 4.1, the FLOV router architecture has multiplexers and demultiplexers added to input/output links, in addition to a single flit buffer in each direction. When a FLOV router is powered-on, it functions like the baseline 4-stage wormhole router, and the muxes/demuxes are set to 0 as well as the single flit buffers are power-gated. When the router is power-gated, all the components of the baseline router are power-gated and the muxes/demuxes are set to 1 to activate the fly-over links.

4.2 Distributed Power-Gating Mechanism

This section explains how the distributed power-gating mechanism works. Initially, as soon as a core is powered down, its attached FLOV router sends a signal to the neighboring routers so that they cannot initiate new packet delivery. Then the FLOV router checks its input buffers for any residing flits. If there is a flit, it is delivered to a downstream router normally and its upstream router is notified to send the remaining flits of the corresponding packet to this router. After emptying all the input buffers, the FLOV router power-gates itself by shutting down the baseline router portion. At the same time, all the muxes/demuxes are switched to 1, and the FLOV router sends a signal to all the neighboring routers to new packet transmission.

Once the FLOV router is power-gated, a flit coming into the router is stored in the FLOV flit buffer without any routing/arbitration. In the next cycle it is delivered to a designated (escape) VC in the downstream router. If a packet uses the FLOV buffer, it will always use a designated (escape) VC in the powered-on FLOV routers to avoid deadlock as explained further in Section 5.

When the core becomes active later, its FLOV router also turns on the baseline router portion after signaling the neighboring routers, to stop a new packet delivery. Only packets in the middle of transmission using the FLOV links are allowed to be transmitted. After emptying all FLOV flit buffers, the muxes/demuxes are switched to 0 and a signal is sent to the neighboring routers to resume normal packet transmission.

Note that, in this mechanism, an FLOV router does not need any global information and there is no centralized decision making unit to control these operations. Each FLOV router decides to shut itself down based on its local information, thus making the NoC more robust and efficient.

5. DYNAMIC ROUTING ALGORITHM DESIGN

In this section the overall FLOV NoC architecture is introduced and then the dynamic routing algorithm is explained.

5.1 FLOV NoC Architecture

Figure 5.1 shows a (4×4) 2D mesh network topology with the proposed FLOV routers. The pattern-shaded routers (3,7,11 and 15) are connected to the memory controller (MC) nodes that should be never power-gated.[1] Therefore we use the baseline routers for these nodes. All other routers are FLOV routers that are connected to processor nodes and can be power-gated if the core is powered down. Maintaining connectivity in the network without any global information is critical to our scheme. This is ensured by a combination of keeping all the routers in the last column powered-on and the routing algorithm. The fly-over links of the power-gated routers along with one reserved VC of each powered-on router form an escape sub-network. This escape sub-network is not only used for deadlock recovery but also is an integral part of the routing algorithm. The routing algorithm, described in detail below, routes packets in the regular VCs using a minimal heuristic whereas packets that cannot be routed to their destinations due to power-gated intermediate routers are directed to the escape sub-network.

5.2 Dynamic Routing Algorithm

The proposed routing algorithm consists of routing for packets in the regular VCs and routing for packets in the escape sub-network. The router will execute a different routing algorithm for packets in regular VCs and escape VCs. A packet in

[1] MC nodes can be located in other places. According to MC placement, the routing algorithm may be slightly different.

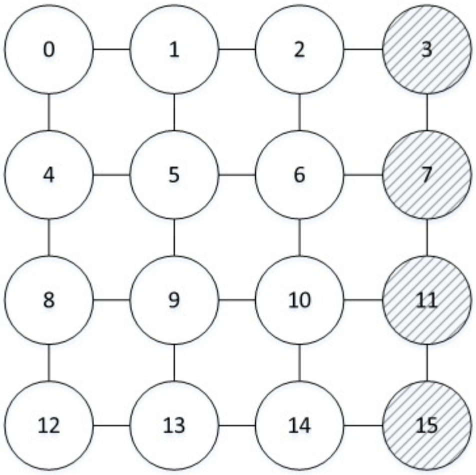


Figure 5.1: FLOV NoC Architecture.

a regular VC can be sent to an escape VC when required by the routing algorithm but the vice-versa is prohibited to avoid deadlock. Note that routing computation is performed in powered-on routers, while power-gated routers only forward packets without changing the direction.

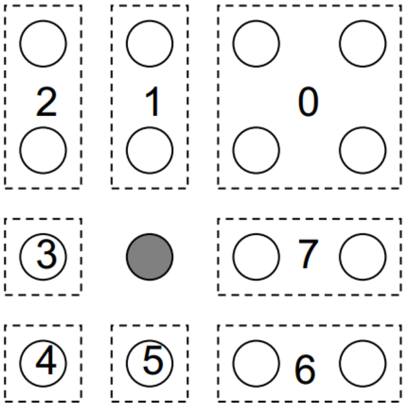
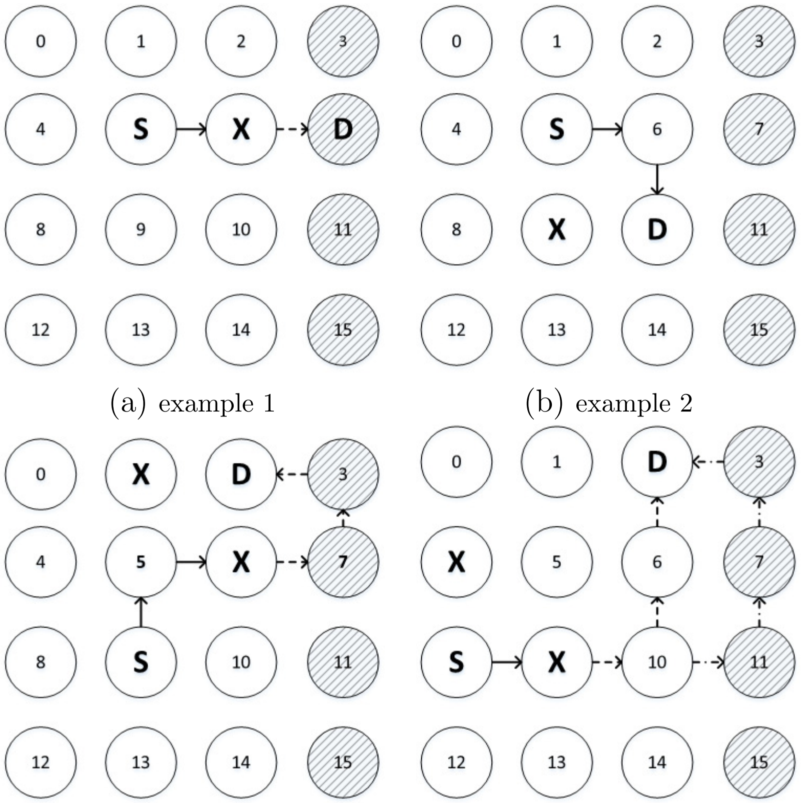


Figure 5.2: Destination Partitioning in a 2D Mesh Network.

We propose a partitioned-based dynamic routing algorithm for the packets in



(a) example 1 (b) example 2

(c) example 3 (d) example 4

Figure 5.3: Routing Algorithm Examples: X indicates a power-gated router.

the regular VCs. Each router partitions the network into sections as shown in Figure 5.2. The routing decision is based on two variables, the partition section which the destination falls into and the status of the neighboring routers. For packets with destinations in sections 1, 3, 5, and 7, the router will send them directly to North, West, South, and East downstream routers, respectively. This is because even in case of power-gated downstream routers, the fly-over links will ensure the connectivity to the destination.

For packets with destinations in sections 0, 2, 4, and 6, the route involves a turn towards the destination. Hence the algorithm has to consider the fact that the packets might not be able to make a turn, if all the downstream routers are powered-gated.

Therefore if the corresponding neighbor router in the Y direction is powered-on, the packet will be sent to this router assuming YX routing. If this neighbor router is power-gated, the router will check the status of corresponding neighbor router in the X direction. If the X neighbor router is powered-on, the router will send the packet to this router.

In case both the routers are power-gated, a viable route to the destination cannot be promised since the farther downstream routers' status is unknown to this router. Therefore the packet will be directed to the escape VC of the downstream router in the X direction towards MC node routers (East). The packet is not sent to the router in Y direction because, in the worst case, if all the downstream routers in the Y direction are powered off the packet will not be able to make a turn and hence we lose connectivity in the network. In contrast once the packets are directed into the escape VCs in the X direction we can guarantee that the packet will be able to make a turn towards the destination in the powered-on MC node router of the corresponding row.

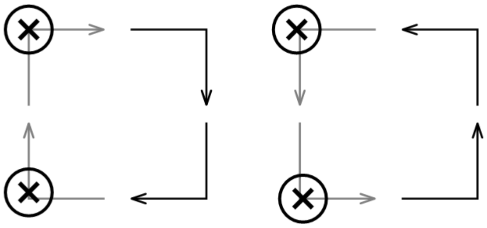


Figure 5.4: Turn Model.

Since power-gated routers only have a single flit buffer in any one direction, a packet going through a power-gated router (fly-over link) should remain in the escape sub-network until it reaches its destination. This means that when the packet goes to the next downstream router using an escape VC, even though the router is powered-on, it has to be placed in the escape VC. This is to avoid deadlock in the escape sub-network, since if a packet is allowed to re-enter the regular VCs, it might backpressure deadlock in the escape VC.

The routing algorithm for powered-on routers in the escape sub-network is also based on the partitioning from Figure 5.2. Packets with destinations in sections 1, 3, 5, and 7, will be sent directly to North, West, South, and East, respectively. Packets that have destinations in sections 0, 2, 4, and 6, should be sent to East where the baseline routers are located. This is because, in the worst case of all downstream routers being powered off, the routers in the rightmost column are still powered-on[2]. Hence the packet will reach one of these routers where it will make a turn towards the destination's row, which ensures packet delivery. Deliverability can be ensured in this way.

The proposed dynamic routing algorithm is explained further using examples from Figure 5.3.

- In Figure 5.3 (a) the destination is in section 7 of the source router's partitions and hence even though the next router is power-gated the packet is forwarded using the fly-over link.

- In Figure 5.3 (b) the destination is in section 6, so the routing algorithm first checks for Router 9's status. Since Router 9 is power-gated the packet is sent to Router 6, which is powered-on, which will then in turn route the packet to its destination.

- Figure 5.3 (c) the destination is in section 0 and hence the router 5's status is checked. Since it is powered-on the packet is forwarded to router 5. Router 5

[2] Further optimization can be possible, which is future work.

then executes the same logic and since router 1 and router 6 are both powered off the packet has to be sent to the escape VC in router 6. The packet is then forwarded from router 6 to router 7 which is powered-on. Router 7 then routes the packet to the escape channel in router 3 where it makes another turn towards the destination. It needs to be observed that the dashed lines indicate that the packet is in the escape VCs. The packet enters the escape VCs in router 6 and hence has to remain in the escape sub-network until it reaches the destination.

- Figure 5.3 (d) shows a case where we can use an optimization in the escape VC routing. Once the packet reaches router 10 which is powered-on, since it is in the escape VC the usual naive routing would route it via routers 11->7->3->D. But the optimization in our algorithm ensures that if router 6 is powered on, the packet can make a turn to the Y direction since the destination is in the section 1 of router 10's partitioning.

5.3 Multicast Support

As mentioned before, the future chip-multiprocessor design needs a bandwidth-efficient routing for multicast traffic. However, there is even extra difficulty of routing on an irregular Mesh with some routers power-gated.

In our routing algorihtm, the routing decisions are made based on which section does the destination locate. This partition is similar to the partition in Recursive Partition Multicast (RPM)[30]. Therefore, for destinations within the same section, same routing decision will be made for them. Therefore, the replication process will be delayed and bandwidth resources are efficiently used. From this observation, we can conclude that our Fly-over routing scheme can support multicast traffic without major modification.

To support multicast, we should make modifications described as below:

- Adding destination list in the head flit of the packet.
- Add logic for replica management, and there is no extra storage overhead for multicast because replication take place in ST stage.

5.4 Deadlock Avoidance

For a reliable design of NoC, the deadlock probelm should also be taken into account. That is because applications require the network to be deadlock-free.

A deadlock occurs when there is a cylic dependence among the path of multiple packets[16]. In this case, all packets in this path are waiting for the buffer occupied by other packets in this path, therefore peventing any packets from making forward progress.

Previous works are proposed to provide efficient routing algorithm in the network. A network can be deadlock-free if there is no channel dependency in a cyclic manner.[10] We can describe a routing algorithm by indicating which turns are permitted in the network. As illustrated in the Figure, there are 8 possible turns in a 2D mesh network. Allowing all of them may cause a cyclic waiting of buffer in the network, leading to network deadlcok. To provide deadlock-free routing, some routing algorithm may forbid part of the turns to break the cyclic dependence. For example, the X-Y routing algorithm does not allow a packet traveling north or south to make a turn to east or west. So 4 of 8 turns are forbidden in the X-Y routing, providing the deadlock freedom of X-Y routing alorithm.

We can also describile our escape routing algorithm using the turn model. As shown in the Figure 5.4, our esacpe model only allow 4 kinds of turns. This breaks the cyclic dependence of channel, therefore our escape routing algorithm is deadlock-free.

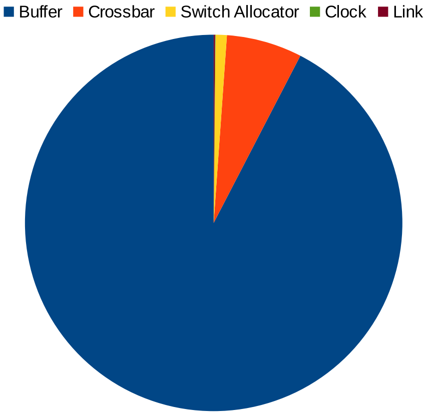


Figure 5.5: The Router Static Power Decomposition

The proposed routing algorithm in the regular VCs is not necessarily deadlock-free. In order to provide the deadlock freedom to the whole routing algorithm, we set up a time out machanism for every router. The number of waiting cycles of a packets in a router is limited to a certain threshold. If the number of cycles exceed the threshold. The packet will be sent to escape channel. The escape resources are comprised of reserved channel in powered-on router and bypass link in powered-off router. According to Duato's Protocol[12], our routing algorithm is deadlock-free.

5.5 Overhead Analysis

In this section we discuss the area and power overhead incurred by the proposed scheme. We extracted the power comsumption result from DSENT [28] model. Under 22 nm technology and 0.06 average injection rate, the static router power decomposition is shown in Figure 5.5. The modifications proposed to the router microarchitecture include 4 multiplexers and 4 demultiplexers in addition to four single flit buffers. The multiplexer and demultiplexer selection signals are only toggled when the router powers on or off and hence the logic needed for the select signals is minimal. The overall area overhead for a single router in 32 nm technology is quantized at 2.8×10^{-3} mm^2 which is 3% of the baseline router area. Every router has to keep track of the status of the four neighboring routers and this would incur a minimal 4 bit overhead per router. The power consumption overhead is accounted for in the DSENT [28] model and hence is included in the power consumption evaluation results in the next section.

6. EXPERIMENTAL EVALUATION

In this Section we evaluate our FLOV scheme by comparing its static, dynamic and total power consumption in addition to the NoC latency, with the Router Parking scheme [26].

6.1 Experimental Methodology

We use a cycle-accurate network simulator that models all the router pipeline stages and link latencies. DSENT [28] is used to estimate the static and dynamic power consumption of the interconnect components with 50% switching activity in 32 nm technology. A 2 GHz clock frequency is assumed for the routers and links. Table 6.1 summarizes the simulated configuration. We use both synthetic and real workloads to evaluate the performance and power-savings of FLOV against the baseline interconnect with no router power-gating (Baseline) and Router Parking (RP), as it is the more recent work which deals with power gating routers attached to inactive (sleeping) cores. We use Uniform Random and Tornado traffic for synthetic workloads and nine benchmarks from the PARSEC benchmark suite [3].

6.2 Synthetic Workload Evaluation

Figure 6.1-6.6 summarizes the simulation results using Uniform Random traffic. Similarly Figure 6.7-6.12 shows the results using Tornado traffic. Each row shows the latency, dynamic and total power consumption for a given injection rate, respectively. In each graph the top row is for the injection rate of 0.02 and the bottom row is for the injection rate of 0.08 flits/cycle/router. The static power consumption for Uniform Random and Tornado are shown in Figure 6.14.

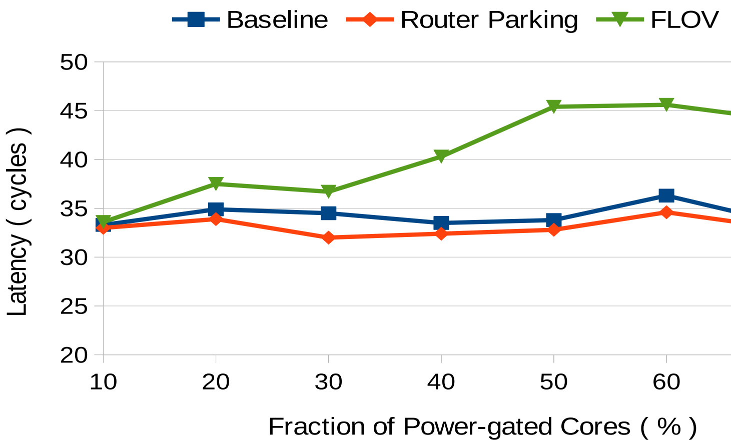


Figure 6.1: Average NoC Latency Comparison for Injection Rates of 0.02 flits / node / cycle under Uniform Random Traffic.

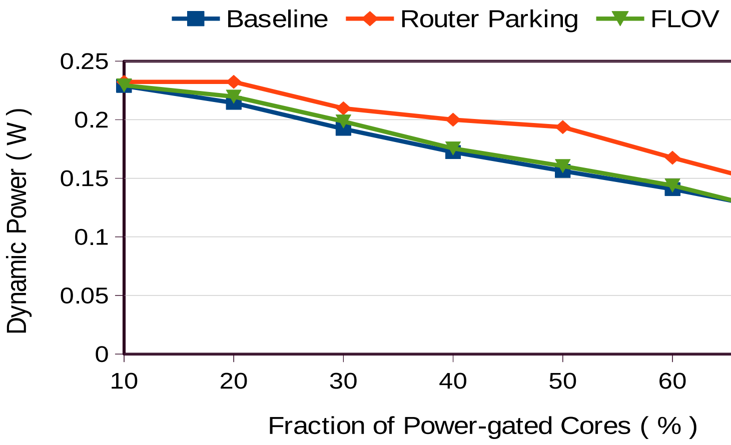


Figure 6.2: Average NoC Dynamic Power Comparison for Injection Rates of 0.02 flits /node / cycle under Uniform Random Traffic.

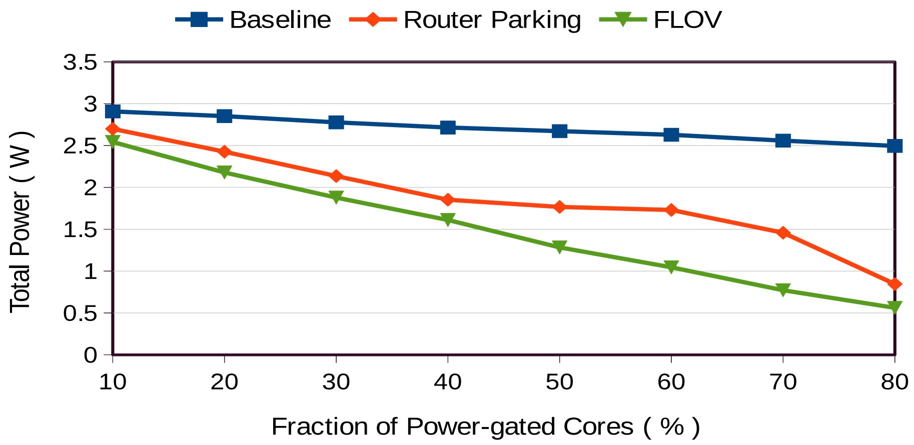


Figure 6.3: Average NoC Total Power Comparison for Injection Rates of 0.02 flits / node / cycle under Uniform Random Traffic.

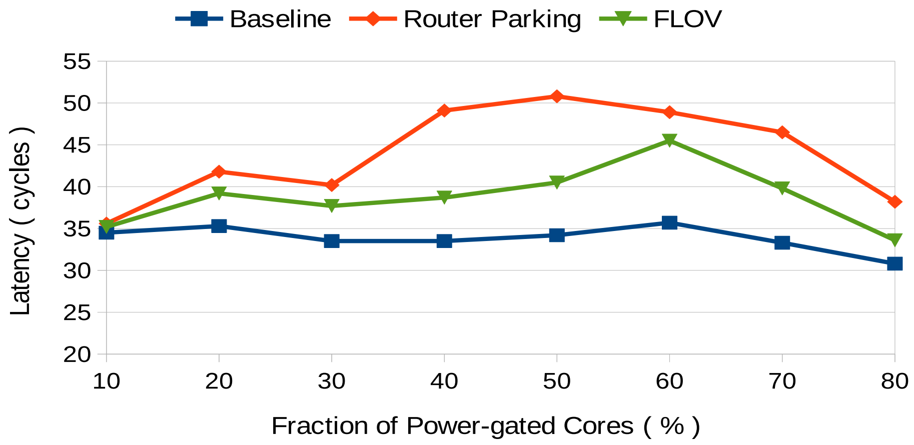


Figure 6.4: Average NoC Latency Comparison for Injection Rates of 0.08 flits / node / cycle under Uniform Random Traffic.

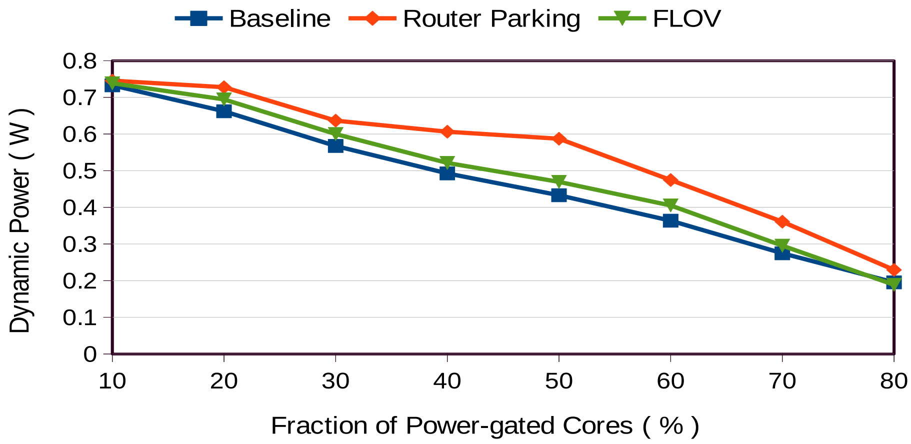


Figure 6.5: Average NoC Dynamic Power Comparison for Injection Rates of 0.08 flits / node / cycle under Uniform Random Traffic.

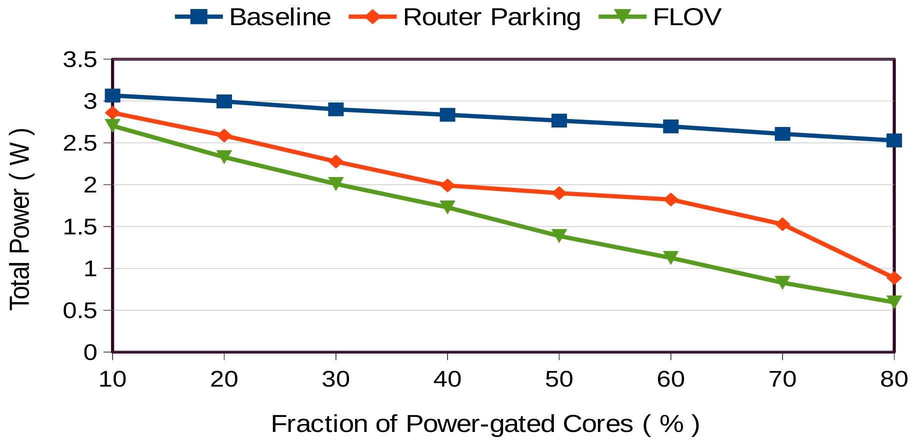


Figure 6.6: Average NoC Total Power Comparison for Injection Rates of 0.08 flits / node / cycle under Uniform Random Traffic.

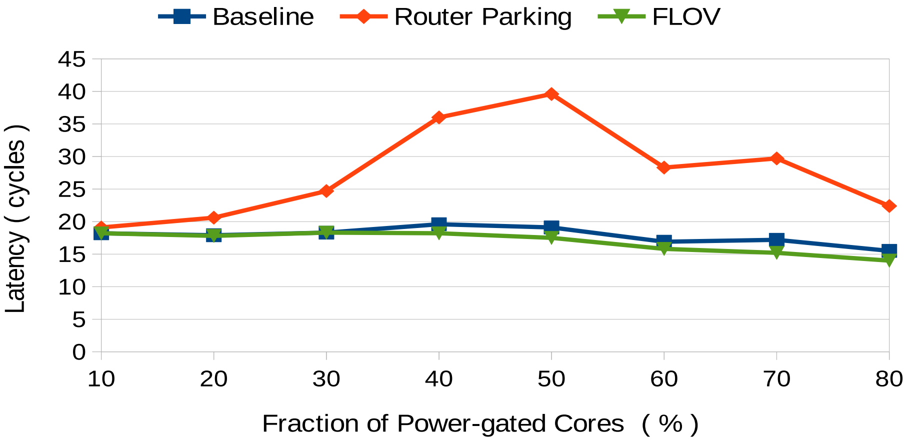


Figure 6.7: Average NoC Latency Comparison for Injection Rates of 0.02 flits / node / cycle under Tornado Traffic.

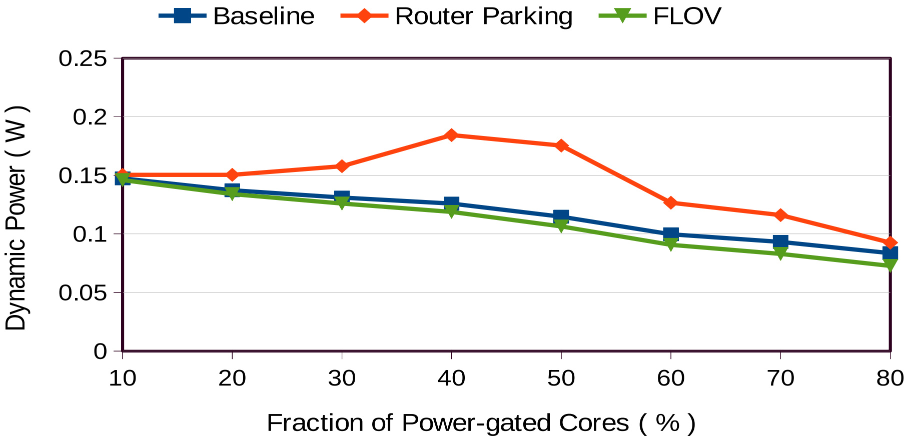


Figure 6.8: Average NoC Dynamic Power Comparison for Injection Rates of 0.02 flits / node / cycle under Tornado Traffic.

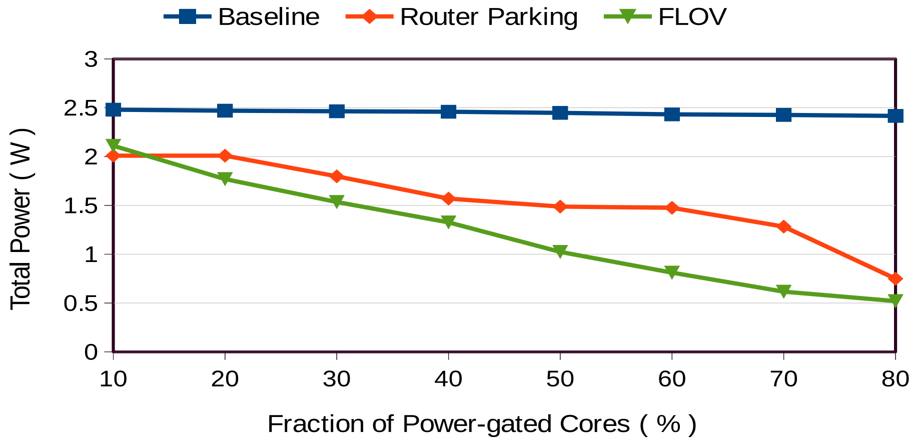


Figure 6.9: Average NoC Total Power Comparison for Injection Rates of 0.02 flits / node / cycle under Tornado Traffic.

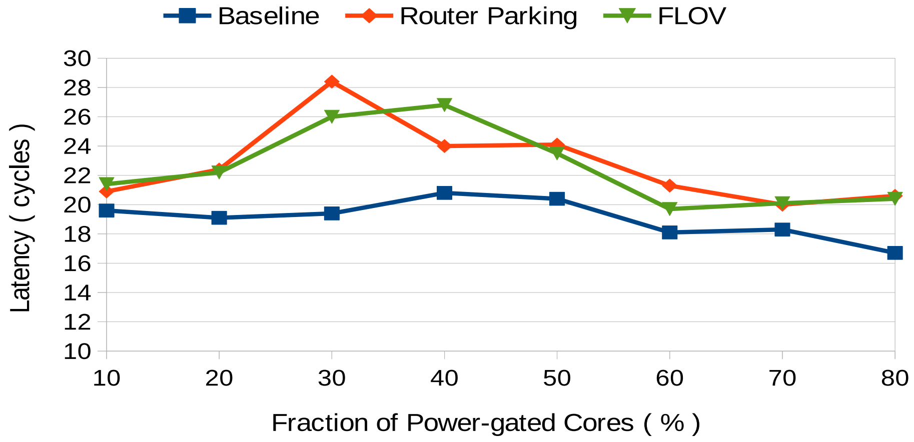


Figure 6.10: Average NoC Latency Comparison for Injection Rates of 0.08 flits / node / cycle under Tornado Traffic.

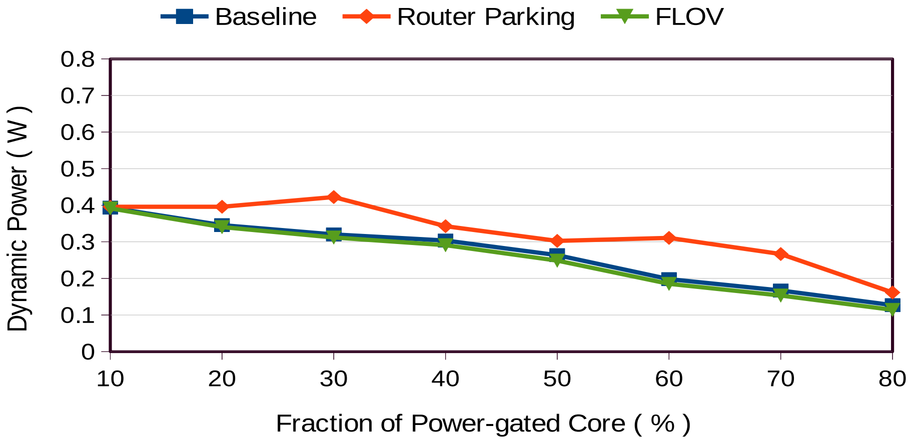


Figure 6.11: Average NoC Dynamic Power Comparison for Injection Rates of 0.08 flits / node / cycle under Tornado Traffic.

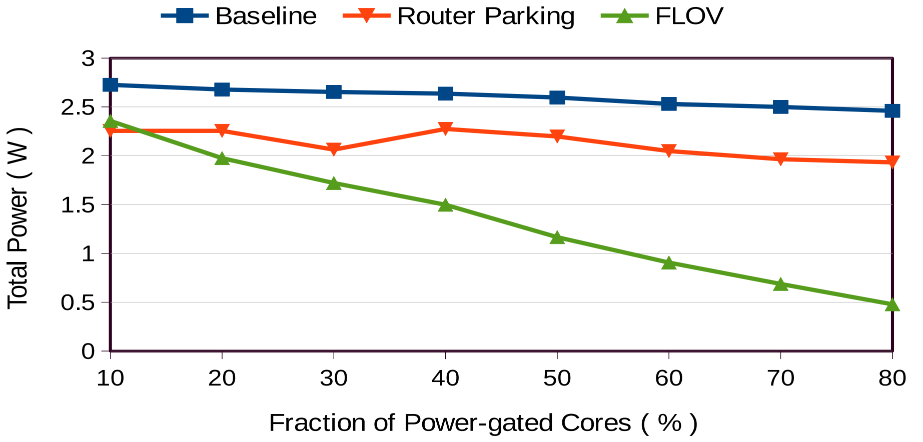


Figure 6.12: Average NoC Total Power Comparison for Injection Rates of 0.08 flits / node / cycle under Tornado Traffic.

Table 6.1: Simulation Testbed parameters

Network Topology	8×8 Mesh
Input Buffer Depth	6 flits
Router	4-stage (4 cycles) router
Virtual Channel	3 regular VC and 1 escape VC
Memory Controllers	8 in the rightmost column
Technology	32 nm
Frequency	2GHz
Link Length	1mm
Link Latency	1 cycle
Power-gating overhead	2.3pJ
Baseline Routing	X-Y Routing

6.3 Performance

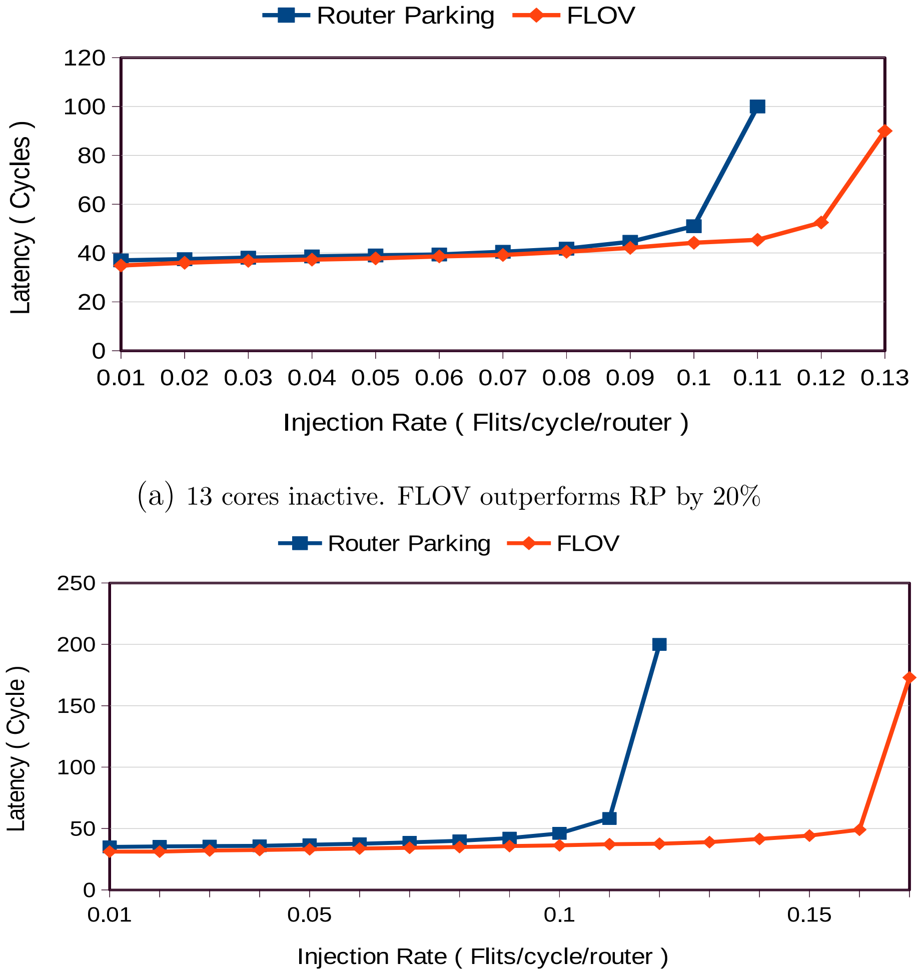
Figure 6.1, 6.4 and Figure 6.7 6.10 show the NoC latency comparison of FLOV with RP and the baseline interconnect. As the number of inactive cores increases FLOV power-gates all the routers attached to the inactive cores, whereas Router Parking makes a dynamic decision based on maintaining network connectivity. We can observe that even though FLOV relatively power-gates more routers, the latency is better compared to Router Parking for all the injection rates. This is because in Router-parking a packet will always need to reroute through the powered-on routers and links connecting them thereby increasing the path length. In FLOV we take advantage of all the links since the packet can be sent through a minimal path although some of the intermediate routers are power-gated using fly-over links. When the deliverability cannot be ensured by the routing algorithm, escape channel routing is used which may increase the hop count. However the fly-over links do not incur the baseline routers per-hop latency (4-stage), thereby reducing the average per packet latency even if the escape routing is used.

Another observation is that as the injection rate increases the latency reduction benefits of FLOV compared to RP also decreases. This is especially true when the fraction of inactive cores is above 50%. This is because more packets need to use the escape channel routing and hence the escape VCs in FLOV are congested at high injection rates when the number of power-gated routers is high. Even in such cases the FLOV latency outperforms Router Parking.

One interesting observation is, under Tornado traffic with an injection rate of 0.08 flits/cycle/router in Figure 6.1(a) we can see that Router Parking exhibits lower latency than FLOV when 40% of cores are inactive. This is due to the fact that Router Parking dynamically turns on additional routers that are attached to inactive cores to negate the impact of higher traffic in the network. This can also be observed from Figure 6.1(c), where the Total power consumption is increased when the fraction of inactive cores goes from 30 to 40%. Hence as more number of routers are turned on Router Parking trades off static power-savings for latency benefits.

Figures 6.13(a) and (b) show the throughput analysis of FLOV versus Router Parking for two scenarios, when 13 and 29 cores are inactive. FLOV has 20 % and 45 % better throughput than Router Parking, respectively. We can observe that Router Parking saturates the network faster compared to FLOV and the difference is magnified when the number of inactive cores is increased. As the number of inactive cores increases Router Parking has to reroute the aggregated traffic through the decreasing number of powered-on routers, which creates traffic hot spots. FLOV, on the other hand, utilizes the fly-over links to distribute the traffic flow across the network thereby reducing the hot spot creation probability.

Figure 6.15 and Figure 6.16 are the results from Uniform Random Multicast work-



(a) 13 cores inactive. FLOV outperforms RP by 20%

(b) 29 cores inactive. FLOV outperforms RP by 45%

Figure 6.13: Throughput Analysis for FLOV vs RP under Uniform Random Traffic.

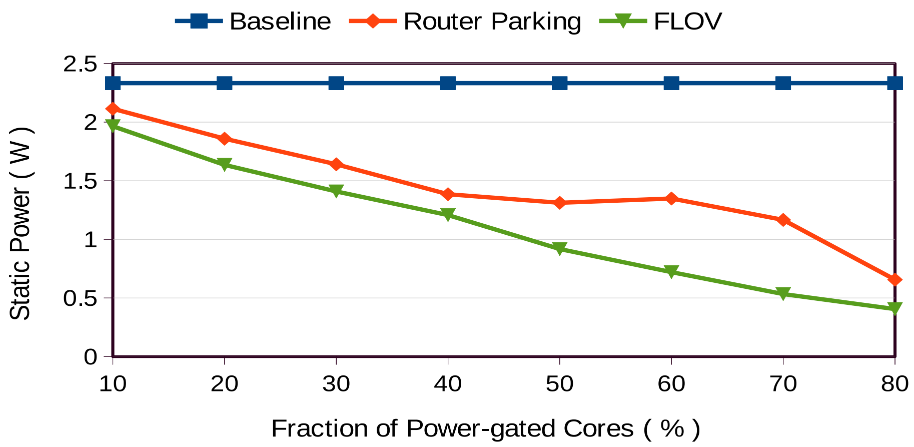


Figure 6.14: Static Power Comparison of FLOV vs RP vs Baseline with No Router Power-Gating.

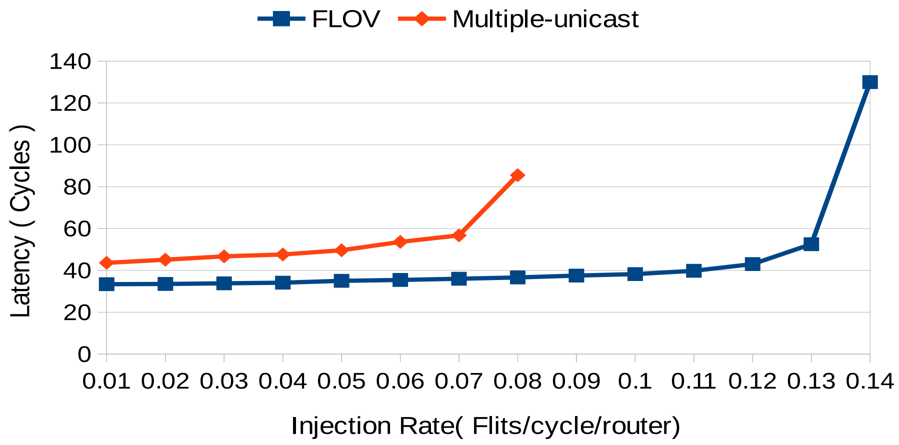


Figure 6.15: Latency and throughput Comparison of FLOV vs Multiple-unicast for multicast workload with 6 routers power-gated.

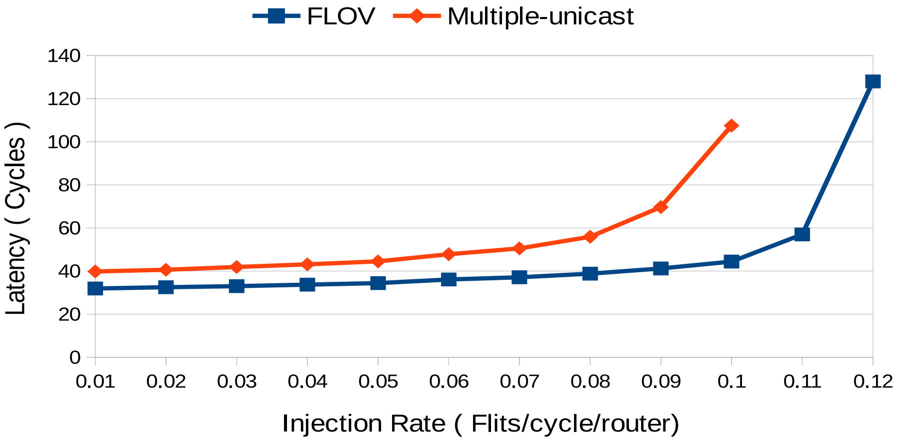


Figure 6.16: Latency and throughput Comparison of FLOV vs Multiple-unicast for multicast workload with 29 routers power-gated.

load. Since Router Parking itself does not support Multicast directly, we compare our routing algorithm against the baseline scheme called multiple-unicast. Multiple-unicast is the simpliest scheme for multicast. When a packet with multiple destinations is to be injected into the network, it should be replicated in the source node and therefore many unicast packets are generated from the replication. The drawback of this scheme is that the replication takes place as early as possible, therefore more bandwidth resources are consumed by these early replicated packets.

In Figure 6.15, there are 6 (10%) cores as well as routers powered-gated In Figure 6.16. there are 29 (45%) cores as well as router power-gated. In both cases, the portion of the multicast packets in the traffic is fixed at 15% and the average number of destinations in the multicast packet is 6. We can observe from these two result that our routing algorithm can acheive lower latency and better throughput than the baseline routing. This is because our routing algorithm make decisions based on the section of the destinations. This scheme delay the replication process and therefore recduce the traffic in the network.

6.4 Power Consumption

Figures 6.2, 6.5, 6.8, and 6.11 show the dynamic and total power consumption of FLOV as compared to Router Parking for multiple injection rates correspondingly. Figure 6.14 shows the static power consumption comparison, which is injection rate and workload independent for FLOV, since all the routers attached to inactive cores are power-gated. Router Parking dynamically decides whether to conservatively or aggressively power-gate routers, using a power-savings versus latency tradeoff prediction based on the interconnect workload. To reduce redundancy of using the same results of FLOV for multiple injection rates we compare against an aggressive RP power-gating scheme, which will make the RP power results also workload independent. This allows for a fair comparison with RP and lets us depict the static power evaluation in Figure 6.14.

We can observe from Figures 6.2 and 6.8 that for multiple injection rates the dynamic power consumption of FLOV is lower than RP, since in Router Parking, every hop in the rerouted packet traversal requires the total router pipeline execution, whereas in FLOV the intermediate power-gated routers use the fly-over links which consume significantly lower power.

In Figure 6.14 the static power consumption of FLOV is significantly lower than the Router Parking scheme and the disparity increases as the number of inactive cores increases. This is mainly due to the fact that FLOV power-gates more routers than Router Parking. Especially when the number of inactive cores is large, the fraction of routers that can be power-gated by Router Parking will saturate.

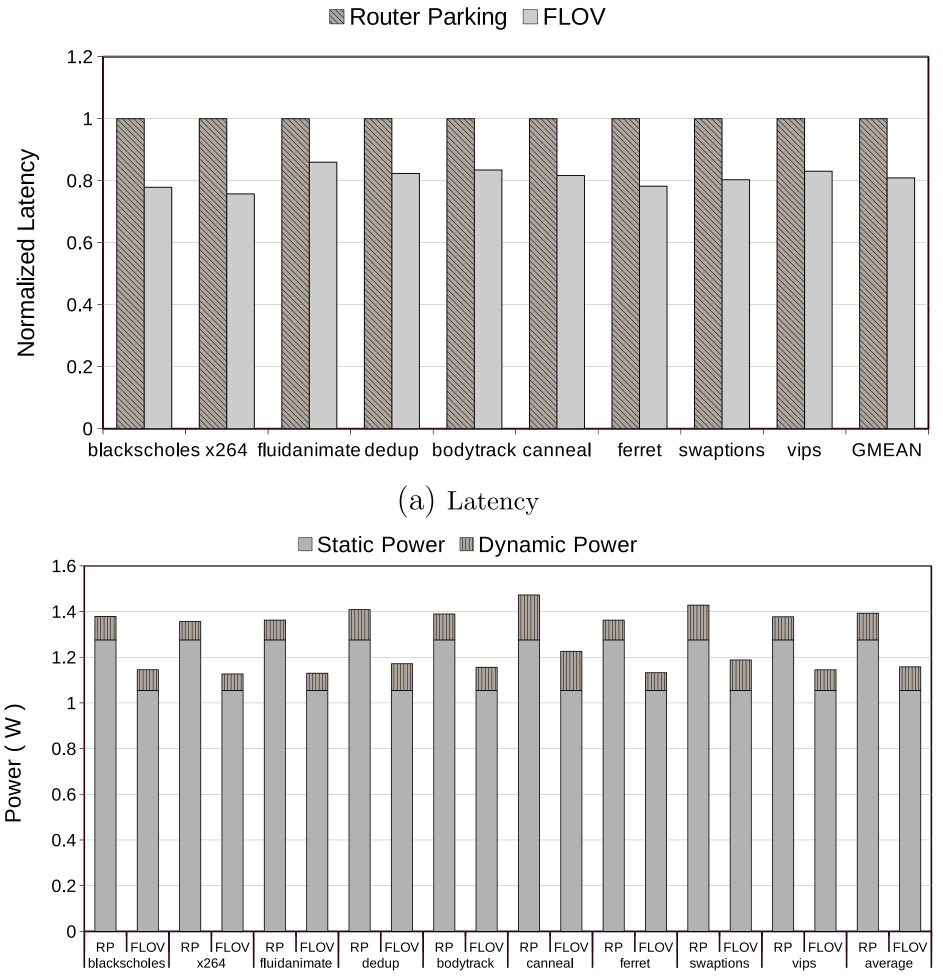
Figures 6.3, 6.6, and 6.9 , 6.12, show the Total power consumption of FLOV versus Router Parking. We can clearly observe that FLOV unanimously has lower power consumption, since FLOV dynamic and static power consumptions are lower than Router Parking as described above.

6.5 Real Workload Evaluation

To examine the behavior of FLOV under real workloads, we run benchmark traces generated by NETRACE [13]. The NETRACE library provides network traces from the PARSEC benchmark suite [3], and the packet dependency is carefully considered in their library. Nine benchmarks from PARSEC are chosen and all experiments are conducted on a fixed interconnect scenario. Our scenario assumes that 29 of 64 cores (45%) are inactive and the distribution is randomly generated and fixed for all the experiments.

We observe from Figure 6.17(a) that the latency of FLOV is lower than Router Parking by 19.2% on average across all the benchmarks. This is in accordance with the latency results from the synthetic workloads.

Figure 6.17(b) shows the static, dynamic and total power consumption comparisons between FLOV and Router Parking. Our scheme reduces the static power consumption by 17.3% on average across the nine benchmarks and the dynamic power consumption by 11.7%.



(a) Latency

(b) Breakdown of Power Consumption

Figure 6.17: Average Interconnect Latency Normalized to RP (a) and Total Power Consumption Breakdown into Static and Dynamic power (b) for Parsec Benchmarks. (GMEAN in (a) is the geometric mean across all the benchmarks.)

7. CONCLUSIONS

We proposed Fly-Over (FLOV), a light-weight distributed router power-gating mechanism. We modified the router microarchitecture to enable the fly-over links and explained in detail the dynamic routing algorithm. The proposed manifestations work in conjunction to ensure network functionality without global network information based decision making. FLOV achieves better NoC power savings due to power-gating more routers and also avoiding aggregated traffic rerouting in the network unlike Router Parking. Our evaluations using synthetic and real workload shows that the average interconnect latency is also reduced compared to Router Parking. We observe that FLOV further reduces the interconnect latency by 19.2% and total power consumption by 16.9% across nine PARSEC 2.1 benchmarks compared with Router Parking. We also observe that our FLOV can also provide better latency as well as throuhput for multicast traffic. We got 70% and 20% throughput improvement respectively when 10% and 45% of the routers are power-gated.

REFERENCES

[1] Murali Annavaram. A case for guarded power gating for multi-core processors. In *High Performance Computer Architecture (HPCA), 2011 IEEE 17th International Symposium on*, pages 291–300. IEEE, 2011.

[2] Arnab Banerjee, Robert Mullins, and Simon Moore. A power and energy exploration of network-on-chip architectures. In *Proceedings of the First International Symposium on Networks-on-Chip*, pages 163–172. IEEE Computer Society, 2007.

[3] Christian Bienia, Sanjeev Kumar, Jaswinder Pal Singh, and Kai Li. The parsec benchmark suite: Characterization and architectural implications. In *Proceedings of the 17th international conference on Parallel architectures and compilation techniques*, pages 72–81. ACM, 2008.

[4] Lizhong Chen and Timothy M Pinkston. Nord: Node-router decoupling for effective power-gating of on-chip routers. In *Proceedings of the 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 270–281. IEEE Computer Society, 2012.

[5] Lizhong Chen, Lihang Zhao, Ruisheng Wang, and Timothy M Pinkston. Mp3: Minimizing performance penalty for power-gating of clos network-on-chip. In *High Performance Computer Architecture (HPCA), 2014 IEEE 20th International Symposium on*, pages 296–307. IEEE, 2014.

[6] Lizhong Chen, Di Zhu, Massoud Pedram, and Timothy M Pinkston. Power punch: Towards non-blocking power-gating of noc routers. In *High Performance*

Computer Architecture (HPCA), 2015 IEEE 21st International Symposium on, pages 1–12. IEEE, 2015.

[7] Xuning Chen and Li-Shiuan Peh. Leakage power modeling and optimization in interconnection networks. In *Proceedings of the 2003 international symposium on Low power electronics and design*, pages 90–95. ACM, 2003.

[8] SPEC CPU2006. Standard performance evaluation corporation, 2006.

[9] William J Dally. Virtual-channel flow control. *Parallel and Distributed Systems, IEEE Transactions on*, 3(2):194–205, 1992.

[10] William J Dally and Charles L Seitz. Deadlock-free message routing in multiprocessor interconnection networks. *Computers, IEEE Transactions on*, 100(5):547–553, 1987.

[11] Robert H Dennard, Fritz H Gaensslen, V Leo Rideout, Ernest Bassous, and Andre R LeBlanc. Design of ion-implanted mosfet's with very small physical dimensions. *Solid-State Circuits, IEEE Journal of*, 9(5):256–268, 1974.

[12] José Duato. A new theory of deadlock-free adaptive routing in wormhole networks. *Parallel and Distributed Systems, IEEE Transactions on*, 4(12):1320–1331, 1993.

[13] Joel Hestness and Stephen W Keckler. Netrace: Dependency-tracking traces for efficient network-on-chip experimentation. *The University of Texas at Austin, Dept. of Computer Science, Tech. Rep*, 2011.

[14] Yatin Hoskote, Sriram Vangal, Arvind Singh, Nitin Borkar, and Shekhar Borkar. A 5-ghz mesh interconnect for a teraflops processor. *IEEE Micro*, (5):51–61, 2007.

[15] Jason Howard, Saurabh Dighe, Sriram R Vangal, Gregory Ruhl, Nitin Borkar, Shailendra Jain, Vasantha Erraguntla, Michael Konow, Michael Riepen, Matthias Gries, et al. A 48-core ia-32 processor in 45 nm cmos using on-die message-passing and dvfs for performance and power scaling. *Solid-State Circuits, IEEE Journal of*, 46(1):173–183, 2011.

[16] Natalie Enright Jerger and Li-Shiuan Peh. On-chip networks. *Synthesis Lectures on Computer Architecture*, 4(1):1–141, 2009.

[17] Natalie Enright Jerger, Li-Shiuan Peh, and Mikko Lipasti. Virtual circuit tree multicasting: A case for on-chip hardware multicast support. In *Computer Architecture, 2008. ISCA'08. 35th International Symposium on*, pages 229–240. IEEE, 2008.

[18] Eun Jung Kim, Ki Hwan Yum, Greg M Link, Narayanan Vijaykrishnan, M Kandemir, Mary Jane Irwin, M Yousif, and Chita R Das. Energy optimization techniques in cluster interconnects. In *Proceedings of the 2003 international symposium on Low power electronics and design*, pages 459–464. ACM, 2003.

[19] Rakesh Kumar, Alejandro Martínez, and Antonio González. Dynamic selective devectorization for efficient power gating of simd units in a hw/sw co-designed environment. In *Computer Architecture and High Performance Computing (SBAC-PAD), 2013 25th International Symposium on*, pages 81–88. IEEE, 2013.

[20] Jungseob Lee and Nam Sung Kim. Optimizing throughput of power-and thermal-constrained multicore processors using dvfs and per-core power-gating. In *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE*, pages 47–50. IEEE, 2009.

[21] Jacob Leverich, Matteo Monchiero, Vanish Talwar, Parthasarathy Ranganathan, and Christos Kozyrakis. Power management of datacenter workloads

using per-core power gating. *Computer Architecture Letters*, 8(2):48–51, 2009.

[22] Hiroki Matsutani, Michihiro Koibuchi, Daisuke Ikebuchi, Kimiyoshi Usami, Hiroshi Nakamura, and Hideharu Amano. Ultra fine-grained run-time power gating of on-chip routers for cmps. In *Networks-on-Chip (NOCS), 2010 Fourth ACM/IEEE International Symposium on*, pages 61–68. IEEE, 2010.

[23] Gordon E Moore et al. Cramming more components onto integrated circuits. *Proceedings of the IEEE*, 86(1):82–85, 1998.

[24] Ritesh Parikh, Reetuparna Das, and Valeria Bertacco. Power-aware nocs through routing and topology reconfiguration. In *Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE*, pages 1–6. IEEE, 2014.

[25] Reetuparna Das Ritesh Parikh and Valeria Bertacco. Power-aware nocs through routing and topology reconfiguration. In *Design Automation Conference 2014, DAC '14, San Francisco, CA, USA, June 1-5, 2014*, pages 1–6. IEEE, 2014.

[26] Ahmad Samih, Ren Wang, Anil Krishna, Christian Maciocco, Charlie Tai, and Yan Solihin. Energy-efficient interconnect via router parking. In *High Performance Computer Architecture (HPCA2013), 2013 IEEE 19th International Symposium on*, pages 508–519. IEEE, 2013.

[27] Vassos Soteriou and Li-Shiuan Peh. Design-space exploration of power-aware on/off interconnection networks. In *Computer Design: VLSI in Computers and Processors, 2004. ICCD 2004. Proceedings. IEEE International Conference on*, pages 510–517. IEEE, 2004.

[28] Chen Sun, CHO Chen, George Kurian, Lan Wei, Jason Miller, Anant Agarwal, Li-Shiuan Peh, and Vladimir Stojanovic. Dsent-a tool connecting emerging photonics with electronics for opto-electronic networks-on-chip modeling. In

Networks on Chip (NoCS), 2012 Sixth IEEE/ACM International Symposium on, pages 201–210. IEEE, 2012.

[29] Michael Bedford Taylor, Jason Kim, Jason Miller, David Wentzlaff, Fae Ghodrat, Ben Greenwald, Henry Hoffman, Paul Johnson, Jae-Wook Lee, Walter Lee, et al. The raw microprocessor: A computational fabric for software circuits and general-purpose programs. *Micro, IEEE*, 22(2):25–35, 2002.

[30] Lei Wang, Yuho Jin, Hyungjun Kim, and Eun Jung Kim. Recursive partitioning multicast: A bandwidth-efficient routing for networks-on-chip. In *Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip*, pages 64–73. IEEE Computer Society, 2009.