A Lightweight Fault-Tolerant Mechanism for Network-on-Chip

Michihiro Koibuchi[1], Hiroki Matsutani[2], Hideharu Amano[2], and Timothy Mark Pinkston[3]

[1]National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyoda-ku,
Tokyo, JAPAN 101-8430, koibuchi@nii.ac.jp
[2]Keio University, 3-14-1, Hiyoshi, Kohoku-ku, Yokohama, JAPAN 223-8522,
{matutani, hunga}@am.ics.keio.ac.jp
[3] University of Southern California, 3740, McClintok Ave., EEB 208 Los Angeles,
California 90089-2562, tpink@usc.edu

Abstract

Survival capability is becoming a crucial factor in designing multicore processors built with on-chip packet networks, or networks on chip (NoCs). In this paper, we propose a lightweight fault-tolerant mechanism for NoCs based on default backup paths (DBPs) *designed to maintain, in the presence of failures, network connectivity of both non-faulty routers as well as healthy processor cores which may be connected to faulty routers. The mechanism provides default paths as backup between certain router ports which serve as alternative datapaths to circumvent failed components within a faulty router. Along with a minimal subset of normal network channels, the set of default backup paths internal to faulty routers form—in the worst case—a unidirectional ring topology that provides network-wide connectivity to all processor cores. Routing using the DBP mechanism is proved to be deadlock-free with only two virtual channels even for fault scenarios in which regular networks degrade to irregular (arbitrary) topologies. Evaluation results show that, for a 2-D mesh wormhole NoC, only 12.6% additional hardware resources are needed to implement the proposed DBP mechanism in order to provide graceful performance degradation without chip-wide failure as the number of faults increases to the maximum needed to form ring.*

1 Introduction

Networks-on-chip (NoCs) have been shown to efficiently interconnect many functional units, tiles, and cores—collectively referred to in this paper as processing elements or PEs—of a chip [19, 3, 18]. As device technology continues to scale into the nanometer regime, the number of PEs on a single chip will increase considerably, making the need for robustly designed NoCs even more pronounced. Along with this extreme device scaling comes an increased likelihood of failures, both transient (soft) and permanent (hard), to occur within the NoC. Soft faults cause data to be momentarily corrupted (e.g., bit errors), which can be corrected within the network through link-level and end-to-end protocol techniques [14]. Hard faults are caused by physical and permanent damage to resources that generate and/or transport data, and the probability of their occurrence depends on various design and technology parameters. Typically, the larger, more complicated and more stressed the structure, the more likely it is to be susceptible to faults.

A NoC consists of routers, links, and network interfaces attached to PEs in the system, each of which may possibly fail. Packets are transported through one or more of these network components, with a large number of PE source-destination pairs sharing each component. Each faulty network component thus affects the communication among a large number of PEs. A faulty network component may exclude healthy PEs from gaining access to the rest of the system (i.e., cause healthy PEs to be masked out, as is done typically) or, even worse, prevent the entire system from operating reliably.

Redundancy is often used to combat failures. To build robust NoCs using redundancy, different spare resources could be used for every component, but fully duplicating hardware in this manner is prohibitively expensive. Moreover, in contrast to the case for off-chip networks, components in on-chip networks cannot be fixed or replaced post deployment. Thus, keeping in mind that the simpler the redundant resource is, the less likely it is to suffer a fault, nominal redundancy should be built into the design of the NoC such that connectivity of all non-faulty network components as well as all healthy PEs attached to faulty components is maintained.

In this paper, we propose a lightweight fault-tolerant mechanism based on the notion of *default backup paths (DBPs)*. It uses nominal redundancy to maintain network

connectivity of non-faulty NoC routers and healthy on-chip PEs in the presence of hard failures occurring in the network. In achieving a lightweight reliable structure, the mechanism provides nominal default paths as backup between certain router ports which serve as alternative datapaths to circumvent failed components (i.e., input buffers, crossbar switch, etc.) within a faulty NoC router. Along with a minimal subset of normal network channels, the set of default backup paths internal to faulty routers form—in the worst case—a unidirectional ring topology that provides network-wide connectivity to all PEs. This lightweight fault-tolerant mechanism is premised on the notion that complicated redundancy techniques need not be used to support high reliability in NoCs. The proposed DBP mechanism is shown to provide graceful performance degradation without chip-wide failure as the number of faults increases to the maximum tolerable to form a ring.

The rest of this paper is organized as follows. In Section 2, we briefly discuss some related fault-tolerant techniques for NoCs to provide additional background. In Sections 3 and 4, we describe the DBP mechanism for reliability-constrained NoCs and discuss how routing with the DBP mechanism can be done in a deadlock-free manner. In Section 5, we evaluate the performance of the DBP mechanism and, finally, conclude the paper in Section 6.

2. Related Work

NoCs are composed of router switches and point-to-point links; thus their hard failures can be individually detected by existing techniques. Unlike off-chip interconnects using bit-serial links, NoC channels have wider bit-widths. It is possible to employ channel reconfiguration techniques to tolerate link faults with graceful performance degradation by decoupling singular wire faults from the other wires composing the channel [21]. Analogous to this, when channel utilization is low, powering down some wires through similar decoupling techniques can provide energy savings within the interconnect [17][1]. Moreover, techniques and supporting theory have been proposed recently for static and dynamic reconfiguration of the network routing algorithm to tolerate faults that may occur [4][11]. These approaches are all orthogonal to the DBP mechanism as it focuses on fault-tolerance support for routers as opposed to channels or the network routing algorithm.

Router architectures have also been proposed that include fault-tolerant mechanisms for bypassing hard faults in some units along the router internal datapath such as the routing computation, input buffer, and switch arbiter units [9]. Similarly, a mechanism called BLAM provides central bypass buffers so that a packet passes the previous misrouted packet on the same input port [20]. Another technique using additional datapaths is called "preferred paths" in order to drastically reduce packet latency at routers [13]. These techniques, while useful for the purposes proposed, do not provide support for bypassing the entire router internal datapath nor network-wide support for ensuring connectivity and deadlock-free routing on those resources. The lightweight fault-tolerant mechanism proposed here does provide this support.

The redundancy techniques for bypassing faults within NoC hardware components can be broadly grouped into three approaches or combinations of approaches: resource sparing, fault-tolerant routing, and network reconfiguration. Their designs often suppose simple fault models that use a network graph in which each vertex is a router and each arc is a bidirectional link. Moreover, faults are simply classified into those of link and router, and router faults are sometimes regarded as all link faults around the router. The DBP mechanism derives from a combination of the above approaches, and its fault model makes use of a network graph in which unidirectional links and DBP resources are explicitly represented.

3. The Default Backup Path Mechanism

In this section, we describe the proposed DBP mechanism, which is mainly designed to (1) maintain the network connectivity of healthy PEs associated with a faulty router and (2) avoid disconnecting the network due to faults occurring along the internal datapath of routers. The basic idea is to bypass faulty datapaths within failed routers through the use of backup default paths that, themselves, have low susceptibility to failure due to their simplicity of design. The set of router input-output port pairs connected by the default backup paths at routers across the network in combination with normal network channels corresponding to those ports must form a connected network that includes all routers and PEs. With this, the DBP mechanism plays the role of a *lifeline* for the NoC, allowing performance to degrade gracefully as the number of router faults increase to the maximum possible (i.e., all routers becoming faulty) without having the network to fail.

NoCs are switched point-to-point networks characterized by a combination of topology, routing, switching technique, flow control, and router microarchitecture. The DBP mechanism can be applied to NoCs composed of direct or indirect topologies, adaptive or deterministic routing algorithms, any switching and flow control technique (including wormhole routing), and most router architectures. To explain the DBP mechanism, a 4×4 2-D mesh using conventional virtual channel routers as detailed in [21] is used as an example of a typical NoC, shown in Figure 1. In the router, the pipeline to transfer packets consists of routing computation (RC), virtual channel allocation (VC) for output ports, switch allocation (SW) for allocating the time-slot on the crossbar, and switch transfer (ST) for flight of flits.

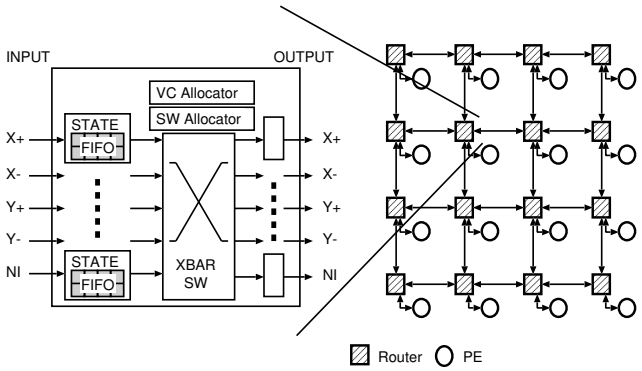


Figure 1. A conventional NoC

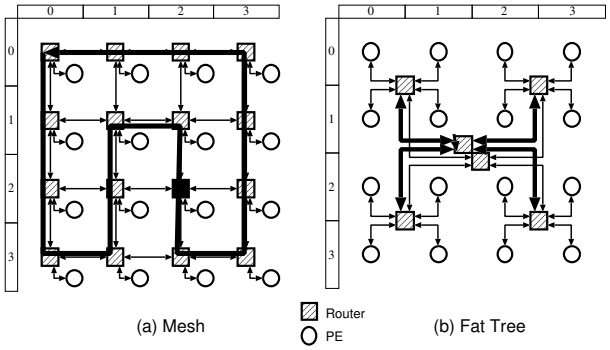


Figure 2. Embedded unidirectional cycles

3.1 Building Default Backup Paths

The DBP mechanism can be applied to NoCs by finding an embedded unidirectional cycle that includes all routers that have a local PE in the topology of the network and adding a bypass datapath along the unidirectional cycle.

3.1.1 Finding an Embedded Unidirectional Cycle

A unidirectional cycle that includes all routers that have a local PE is constructed in the NoC topology. Since either a Hamiltonian cycle (Figure 2.a) or a ring along a spanning tree (Figure 2.b) can be the unidirectional cycle, an arbitrary topology can always be embedded [15]. The input and output router ports in the cycle are referred as "DBP ports" in this paper.

3.1.2 Adding Default Paths

We add default paths that bypass the original datapath (i.e., crossbar) within a router such that packets can be transferred along the unidirectional cycle without traversing the router input buffers, crossbar, routing-computation unit, VC/SW allocators, etc., as follows.

1. For each router, in addition to the datapath through the crossbar, an input DBP port is directly connected to a PE local (i.e., network interface or NI) output port by hard wires with a multiplexer, as shown in Figure 3. The newly introduced default backup path to the NI enables packets to bypass all router modules as it sinks them at the local PE. It may or may not require a FIFO buffer as a repeater, depending on the implementation (i.e., critical path margins), as shown in Figure 3.

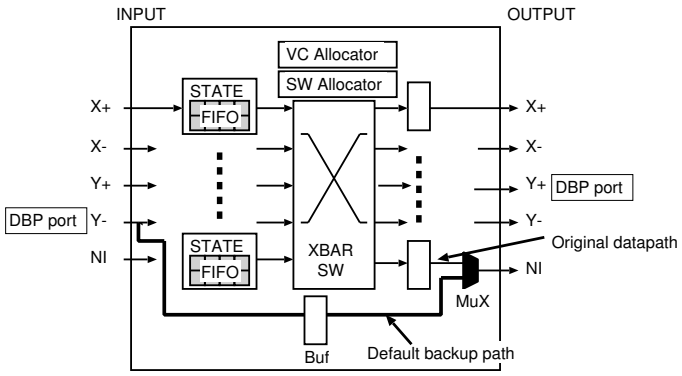


Figure 3. Default backup path to local PE (black router in Figure 2)

2. For each router, as in datapaths through the router crossbar, a PE local (i.e., NI) input port is directly connected to the output DBP port (which connects to the neighboring router by the channel link) by hard wires with a multiplexer, as shown in Figure 4. This newly introduced default backup path from the NI enables packets to bypass all router modules as it sources them from the local PE.

3. A function must be added to NIs that forwards packets that happen to sink unintentionally at an NI through the default backup path of the associated router due to a fault(s). This functionality is similar to that used in the In-Transit Buffer technique [6].

The default backup path and the original datapath via the crossbar can be selected by the channel multiplexer, as shown in Figures 3 and 4. For the sake of control simplicity, as long as the router is healthy, its channel multiplexer always selects the original datapath.

When there are no local PEs at a router, a pair of input and output DBP ports for neighboring routers is directly connected along the unidirectional cycle, as shown in Figure 5. This is the case for indirect networks, such as non-leaf switches in a fat tree. Conversely, when there are multiple PEs at a router, an input and all local output ports are directly or indirectly connected through no crossbars. All local input ports are also connected to an output port through no crossbars.

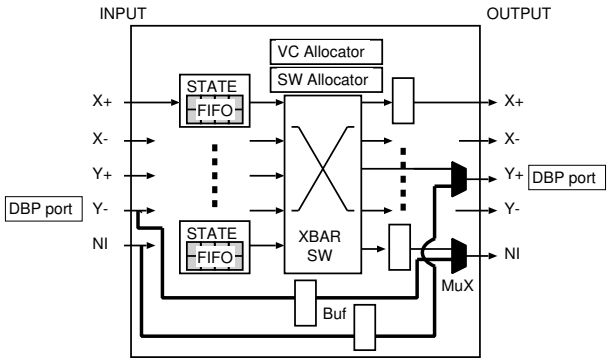


Figure 4. Default backup paths to/from local PE (black router in Figure 2)

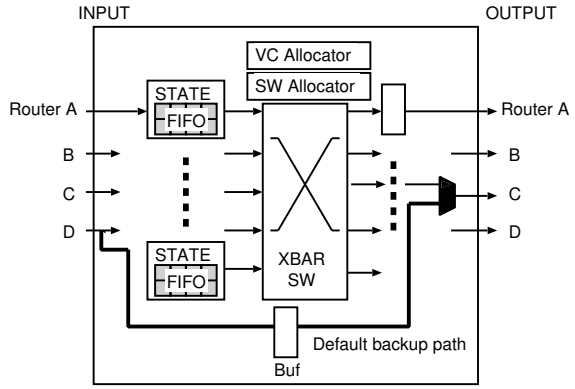


Figure 5. Default backup path between routers on indirect networks

3.2 Behavior of the DBP Mechanism

As long as all routers are healthy, none of the default backup paths are used, as described in the previous subsection. Only when (1) a local PE cannot send or receive packets or (2) a packet cannot be properly transferred along the datapath via the crossbar to/from the DBP port (caused by a hard fault on a router) is the failed datapath replaced by the corresponding default backup path. Only the default backup path at a faulty router is enabled simply by switching the channel multiplexer. Then a healthy PE associated with the faulty router continues to communicate with the other PEs via the default backup path.

When all default backup paths are enabled—as when all router crossbars fail—the network topology takes the form of a unidirectional ring, such as the embedded cycle in Figure 2. Then, a packet is transferred via both intermediate routers and their local PEs. In such a case, the network interface at a PE receives packets destined for the other PEs and forwards them on to the intended PEs from the associated router (the detailed routing mechanism is discussed in the next subsection). Since none of the NIs have to consume an entire packet immediately, a wormhole switching technique can also be used.

The enabled default backup path to local PEs disables the packet transfer from the other port to the local PE, and the PE is able to receive packets only from a single neighboring router via the default backup path. In contrast, the enabled default backup path from the local PE disables the packet transfer from the local PE to ports except the output DBP port. Although it can decrease the throughput around the local PE associated with the faulty router, the system still works correctly using all the healthy PEs. The default backup path can transfer packets with shorter latency and lower energy than router pipelines on the original datapath via the crossbar because the default backup path consists only of hardwires or repeater buffers without routing. These trade-offs in performance are evaluated in Section 5.

3.3 DBP Mechanism for Other Component Failures

Although the DBP mechanism is resilient to router failures, other network components such as the default backup path itself may also fail, though the likelihood is much lower. Here, we present the DBP mechanism extended to tolerate the faults of the network interface, the link, and the default backup path itself.

3.3.1 Bypassing NI Faults

When the network interface at PEs are permanently broken, forwarding through the corresponding default backup path to the intended PE whose NI may be healthy is prevented. To tolerate NI faults so as to allow packets to reach their intended destinations, we propose to add another default backup path between the input and output DBP ports of a router, as shown in Figure 6. This provides an additional default backup path at the router to perform the function similar to that used for forwarding through the NI at the router, as in Figure 4. Only when both the router datapath and the network interface fail would the newly introduced default backup path be enabled. Its operation bypasses both the faulty router datapath and the faulty NI, and it operates similar to the DBP for indirect networks shown in Figure 5.

3.3.2 Bypassing Damage of Link Wires

When the channels connected to DBP ports are faulty, the default backup paths are of no use. To tolerate the damage, the DBP mechanism can be extended by employing multiple DBP cycles that compose different topologies. When the link wires fail on a channel along one DBP unidirectional cycle, another backup DBP cycle that consists of healthy channels can be invoked. To support multiple unidirectional DBP cycles, a DBP router must be equipped with more than one DBP port to/from neighboring routers. To

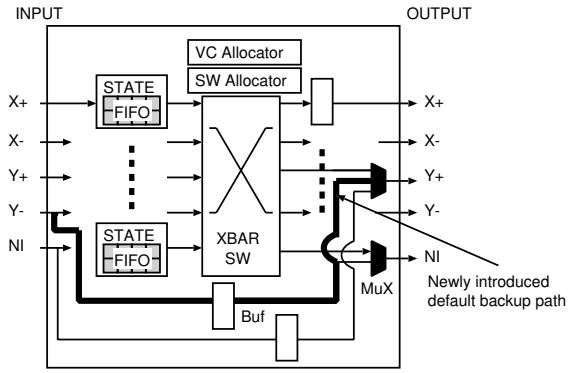


Figure 6. Default backup path bypassing both router modules, and network interface

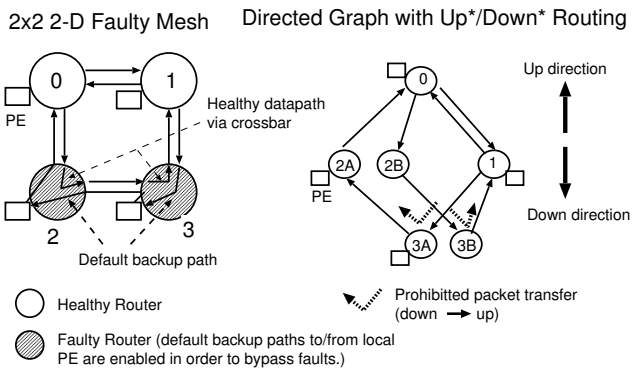


Figure 7. Disconnecting DBP network by up*/down* routing

maximize their fault tolerance, a large number of ports are set as DBP ports (all ports are marked as DBP), even though this increases hardware complexity.

3.3.3 Bypassing Faults of Default Backup Path Itself

Because of the simplicity of the DBP mechanism, the probability that it will experience a hard failure is low, but not *zero*. To tolerate this hard failure, multiple unidirectional rings and default backup paths bypassing both the local PE and router modules introduced in this section can be applied. Another method duplicates some of the hardware. It is reasonable to duplicate to some extent the default backup paths, because the nominal hardware needed is small and much simpler compared to that of the router datapath.

4 Deadlock-free Routing in DBP Networks

Although on-chip topologies, such as meshes, are usually regular, faults make the topologies irregular and hard for the network to establish paths that are free of routing deadlocks. Moreover, the DBP mechanism introduces a new routing problem in which ports can be partially disconnected by enabling the unidirectional default backup path on a router. In this section, we present ways in which the deadlock problem in DBP networks can be resolved based on former well-known theory and techniques.

4.1 Network Graph

To design deadlock-free paths, interconnection networks are usually modeled using a connected directed graph, $I = G(N, C)$. Vertices of graph N are the set of routers, whereas arcs of graph C are the set of channels[5].

Since the default backup path disables the datapath between the corresponding port and crossbar, the ports occupied by the default backup path and the remaining ports are physically disconnected in the same router. To make a directed graph that corresponds to the DBP mechanism, the router that enables the default backup path is distinguished as two sub-routers: the set of ports corresponding to the enabled default backup paths (i.e., set "A") and the set of remaining ports (i.e., set "B"), as shown in Figure 7 for the directed graph with Up*/Down* routing.

4.2 Building Routing Paths

Using the network model introduced in the previous subsection, we present a deadlock-free routing design. Since topology-agnostic routing algorithms, such as Up*/Down* routing that transfers packets along tree-based paths [16], provide a deadlock-free path set on arbitrary topologies, this path set seems to fit with DBP networks. However, they cannot be directly applied to the NoCs using the DBP mechanism. In the case of Up*/Down* routing, as shown in Figure 7, there are no legal paths from PE 0 and 1 to PE 2 as both require "down" to "up" transitions. This is because most topology-agnostic routing algorithms assume that each link consists of bidirectional channels, while the default backup path has only unidirectional paths. Consequently, a new routing strategy is needed for DBP networks.

We present a deadlock-free routing design using v virtual channels ($2 \leq v$) for the DBP mechanism on arbitrary topologies, where "virtual channel network" stands for a logical network using a single virtual channel number.

1. Routing restrictions are imposed for all virtual channels, in order to make the deadlock-free path set that allows packet transfer along the unidirectional ring cycle.

    - Routing restrictions within each virtual channel network are decided to satisfy deadlock freedom as long as every packet is routed inside the virtual channel network. It could be done by applying

an existing deadlock-free routing algorithm for parallel computers [5].

- To prevent deadlocks across virtual channel networks, packet transfer to a lower virtual channel network is prohibited.

As shown in Figure 8, in the case of a 2-D mesh, for example, the West-First Turn Model [7] is used as a deadlock-free escape virtual channel network. Its routing rule is simple: a packet is first transferred to the west direction zero or more hops, then adaptively to the destination without any turns to the westward direction [7]. Allowed paths include a path set of dimension-order routing used in various current NoCs. If a packet must take a forbidden turn, it must change to a virtual channel network with an incremented virtual channel number.

2. Find shortest paths between each source-destination pair under the condition with the above routing restrictions on the network as follows.
   (a) Search shortest paths when virtual channel $i = 0$ is used at the source.
   (b) Search the paths that have the same length as the paths in Step 3(a), when the virtual channel $i \leftarrow (i + 1)$ is used at the source.
   (c) Repeat Step 3(b) until no legal paths are found, or until the search which uses virtual channel $(v-1)$ at the source is complete.
   (d) When the target routing is deterministic, select a single path using a path selection algorithm when multiple shortest paths are found between a source-destination pair.

In the case of adaptive routing, a path set obtained by Step 3(c) is used. Step 1 guarantees paths between all pairs of PEs that can be non-minimal. Since NoCs usually have regular topologies, such as a 2-D mesh, we can apply routing algorithms used in regular topologies for the DBP network, as shown in Figure 8. Two virtual channels are sufficient for deadlock-free routing on an amended unidirectional ring. As long as deadlock freedom is maintained, packets can use channels at both DBP ports and normal (non-DBP) ports. A packet can thus be sent via normal ports as a short cut, compared with non-minimal paths along the unidirectional ring in an arbitrary topology.

Theorem *The DBP routing mechanism using v virtual channels ($2 \leq v$) is deadlock-free.*

Proof *No knotted cyclic dependencies can form within each virtual-channel network ($0, 1, \ldots, v-1$) as packets must follow the routing restrictions imposed by a deadlock-free routing algorithm when transferred within each virtual channel network. No knotted cyclic dependencies can form across virtual channel networks as packets are passed between virtual channel networks in increasing order only. As there are no deadlocks within virtual channel networks nor across virtual channel networks, the DBP routing mechanism is deadlock-free.*

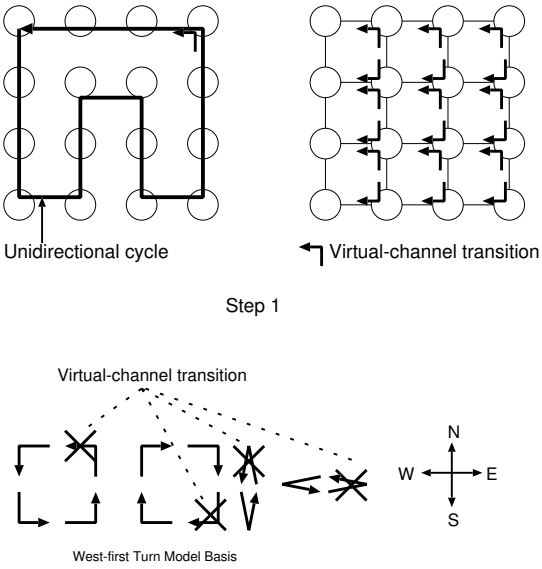


Figure 8. Routing restriction on a DBP network

This routing approach is similar to virtual channel transition-based routing [10]. In the case of no virtual channels, the In-Transit Buffer method [6] that temporally stores some packets at NIs so as to break channel cyclic dependency can also be used. In that method, the NI associated with the router at the dateline temporarily stores only the packets transferred along the unidirectional ring.

5 Evaluation of the DBP Mechanism

The DBP mechanism was evaluated through simulations in terms of the network logic area, energy consumption, and throughput.

5.1 Hardware Cost

The network logic area in a NoC is mainly composed of routers and network interfaces that connect processing elements to the network. Using the Synopsys Design Compiler, the generated NoC design with a ASPLA 90 nm standard cell library, and we completed its place and route. After the place and route, the behavior of the synthesized NoC design was confirmed assuming an operating frequency of 500MHz.

We implemented a conventional wormhole router whose architecture was fully three-stage pipelined. The flit-width

Table 1. Breakdown of the 5-port DBP router and NI area (Kilo gate)

	No VC	2-VC
Xbar & Arb	2.396 (10.76%)	10.056 (20.21%)
Channels	13.598 (61.10%)	27.457 (55.17%)
DBP	1.690 (7.59%)	3.376 (6.78%)
Misc	1.196 (5.37%)	2.371 (4.76%)
NI	3.376 (15.17%)	6.507 (13.07%)
Total	22.255 (100.00%)	49.766 (100.00%)

was set to 64 bits, and each pipeline stage had a buffer for storing one flit. Each input port has a FIFO buffer to store four flits. The routing decisions were stored in the header flit prior to packet injection (i.e., source routing); thus routing tables that require register files for storing routing paths were not needed in each router, resulting in a low cost router implementation. This router architecture is the same as the one used in [12].

The DBP router has a single DBP input/output port, and the other features are the same as those of the conventional router. Its router structure fits with a 2-D mesh, and its DBP unidirectional cycle is shown in Figure 8. We implemented the default backup path by inserting a FIFO buffer for storing two flits in each router. The network interface is designed to interface between a PE and its router with minimum hardware. We implemented a simple NI that employs a two-flit FIFO buffer for switching flits.

Figures 9(a) and 9(b) show the place and route results of 3-, 4-, 5-, and 6-port routers. In these figures, "Conv" stands for a conventional wormhole network with 2-D mesh for 16 PEs, while "DBP" stands for it with DBP mechanism using the unidirectional cycle in Figure 2. Using the values in Figures 9(a) and 9(b), Figure 9(c) shows the total router area of the 2-D mesh network, where "16-0VC" stands for 16-routers without virtual channels.

These results indicate that the DBP mechanism increases only by up to 12.6% the amount of hardware for the 2-D mesh. This is because that the DBP mechanism requires only at least two additional datapath, paths from/to local PE to/from the neighboring routers, which consists of a multiplexer and a 2-flit FIFO buffer for a DBP link bypassing the internal router modules. Table 1 shows the amount of NI area and itemizes the network logic area of the 5-port router. In Table 1,"DBP" stands for the hardware amount for the default backup path at a router. "Misc" includes inserted buffers for adjustment avoiding timing violation. The ratio of additional hardware for the DBP mechanism reduces as the number of ports per router increases. Note that every virtual channel requires a buffer, and the virtual-channel mechanism makes the structure of the arbiter and crossbar more complicated; thus increasing by 124% the router hardware, as reported in [12].

5.2 Throughput and Latency Performance

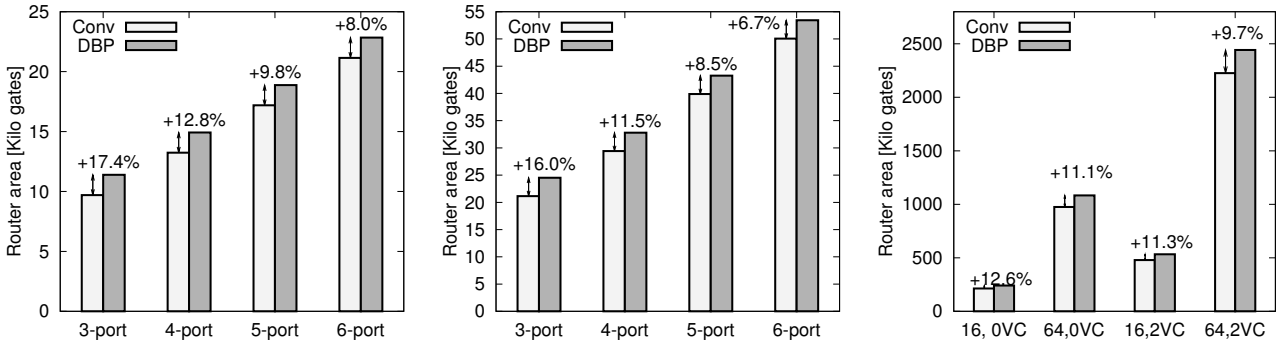
A flit-level simulator written in C++ was used for measuring the throughput. Every router has three, four, or five ports, and a single PE connected to every router. Wormhole switching was used as the switching technique of the router, and a FIFO buffer was included along the default backup path. The other features are the same as those used for estimating the amount of hardware. The PEs inject packets independently of each other, and we set the packet length at 16 flits, including one header flit. The destination of a packet is randomly determined, resulting in a uniform traffic pattern. Each host injects packets synchronized to the same interval, leading to bursty traffic like that in most scientific applications.

A 2-D mesh using one or two virtual channels is employed, and the West-First Turn Model [7] is also assumed for routing on each virtual network, as explained in Section 4. Since current NoCs usually support deterministic routing, we evaluated the DBP mechanism with deterministic routing. To find a deterministic path from a set of alternative path provided by an adaptive routing algorithm such as West-First Turn Model, a path selection algorithm is needed, as explained in Section 4. We use Sancho's algorithm [8], which is based on a static analysis of routing path, to determine a single path between each source-destination pair. In the case of no hardware faults with two virtual channels, both virtual channels are used in order to avoid Head-of-Line (HOL) blocking as follows: virtual channel 0 is used for paths to even-numbered destinations while virtual channel 1 is used for paths to odd-numbered destinations. In the case of faults, the DBP mechanism establishes paths using the procedure described in Section 4.

In the case of no virtual channels, the In-Transit Buffer method [6] is employed at a single router located at the dateline crossing the DBP unidirectional cycle (Figure 8). Theoretically, infinite buffers are needed at the NI (or PE) associated with the dateline In-Transit Buffer router [6]. Since infinite memory cannot be provided in NoCs, some packets cannot be temporarily stored when the buffer is filled with packets waiting for their reinjection. In this case, such overflowed packets are discarded at the intermediate node, and a NACK (negative acknowledgment) must be sent to its original source node so as to request a retransmission of the discarded packet. This method simply extends the end-to-end flow control mechanism in the higher-level protocol on NoCs.

Twenty different fault patterns are randomly generated for each fault scenario evaluated (i.e., number of faults). In this evaluation, a fault is considered to be a failure anywhere along the datapath between an input and output port pair of a router. As we are evaluating the DBP mechanism considering faults occurring only in the network, all PEs are assumed to be healthy.

Figures 10(a) and 10(b) show the throughputs of the



(a) Router comparison, no VCs

(b) Router comparison, two VCs

(c) Routers of 2-D mesh networks

Figure 9. Hardware amount of DBP and wormhole networks

DBP networks, averaged over 20 different fault patterns for each number of faults. The 0-fault case is simply the network without the need for the DBP mechanism. Figures 10(c), 10(d), 10(e) and 10(f) show the relation between the average latency, the accepted traffic, and the average packet hops of the DBP networks whose faulty patterns provide nearly average relative performance. Notice that the number of faults ranges from $zero$ to 48 on 16-router and 224-router networks, causing the topology to degenerate from 2-D mesh to a unidirectional ring. The DBP mechanism thus allows the network to gracefully degrade in the presence of a an increasing number of faults without disconnecting healthy PEs and routers from the network. Figures 10(c), 10(d), 10(e) and 10(f) show the average path hop count of the DBP networks. This is the main factor behind the performance degradation (i.e., lower throughput and increased latency).

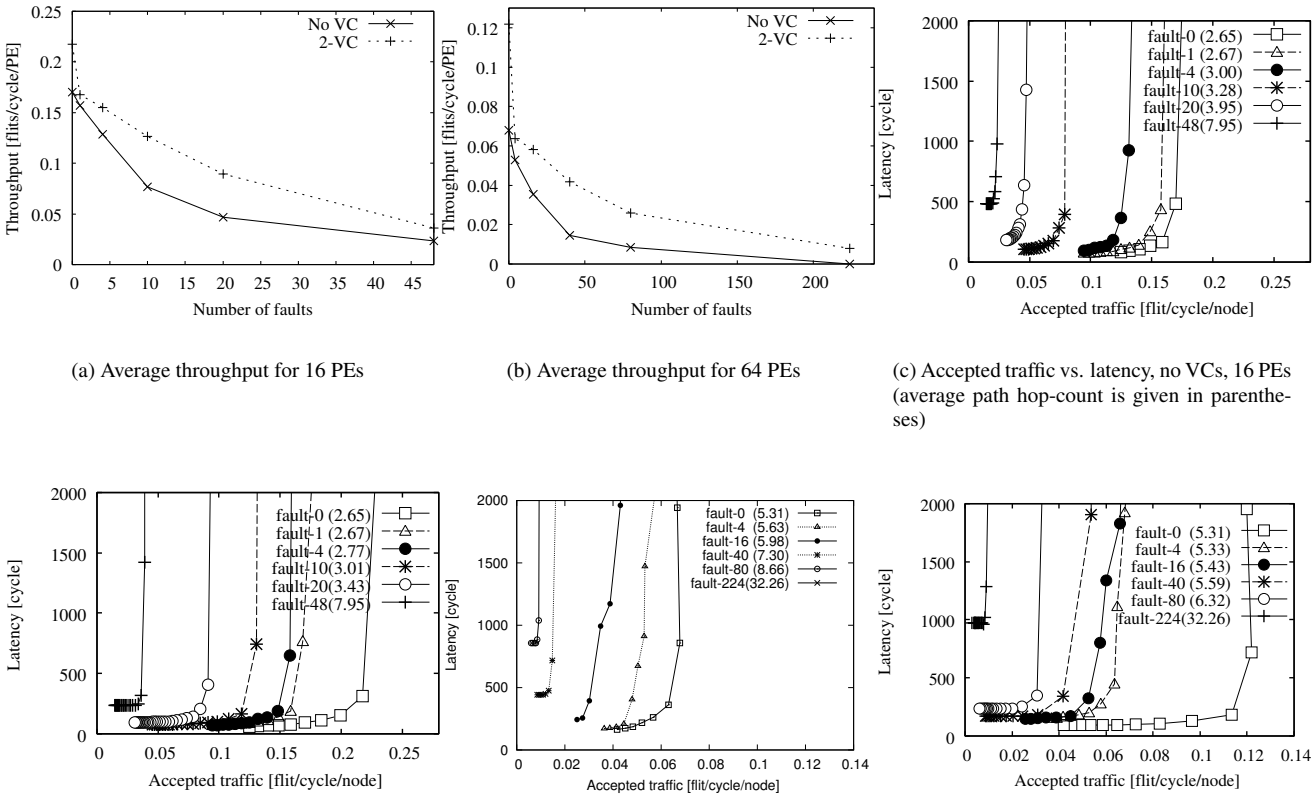
5.3 Energy Consumption

To estimate the power consumption of the router mentioned previously, the following steps were performed: (1) synthesis by Synopsys Design Compiler, (2) place and route with buffer insertions at CTS using Synopsys Astro, (3) post place-and-route simulation by Cadence Verilog-XL to obtain switching activity information of the router, and (4) power analysis based on the switching activity using Synopsys Power Compiler. A 90nm CMOS process with a core voltage of 1.0V was selected in this analysis. Clock gating and isolation were fully applied to the router to minimize its switching activity.

The router was simulated at 500MHz with various fixed workloads (throughputs), in the same manner as in [2]. A packet stream is defined as intermittent injections of 16-flit packets, which utilize about 30% of the maximum link bandwidth of a single router link. Each header flit contains a fixed destination address, while data flits contain a random payload. The number of packet streams injected into the router was changed so as to generate various workloads. We assume 64-bit networks with 16 and 64 PEs placed in 6 $\times$ 6mm^2 and 12 $\times$ 12mm^2 chips, respectively.

Figures 11(b) and 11(c) show the results of average energy consumption of packet transfers between PEs in the case of uniform traffic. We calculated the energy consumption using the energy of the original datapath, default backup path, NI, channel wire, and the number of packet hops along the path. The case of no faults is simply the network without the need for the DBP mechanism. The other cases make use of the DBP mechanism to maintain network connectivity in the presence of faults. With two virtual channels, the DBP network consumes only 9.5% more energy on average than the fault-free network when the number of faults increases to 80. As shown in Figures 10(c), 10(d), 10(e) and 10(f), the number of hops increases as the number of faults increases, which decreases the throughput. Because of the longer path hops, the energy consumption of packet transfer between PEs increases in the DBP network.

As shown in Figure 11(a), energy per router via of the default backup path is less than that of the normal router datapath via the internal crossbar as the simpler default backup path bypasses all the internal router modules. Since the default backup path transfers packets via a network interface at an intermediate router, the energy consumption at network interfaces is increased compared to that of the network interfaces in the conventional healthy network. All the above factors affect the total energy consumption of the packet transfer between PEs on the DBP network. It can be seen that the DBP network, especially with virtual channels, does not so much increase the energy consumption in most cases. However, when all router datapaths except default backup paths fail, a unidirectional ring topology manifests and the total energy consumed is drastically increased up to 128% in 16 PEs, and 352% in 64 PEs. This is because the average path hop count is drastically increased, as shown in



(a) Average throughput for 16 PEs

(b) Average throughput for 64 PEs

(c) Accepted traffic vs. latency, no VCs, 16 PEs (average path hop-count is given in parentheses)

(d) Accepted traffic vs. latency, two VCs, 16 PEs (average path hop-count is given in parentheses)

(e) Accepted traffic vs. latency, no VCs, 64 PEs (average path hop-count is given in parentheses)

(f) Accepted traffic vs. latency, two VCs, 64 PEs (average path hop-count is given in parentheses)

Figure 10. Throughput of DBP networks

Figures 10(c), 10(d), 10(e) and 10(f), and the number of hops increases as the number of faults increases.

6. Conclusion

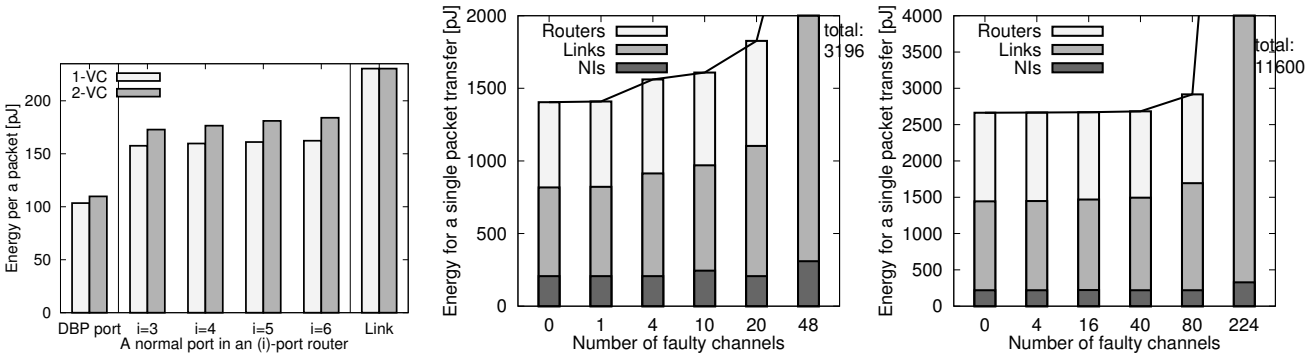
In this paper, we propose a lightweight fault-tolerant mechanism based on the notion of default backup paths which maintain network connectivity in the presence of faults occurring in NoC routers. The mechanism is lightweight in that only a nominal amount of simple redundant hardware is needed to maintain network connectivity. With the DBP mechanism, in addition to the original router datapath, default paths connect certain input and output ports of a router to allow any faulty components along the datapath within a router to be bypassed, such as failed input channel buffers, failed crossbar switches, and other router microarchitectural components. As faulty routers cause the network topology to become partially irregular, deadlock-free paths along the network resources—including the default backup paths—must be ensured. Well-known techniques can be applied straightforwardly to networks employing the default backup path mechanism to guarantee deadlock freedom in routing packets around and through network faults. Evaluation results show that the proposed mechanism increases hardware costs by a modest 12.6% for 2-D mesh wormhole networks while providing graceful performance degradation without chip-wide faulure as the number of faults increases to the maximum possible. The DBP mechanism can, thus, be regarded as serving the role of a lifeline to increase the the lifetime of NoCs and processor chips built from them.

Acknowledgments

This work was partially supported by Joint Research Fund, “Network-on-Chip Architecture”, National Institute of Informatics, JST CREST, and KAKENHI #1880008. It was also supported, in part, by NSF grant CCF-0541417. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] M. Alonso, J. M. Martinez, V. Santonja, P. Lopez, and J. Duato. Power Saving in Regular Interconnection Networks



(a) Power comparison of default backup path, datapath via crossbar, and link

(b) Power comparison of DBP and healthy networks (16 PEs, no VCs)

(c) Power comparison of DBP and healthy networks (64 PEs, two VCs)

Figure 11. Power comparison of DBP and healthy networks

Built with High-Degree Switches. In *International Parallel and Distributed Processing Symposium*, 2005.

[2] A. Banerjee, R. Mullins, and S. Moore. A Power and Energy Exploration of Network-on-Chip Architectures. In *Proceedings of the International Symposium on Networks-on-Chip*, pages 163–172, May 2007.

[3] D.Burger, S.W.Keckler, K. McKinley, M.Dahlin, L. John, C.Lin, C. Moore, J.Burrill, R. McDonald, W.Yoder, and the TRIPS Team. Scaling to the End of Silicon with EDGE Architectures. *IEEE Computer*, 37(7):44–55, July 2004.

[4] J. Duato, O. Lysne, R. Pang, and T. M. Pinkston. Part I: A theory for deadlock-free dynamic network reconfiguration. *IEEE Transactions on Parallel Distributed Systems*, 16(5):412–427, 2005.

[5] J. Duato, S. Yalamanchili, and L. Ni. *Interconnection Networks: an engineering approach*. Morgan Kaufmann, 2002.

[6] J. Flich, P. Lopez, M. P. Malumbres, and J. Duato. Boosting the Performance of Myrinet Networks. *IEEE Transactions on Parallel and Distributed Systems*, 13(7):693–709, July 2002.

[7] C. J. Glass and L. M. Ni. The Turn Model for Adaptive Routing. *Proceedings of International Symposium on Computer Architecture*, pages 278–287, 1992.

[8] J.C.Sancho, A.Robles, and J.Duato. An effective methodology to improve the performance of the up*/down* routing algorithm. *IEEE Transactions on Parallel and Distributed Systems*, 15(8):740–754, Aug. 2004.

[9] J. Kim, C. A. Nicopoulos, D. Park, N. Vijaykrishnan, M. S. Yousif, and C. R. Das. A Gracefully Degrading and Energy-Efficient Modular Router Architecture for On-Chip Networks. In *International Symposium on Computer Architecture*, 2006.

[10] M. Koibuchi, A. Jouraku, K. Watanabe, and H. Amano. Descending Layers Routing: A Deadlock-Free Deterministic Routing using Virtual Channels in System Area Networks with Irregular Topologies. In *Proceedings of the International Conference on Parallel Processing*, pages 527–536, Oct. 2003.

[11] O. Lysne, T. M. Pinkston, and J. Duato. Part II: A methodology for developing deadlock-free dynamic network reconfiguration processes. *IEEE Transactions on Parallel Distributed Systems*, 16(5):428–443, 2005.

[12] H. Matsutani, M. Koibuchi, and H. Amano. Performance, cost, and energy evaluation of fat h-tree: A cost-efficient tree-based on-chip network. In *Proc. of the IEEE International Parallel and Distributed Processing Symposium (IPDPS'07)*, Mar. 2007.

[13] G. Michelogiannakis, D. Pnevmatikatos, and M. Katevenis. Approaching Ideal NoC Latency with Pre-Configured Routes. In *Proceedings of the International Symposium on Networks-on-Chip*, May 2007.

[14] S. Murali, T. Theocharides, N. Vijaykrishnan, M. J. Irwin, L. Benini, and G. D. Micheli. Analysis of Error Recovery Schemes for Networks on Chips. *IEEE Design and Test of Computers*, 22(5):434–442, 2005.

[15] V. Puente, J. Gregorio, F. Vallejo, and R. Beivide. Immunet: a cheap and robust fault-tolerant packet routing mechanism. In *International Symposium on Computer Architecture*, pages 198 – 209, June 2004.

[16] M. D. Schroeder et al. Autonet: a high-speed, self-configuring local area network using point-to-point links. *IEEE Journal on Selected Areas in Communications*, 9:1318–1335, 1991.

[17] V. Soteriou and L.-S. Peh. Design-Space Exploration of Power-Aware On/Off Interconnection Networks. In *Proceedings of the International Conference on Computer Design*, pages 510–517, Oct. 2004.

[18] STI. Cell broadband engine documentation. *available at www.ibm.com/developerworks/power/cell, http://cell.scei.co.jp*, Aug. 2005.

[19] M. B. Taylor et al. The Raw Microprocessor: A Computational Fabric for Software Circuits and General Purpose Programs. *IEEE Micro*, 22(2):25–35, Apr. 2002.

[20] M. Thottethodi, A. R. Lebeck, and S. S. Mukherjee. BLAM : A High-Performance Routing Algorithm for Virtual Cut-Through Networks. In *International Parallel and Distributed Processing Symposium*, page 45b, 2003.

[21] W.J.Dally and B.Towles. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann, 2003.