Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.470 – 477

RESEARCH ARTICLE

FAULT TOLERANT DEFLECTING ROUTER WITH HIGH FAULT COVERAGE FOR ON-CHIP NETWORK

[1]Mr. Vishnu K P, [2]Mr. T Shanmuganathan

[1]PG Scholar, Department of Electronics and Communication Engineering,
Hindustan University, Chennai, Tamil Nadu, India

[2]Assistant Professor, Department of Electronics and Communication Engineering,
Hindustan University, Chennai, Tamil Nadu, India
[1] kp3670@gmail.com; [2] thangatamizh@gmail.com

Abstract*— Continuous scaling of CMOS technology makes it possible to integrate a large number of heterogeneous devices that need to communicate efficiently on a single chip. For this efficient routers are needed to takes place communication between these devices. As the chip scales, the probability of both permanent and transient faults is also increasing, making Fault Tolerance (FT) a key concern in scaling chips. This project, proposes a fault-tolerant solution for a bufferless network-on-chip, including an on-line fault-diagnosis mechanism to detect both transient and permanent faults, a hybrid automatic repeat request, and forward error correction link-level error control scheme to handle transient faults and a reinforcement-learning-based fault-tolerant deflection routing (FTDR) algorithm to tolerate permanent faults.

***Keywords*— Deflection routing; Fault-tolerance; On-line fault diagnosis; Permanent fault; Transient fault**

1. INTRODUCTION

As the technology scales down, the gate delay decreases, but the wire delay increases relatively and this global wire delay becomes the main factor which can decide the overall performance. Difficult timing closure becomes the main problem among many design issues which is caused by long global wire delay. In order to solve these long global wire delay and scalability issues, many studies suggested the use of a packet based communication network which is known as Network-on-Chip (NoC). NoC approach has emerged as a promising solution for on-chip communications to enable integrating various processors and on-chip memories into a single chip.

As the CMOS technology scales down to the nanometer domain, smaller feature size, lower voltages and higher frequencies increase the number and higher of occurrence of intermittent and transient faults, besides manufacturing defects and wear out effects which lead to permanent faults are also inevitable. There are mainly two types of faults which come across

NoC architectures, namely transient faults and permanent faults. The methods used to deal with these faults are flow control based and fault tolerant routing methods. A good fault-tolerant routing algorithm should ensure "0 lost packet" in whatever fault patterns as long as a path exists

In order to have higher speed and lower cost than a wormhole or a virtual channel router or bufferless routers are incorporated on NoC. Except one input register for each input port, there are no other buffers in the bufferless router. Due to the lack of buffers, deflection routing is utilized in the bufferless router to route packets to neighbouring routers immediately without buffering in the router. The complete adaptive feature of deflection routing provides the potential to route packets to avoid faulty links/routers and achieve fault-tolerance.

Here, in order to avoid transient and permanent faults a fault-tolerant solution, including an on-line fault diagnosis mechanism, a link-level error control scheme, and a fault tolerant routing algorithm is proposed for the bufferless NoC. Here the fault diagnosis mechanism uses the single-error-correcting and double-error-detecting (SECDED) Hamming code to detect both transient and permanent link faults. A hybrid automatic repeat request (ARQ) and forward error correction (FEC) link-level error control scheme using retransmission is proposed to handle transient faults.

The FTDR algorithm guarantees zero lost packets as long as the fault pattern does not cut the network into two or more disconnected parts. Simulation results demonstrate that under synthetic workloads, in the presence of permanent link faults, the throughput of an 8 × 8 network with FTDR algorithm is 14% higher on average than that with the fault-on-neighbor (FoN) aware deflection routing algorithm and the cost-based deflection routing algorithm respectively. It also achieves almost 2 × less hop counts on average than that with the cost-based algorithm. Under real application workloads, the FTDR algorithm achieves 20% less hop counts on average than that of the FoN algorithm.

2. FAULT DETECTION MECHANISMS FOR NETWORK ON CHIP

In order to run a fault-tolerant system smoothly the first thing to be done is to detect the location of the faults. The fault detection mechanism should also be able to distinguish transient faults from permanent faults. In order to detect transient link errors the methods used are error coding techniques, such as cyclic redundancy check and parity codes. To detect permanent errors in NoC there is an in-line test method to test each adjacent pair of wires and a syndrome storing-based error detection method based on evaluation of consecutive code syndromes at the receiver. And there are also few works focusing on detecting transient faults and permanent faults at the meantime.

There are mainly three techniques to handle transient faults in NoC and they are Automatic repeat request (ARQ), Forward error correction (FEC), and Hybrid ARQ (HARQ). Also transient faults can be handled at both link-level and transport level. In ARQ-based error control the packet is retransmitted if it is found to have errors. Such packets are retransmitted until it is received error free. The error detection is usually implemented through a cyclic redundancy check (CRC). A simple error-detecting code is applied to the packet before transmitting, and at the receiver side a checksum will be calculated to ensure that no error has occurred. If the checksum does not add up to the right value, the packet is retransmitted.

FEC or channel coding is a method used to increase the performance of error control. This is achieved by the use of error-correcting codes (ECCs) that add redundancy to the packet, which allows a certain amount of bit-errors to be detected and

corrected at the receiver side. The main drawback is the cost of the redundancy, the parity bits, which increase the packet size. ARQ provides reliable communication through retransmissions, which will be costly in poor channels where retransmissions occur frequently. FEC performs better in poor channels, while the redundant bits become an undesired cost when channel conditions are good. HARQ schemes exploit the advantages of both, by methods of combining ARQ and FEC.

The methods used to handle permanent faults are stochastic and deterministic fault tolerant routing. Stochastic communication transfers redundant packets through different paths to avoid faults. Deterministic fault-tolerant routing algorithm is based on the shape of the fault region .Based on the fault region it can be categorized into two classes: one can handle regular fault regions (example convex and concave shapes) and the other, which is also known as topology-agnostic, can handle irregular fault regions.

3. NoC ARCHITECTURE AND FAULT DIAGNOSIS

The NoC architecture is based on a 2-D mesh topology, Nostrum NoC. Each processing element is attached to a router (R), as shown in Fig. 1. The difference from the ordinary 2-D mesh is that the boundary output is connected to the input of the same router. This can be viewed as an additional packet buffer. All incoming packets are prioritized according to their hop counts, which record the number of hops the packet has been routed. The router makes routing decision for each arriving packet from the highest priority to the lowest.

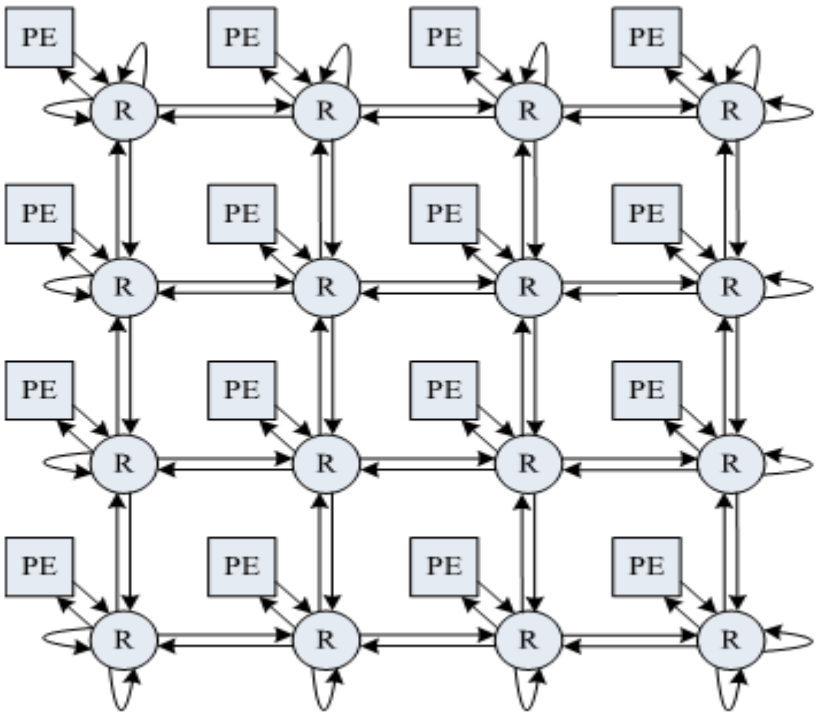

Figure 1: NoC architecture

The basic data transfer unit in this paper is a packet. A packet, which has 114 bits, contains a 34-bit head and an 80-bit payload. A valid bit (V) is used to mark a packet valid or not. Here, faults are considered as faulty links which can be both transient and permanent faults. For deflection router, the number of input ports should be equal to the number of output ports, so permanent link failures are assumed to be bidirectional. In each router, a four-bit fault vector is used to represent the fault status of its four links (North, East, South, and West). A "1" in the fault vector represents the corresponding bidirectional links are broken. The faulty region can be any shape as long as it does not disconnect the network.

4. EXISTING METHODS

4.1. LINK-LEVEL FAULT DETECTION AND PROTECTION

To perform fault diagnosis we use single error correcting double error detecting hamming code (SECDED) technique. SECDED can correct single bit errors and detect double errors. To correct multiple error bits, the Bose, Chaudhuri and Hocquenghem (BCH) code can be considered however, as the number of bits to be corrected increases, the hardware complexity and decoding time will increase significantly.

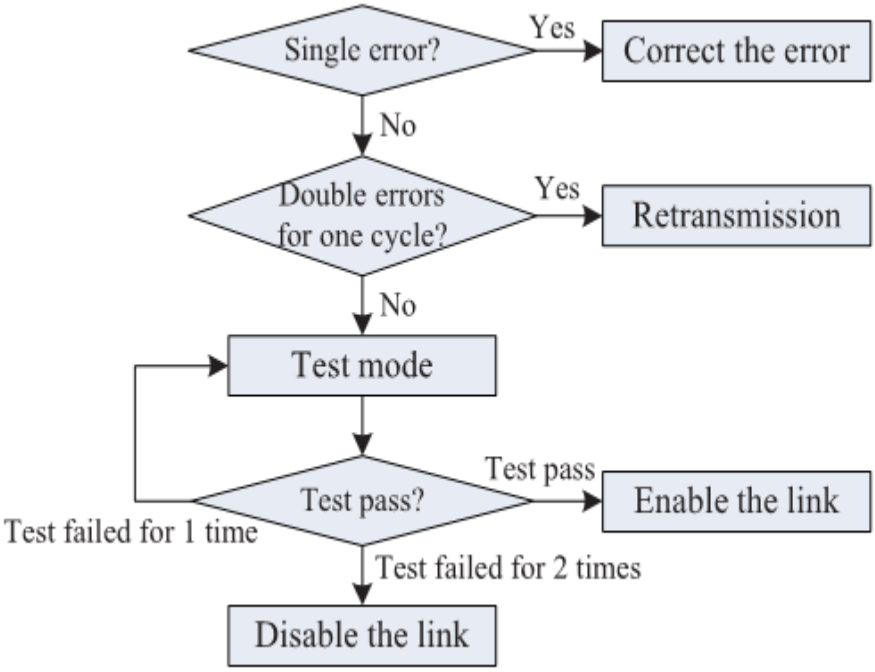

Figure 2: Fault diagnosis process.

The above mentioned flow chart can be explained as, if it is a single bit error whatever the error may be it will correct it automatically. If it detects a two-bit error in any one part of the encoding packet for one cycle, which is considered as a transient fault, it will require the upstream router to retransmit the packet. If the syndromes of two consecutive received packets are the same, which means the retransmitted packet contains the same two-bit error, in order to check whether the link contains real permanent faults, the router enters into a test mode to test this faulty link by applying test vectors. The test process will be

conducted at most twice. In the test mode, the link being tested is temporarily disabled. The downstream router requires the upstream router to send the test vectors by setting a test_initial signal to "1". If the downstream router detects any one of the four test vectors still containing the same error, the test process will be conducted again. If the second test still fails, the link is marked as a permanent faulty link. If all tests pass, the link will be enabled again.

4.2. LINK-LEVEL ERROR CONTROL SCHEME

The different methods used to eliminate transient faults are ARQ, FEC and hybrid ARQ and the algorithms for avoiding permanent faults are cost based routing algorithm and fault on neighbor aware routing algorithm. In ARQ scheme the packet is retransmitted if it is found to have errors. And it is retransmitted until the packets are reached error free. The error detection is usually implemented through a cyclic redundancy check (CRC). A simple error-detecting code is applied to the packet before transmitting, and at the receiver side a checksum will be calculated to ensure that no error has occurred. FEC is a method used to increase the performance of error control. FEC is also called channel coding. This is achieved by the use of error-correcting codes (ECCs) that add redundancy to the packet, which allows a certain amount of bit-errors to be detected and corrected at the receiver side. The main drawback is the cost of the redundancy, the parity bits, which increase the packet size. FEC introduce encoding and decoding costs.

The ARQ scheme using retransmission performs well without incurring much latency for low error rates; however, at higher error rate the hybrid ARQ/FEC scheme with slight hardware overhead provides better performance than the pure ARQ scheme. Here we use a hybrid ARQ scheme for error control to tolerate transient faults during packets transmission. In the case of a single-bit error in any part of the packet, the error can be corrected after the packet has been decoded. If any part of the packet contains a two bit error for one cycle, the router which receives the packet will require the router, which sends the packet, to retransmit the packet. The hardware structure of the hybrid ARQ is shown below.

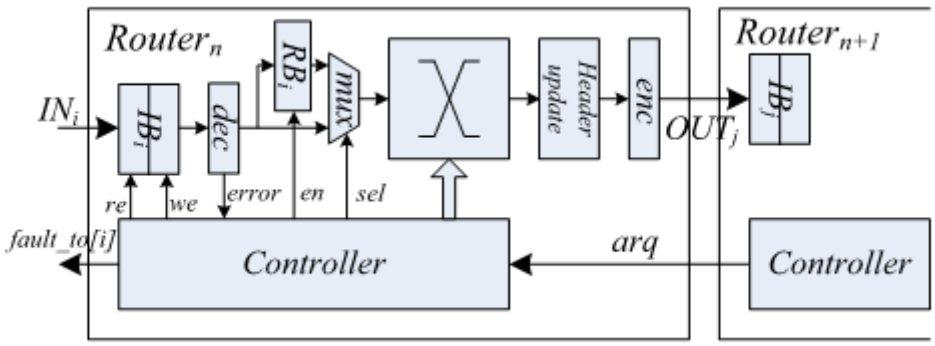

Figure 3: Hardware structure of link-level error control scheme.

Here for each input port of the router we have an input buffer (IBi) with two entries instead of the original one and the boundary input port of the boundary router has an input buffer with three entries. Additionally, a retransmission buffer is used to buffer the packet which may be retransmitted. After decoding, the packet will be written to retransmission buffer .A2-to-1

multiplexer is used to select to send a new packet or retransmit the last packet. A request signal arq is introduced between two neighbouring routers to indicate whether the last packet should be retransmitted or not. The fault information transmission signal (fault_to[i]) is used to disable the outgoing link i of the upstream router temporarily.

5. PROPOSED METHOD

In order to have a high throughput system one must have system which should handle both transient faults and permanent faults. In our existing work we have methods like hybrid ARQ scheme to avoid transient faults. But we must also incorporate algorithms which can handle permanent faults. In our proposed system a fault tolerant deflecting algorithm (FTDR) is presented to deal with permanent faults. The earlier methods used to deal with permanent faults are cost based and fault on neighbor (FoN) aware routing algorithms.

5.1 Q-ROUTING

Q-routing is an adaptive routing algorithm based on a variant of the reinforcement learning-Q-learning, which makes routing decision using only local information without having to know the network topology in advance. Q-routing is a table-based routing algorithm. For example, the routing table entry Q^x(d,y) denotes the lowest estimated delivery time from x to d through neighbor y. If the router x sends a packet to d through y, x receives the minimum estimated delivery time from y to d ($min_z\ Q^y_{t-1}$(d, z)) after the packet is sent to y. Then the estimated delivery time Q^x (d,y) can be updated, as shown

$$Q^x_t(d,y) = (1-\alpha)\ Q^x_{t-1}(d,y) + \alpha\ (b^x_t + min_z\ Q^y_{t-1}(d,z)) \quad (1)$$

5.2 FTDR ALGORITHM

For deflection routing, we use the number of hops to destination as Q-value instead of the estimated delivery time to build the routing table Q in each router. $Q^x(d,y)$ is defined as the minimum number of hops from x to d through y. Different from the estimated delivery time Q-value of the original Q-routing , $Q^x(d,y)$ is a deterministic value which is equal to one hop plus the minimum number of hops from y to d (defined as $min_z Q^y(d,z)$). The FTDR switch architecture can be shown as follows:

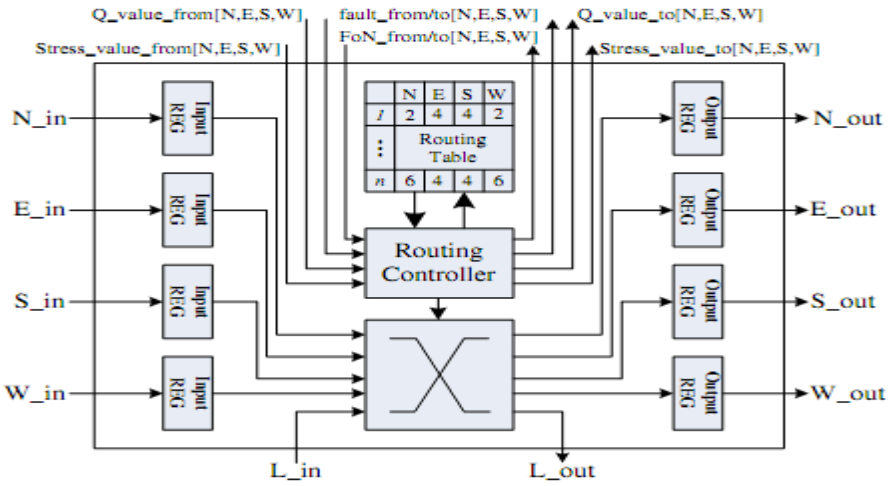

Figure 4: FTDR switch structure

When a router x sends a packet to d through y, y returns the minimum number of hops from y to d back to x. Then x updates the corresponding entry with one hop plus the minimum number of hops from y to d ($min_z\ Q_{t-1}^{y}(d,z)$). Our deflection router has only one input register, which means the packet does not have to wait in the router, sob_t^x is 1. There are n × m entries in the routing table, where n is the number of routers in the network and m is the number of neighbouring routers. For 2-D mesh, m is four. For a given topology, the initial routing table is fixed. The router chooses a direction with the minimum number of hops to destination to send a packet. In the case of several directions with equal number of hops to destination, the router will choose one of them with the smallest stress value.

If there is no fault in the network, the routing table cannot be updated. If one link of the router is broken or temporarily disabled during testing, all table entries corresponding to this direction are set to "∞". After a learning period, the table entries will converge to a fixed value which denotes the minimum number of hops from each port to each destination. Additionally, we use the two-hop fault information to reduce the average hop counts. If a router detects that one of its neighbours y along direction d has only one link not faulty based on the two-hop fault information, the table entries from d to all destinations except y are set to "∞". If a two-hop link is faulty (FoN_from[d] [j] = 1, j ∈ {North, East, South, West}), the table entries from d to all destinations along j are updated with the previous table entry plus 2. If the temporarily disabled link is enabled again, all entries corresponding to this direction are updated from "∞" to the minimum number of hops from this direction to the destination. If the Q-value is not zero, the corresponding table entry will be updated with the Q-value.

6. RESULTS AND OBSERVATIONS

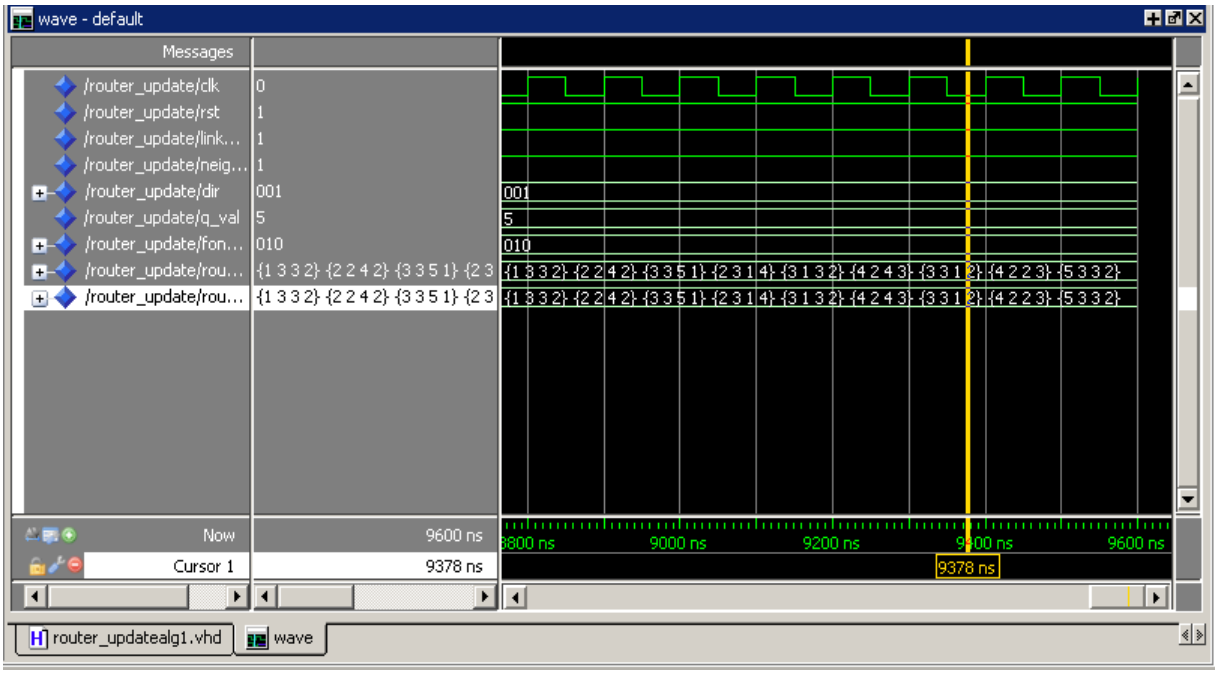

Figure 5: Simulation output for FTDR algorithm

7. CONCLUSION

In this work, a fault-tolerant solution for a bufferless NoC to protect it from both transient and permanent faults on the links has been explained. The main contributions of this paper can be summarized as follows, first of all an on-line fault diagnosis mechanism which utilizes SECDED Hamming code to detect both transient and permanent faults is explained. Then a hybrid ARQ scheme, which can achieve graceful degradation even at a high fault rate, is used to tolerate transient errors during transmission. A FTDR algorithm is proposed which guarantee "0 lost packet" reconfigures the routing table through a reinforcement learning method to route packets avoiding permanent faults. Compared to previous works we have incorporated techniques which will eliminate both transient and permanent faults. The throughput of FTDR is 14% and 23% higher on average than FoN and cost-based deflection switch respectively.

REFERENCES

[1] A. Kohler, G. Schley, and M. Radetzki, "Fault tolerant network on chip switching with graceful performance degradation," IEEE *Trans. Comput.-Aided Design Integr. Circuits Syst*., vol. 29, no. 6, pp. 883–896, Jun. 2010.

[2] C. Constantinescu, "Trends and challenges in VLSI circuit reliability," *IEEE Micro*, vol. 23, no. 4, pp. 14–19, Jul.–Aug. 2003.

[3] A. Patooghy and S. G. Miremadi, "XYX: A power & performance efficient fault-tolerant routing algorithm for network on chip," *in Proc.17th Euromicro Int. Parallel, Distrib. Netw. Based Process. Conf.,* 2009, pp. 245–251.

[4] C. Feng, Z. Lu, A. Jantsch, J. Li, and M. Zhang, "FoN: Fault-on -neighbor aware routing algorithm for networks-on-chip," in *Proc. 23rd IEEE Int. SoC Conf.,* Sep. 2010, pp. 441–446.

[5] D. Bertozzi, L. Benini, and G. De Micheli, "Error control schemes for on-chip communication links: The energy-reliability tradeoff*," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.,* vol. 24, no. 6, pp. 818–831, Jun. 2005.

[6] H. Zimmer and A. Jantsch, "A fault model notation and error-control scheme for switch-to-switch buses in a network-on-chip," in *Proc. 1st IEEE/ACM/IFIP Int. Conf. Hardw./Softw. Codesign Syst. Synth.*Oct.2003, pp. 188–193.

[7] M. Hayenga, N. E. Jerger, and M. Lipasti, "SCARAB: A single cycle adaptive routing and bufferless network," in *Proc. 42nd Annu. IEEE/ACM Int. Symp. Microarch. ,* Dec. 2009, pp. 244–254.

[8] M. Pirretti, G. M. Link, R. R. Brooks, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "Fault tolerant algorithms for network-on-chip interconnect," in Proc. IEEE Comput. Soc. Annu.Symp. VLSI, Feb. 2004 pp. 46–51.

