Fault Injection-based Evaluation of a Synchronous NoC Router

Ashkan Eghbal, Pooria M. Yaghini, Hossein Pedram, and Hamid R. Zarandi
Amirkabir University of Technology
E-mails : {ashkanxy, pooria.yaghini, pedram, h_zarandi}@aut.ac.ir

Abstract

This paper evaluates fault-tolerant behavior of an NoC router through simulation-based method. A structural-level VHDL environment has been employed to estimate fault injector signal's (FIS) effects. Different fault models such as dead clause, stuck-then, micro-operation, crosstalk, and SEU have been injected to evaluate the transient faults' effects. According to the results, up to 48% of the injected faults cause system failure and also about 51% are overwritten before turning into errors. Less than 1% of injected faults treat as latent errors. The average of fault latency has been investigated as 194ns. Almost 70%, 31%, and 35% of injected faults are overwritten in buffer, routing unit, and switch components, respectively. Routing unit is also recognized as the most tenuous component.

1. Introduction

The emergence of nano-scale circuits accompanies the advantage of employing more gates in a single chip. Utilizing more components reminds the necessity of employing superior connection infrastructure in place of traditional System-on-Chip (SoC). The Network-on-Chip (NoC) infrastructure has been considered to accommodate better modularity, scalability, and higher bandwidth compared to bus-based infrastructure. Controlling the physical parameters in the fabrication process is unexpected due to the shrinkage of the technology size [1]. Fault occurrence such as crosstalk, single event upset (SEU), and single event transient (SET) caused by electromagnetic interferences (EMI), alpha particles' strikes, and cosmic radiation can affect the functionality of NoC router or, eventually lead it to failure [2]. Fault-tolerant design (FTD) has been considered to reduce or mask such phenomena. Rerouting algorithms have been proposed to update the routing table [3][4]. The idea of bypassing faulty data paths within failed routers has been proposed as light weight fault tolerant method [5]. Only SEU and cross talk fault effects have been evaluated on a synchronous NoC router [6][7]. In some cases the reliability of NoC router has been assessed by the help of analytical method imposing some assumptions which might not be accepted. In [8], an analytical model of an NoC architecture has been presented with the assumption of not being the faulty and destination router at same columns in a mesh topology. An exhaustive research seems necessary to find the most tenuous components in an NoC router while keeping the cost of overhead low. Such experimental results can help the designer to utilize well-chosen fault-tolerant methods based on the application. This experiment evaluates the reliability of NoC router against various fault models through simulation-based method with the help of SINJECT. SINJECT is a fault injection tool which examines the robustness of modeled system in Verilog or VHDL[9].

This paper is organized as follows: Section 2 introduces the implemented NoC router in this experiment. Part 3 includes the fault injection characteristics. The experimental results have been considered in different aspects in fourth portion. Finally, section 5 delivers some conclusion remarks.

2. Synchronous router architecture

The architecture of the synchronous NoC router applying in this experiment has been indicated in Figure 1. It is apt to be employed in both mesh and torus topology supporting five bidirectional ports. It has been synthesized with the help of *Leonardo*, which results in 38560 gates. It consists of three main components including input buffer, routing unit, and switch component. Two state diagrams in the buffer component are responsible to take and to transmit a flit in a single clock. The upper state diagram is triggered in positive edge of clock and the lower one is stimulated in negative edge of clock. Employing both edges of clock increases the performance of NoC router. The routing unit component is the central unit including sub components such as buffer management, header processor, arbitration unit, and channel allocator. The buffer management unit is in charge of controlling the buffer space to preclude losing older flits before their transmission. The buffer is implemented as a circular queue to optimize of applying buffer efficiently. New routing decision is made by implementing *XY* routing algorithm once a new header flit has been detected and the previous

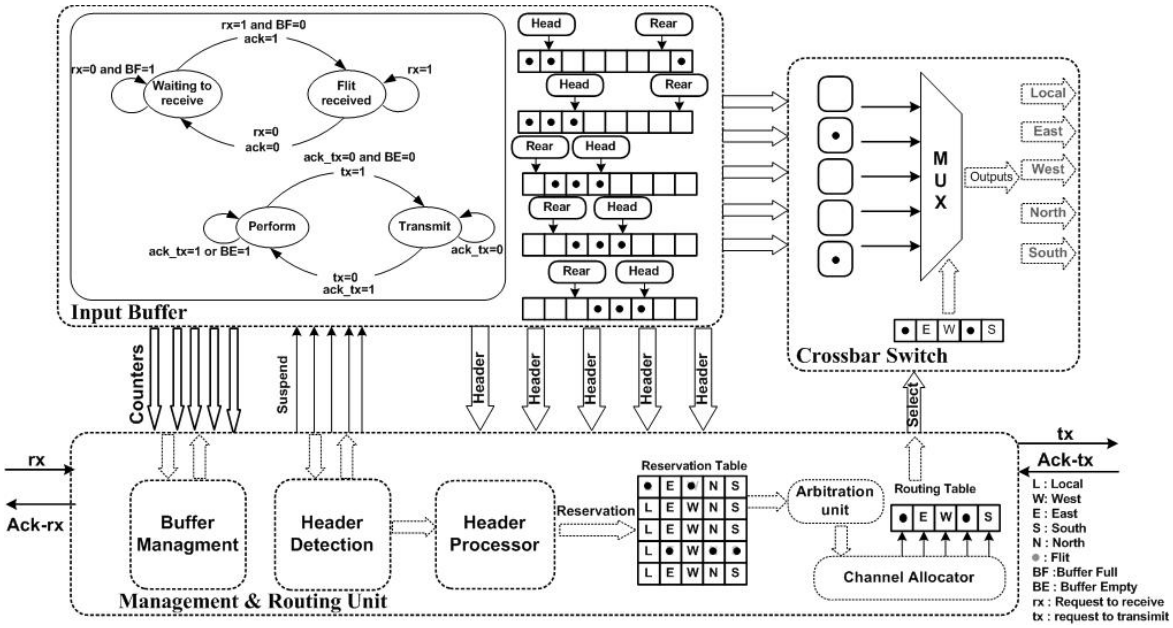

Figure 1. A synchronous NoC router architecture

previous packet is completely transmitted. Each input channel reserve one of the output ports when routing process has been accomplished. An arbitration unit is employed to lock the dedicated output channel until the end of transmitting a complete packet. The initial state of router is called *Waiting to receive*. An Incoming request signal, *rx*, changes the state of router into *flit received* state in following positive edge of the clock if input buffer unit is not full. Accomplishing this transmission sets the *ack* signal to high. In the negative edge of the clock, router transmits the stored flits of the buffer. The system turns into *transmit* state if there exists any routed flit in buffer and the selected output port is not occupied by the other input channels. The third component is a switch. The controlling signals of routing unit select one of output port for an incoming header flit. All the remaining flits of a packet follow their header by implementing the wormhole switching. They are blocked if their header is impeded on its way toward destination. The dedicated port gets free to serve other input channels once a packet transmission has been finished.

3. Fault injection characteristics

Transient faults are much more common than permanent ones in upcoming digital systems [10]. Such experiment examines the effects of transient faults by activating each FIS for a short period of time which a flit needs to pass a router. The injected faults treat as a permanent if the selected life time equals the total running time of experiment. 112 FISes have been injected into the VHDL model of the NoC. The employed NoC router is simulated with a 10ns clock period. According to the part 2 a flit needs one clock period to reach its destination in normal condition. The faults have been injected randomly by the average duration of 20ns (The fault duration is exponentially distributed). The FISes are activated with a uniform distribution between 200ns to 8000ns, in normal condition. The fault injection of each FIS has been repeated 100 times. The FISes are distributed through main components based on their complexity. This experiment considers single fault effects. In other words, the FISes are activated and their effects are observed separately. Table 1 specifies the infected points and injected fault models in more detail. Five different fault models, including dead clause, stuck-then, micro-operation, crosstalk, and SEU have been injected into the VHDL model of NoC router. Switch component is infected by fewer FISes in contrast to the other components. SEU fault model affects sequential circuits. Based on the router specification in part 2, the switch consists of a combinational circuit without any decision making functionality. SEU fault models are injected into routing unit and buffer components. Both of such components consist of sequential circuits and decision making subcomponents.

4. Experimental results

Table 2 shows the number and percentage of overwritten faults, latent errors, failure experiments, and the average of fault latency for each of component separately and in total. The results are based on the comparison of golden run and faulty run of simulation. According to the result, 69.76%, 31.17%, and 35.44% of real injected faults are overwritten (replaced by new

Table 1. Fault injection distribution

Fault model	Buffer	Routing-unit	Switch
Dead-clause	6	10	6
Stuck-then	2	8	6
Micro-operation	3	5	----------
Crosstalk	9	12	12
SEU	13	20	----------
Total	33	55	24

values before being detected), in buffer, routing unit, and switch components, respectively. Switch component got the second rank of failure rate although it is infected less than the others. That might because of the high activity of this component which is involved per each flit's transmission. The highest failure results are related to the routing unit due to its decision making role. Figure 2 shows the percentage of overwritten faults in progressive time. The highest ratio of overwritten faults can be found in buffer component and the lowest one relates to the routing component. That might be behind the fact that an eight flit buffer has been employed to implement a wormhole switching in this architecture. Fault occurrence in input buffer might be overwritten at most each eight clock whenever the new flit is written to the buffer. On the other hand, the routing unit's values are updated whenever the incoming flit is header. The fault occurrence on data-payload flits are not detected in this experiment since these types of failures must be distinguished in the destination IP core. Only fault effect on header flit in buffer component can result in miss routing and consequently a failure.

5. Conclusion

In this experiment a synchronous NoC router architecture has been introduced that is capable of receiving and transmitting a flit in a single clock. The fault-tolerant behavior of the router has been evaluated to keep the redundancy overheads low in upcoming designs. Routing unit has been detected as the most tenuous component among the all. The overwritten fault percentage of this component is reported as the lowest one. An information redundancy as an error correction remedy might be best choice to impose lower hardware penalty while achieving the robustness. Some error detection techniques which need retransmission packets dramatically influence the performance of the NoC router. Employing some fault masking method to reduce the fault propagation might be another approach of strengthening the NoC router.

Table 2. Fault injection results

Module	Overwritten Faults		Latent Errors		Failure Experiments		Average of Fault Latency (ns)
	#	%	#	%	#	%	
Input Buffer	2015	69.76	5	1.14	845	29	241
Routing Unit	760	31.17	5	0.83	1665	68	174
Switch	168	35.44	2	0.56	306	64	166
Total	2943	51	12	0.2	2816	48.8	194

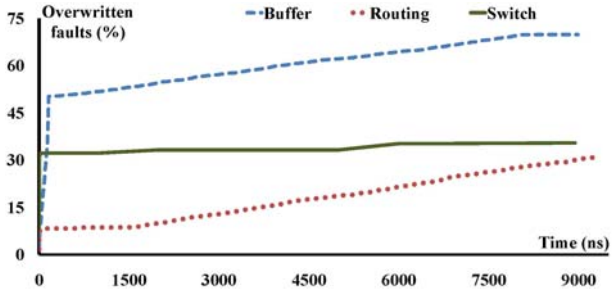

Figure 1. Overwritten faults percentage versus time

References

[1] 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 & Test of Computers*, Volume 22, Issue 5, 2005, pp. 434-442 .

[2] C. Grecu, L. Anghel, P.P. Pande, A. Ivanov, and R. Saleh, "Essential Fault-Tolerance Metrics for NoC Infrastructures", *13th IOLTS*, 2007, pp. 37-42.

[3] M. Ali, M. Welzl, and S. Hellebrand, "A Dynamic Routing Mechanism for Network on Chips", *Norchip Conference*, 2005, pp. 70-73.

[4] M. Ali, M. Welzl, S. Hessler, and S. Hellebrand, "An Efficient Fault-Tolerant Mechanism to Deal with Permanent and Transient Failures in a Network on Chip", *International Journal of high Performance System Architecture*, Vol.1, No.2, 2007, pp. 113-123.

[5] M. Koibuchi, H. Matsutani, H. Amano, and T.M. Pinkston, "A Lightweight Fault-Tolerant Mechanism for Networks-on-Chip", *Second ACM/IEEE International Symposium Networks-on-Chip*, 2008, pp.13-22.

[6] A.P. Frantz, M. Cassel, F.L. Kastensmidt, E. Cota, and L. Carro, "Crosstalk and SEU Aware Networks on Chips", *IEEE Design and Test of Computers*, 2007, pp.340-350.

[7] A.P. Frantz, F.L. Kastensmidt, L. Carro, and E. Cota, "Dependable Network-on-Chip Router Able to Simultaneously Tolerate Soft Errors and Crosstalk", *IEEE International test Conference*, 2006, pp. 1-9.

[8] A. Dalirsani, M. Hosseinabady, and Z. Navabi, "An Analytical Model for Reliability Evaluation of NoC Architectures", *13th IOLTS*, 2007, pp. 49-56.

[9] H.R. Zarandi, S.G. Miremadi, and A. Ejlali, "Dependability Analysis Using a Fault Injection Tool Based on Synthesizability of HDL Models", *Proceedings of the International Symposium on Defect and Fault Tolerance in VLSI Systems*, 2003, pp. 485-492.

[10] R. C. Baumann, "Soft Errors in Commercial Integrated Circuits", *International Journal of High Speed Electronics and Systems*, Vol. 14, No. 2, 2004, pp. 299-309.