A Fault Tolerant Mechanism for Networks-on-Chip

Yerkingali Tileugali

Abstract— Advances in CMOS integrated circuit fabrication technology has made it possible to integrate a large number of transistors in a single chip. Although this has considerably increased their performance, it has also increased their vulnerability to wear out and failure mechanisms. For example failure of a single wire in the on chip communication structure of the system can lead to system and device failure. In this paper we address the permanent failures that can happen in the on-chip links and render the system inoperable. By re-employing the healthy wires in a partially failed link, our methodology is capable of maintaining the connectivity in the communication system, even after some of the links fail.

Index Terms— Networks on chip, Fault tolerance, Serialization, Verilog, FPGA, Link.

1 Introduction

Networks on chip (NoC) (shown in Fig. 1) are adopted as communication structure of the modern high performance multi-core systems on chip (MPSoC) [1], [2]. This is due to their higher performance compared to traditional bus based system [3], [4]. However, NoC is no exception to the effects of wear out and failure mechanisms (such as TDDB, NBTI, and SM) that are accelerated in modern integrated circuits [5]. The effect of these failures on NoC is even more destructive because failure of a single wire in NoC structure can lead to NoC and eventually system failure.

One of the strong points about NoC that has made it a system of choice in design of communication structure of MPSoC is its modularity. This feature presents an excellent opportunity to use fault tolerant design methods in its design and implementation.

The problem of link failure in NoC can be addressed in different ways. One of the popular methods for example is to use dynamic routing algorithms [6][7][8][9][10]. Researchers have implemented and proposed different dynamic routing algorithms that can deal with this problem. The main idea here is to continuously monitor links. This is usually done through integrating a fault detection mechanism. Once a failed link is identified the fault detector system informs the routing controller about the link failure and its location. The routing controller then uses this information to run its routing algorithm to identify new routing paths that would bypass the failed link. Basically this method is based on identifying the faulty links and avoiding routing packets through them. The main draw backs with this method are very high cost of implementation and that most of the currently available methods cannot guarantee that deadlock and livelock situations will not occur in the system.

In this paper, we propose a new methodology to deal with the problem of link failure in the concept of NoC. Our proposed methodology is based on using the remaining healthy wires in a faulty link to maintain connectivity between upstream and downstream routers.

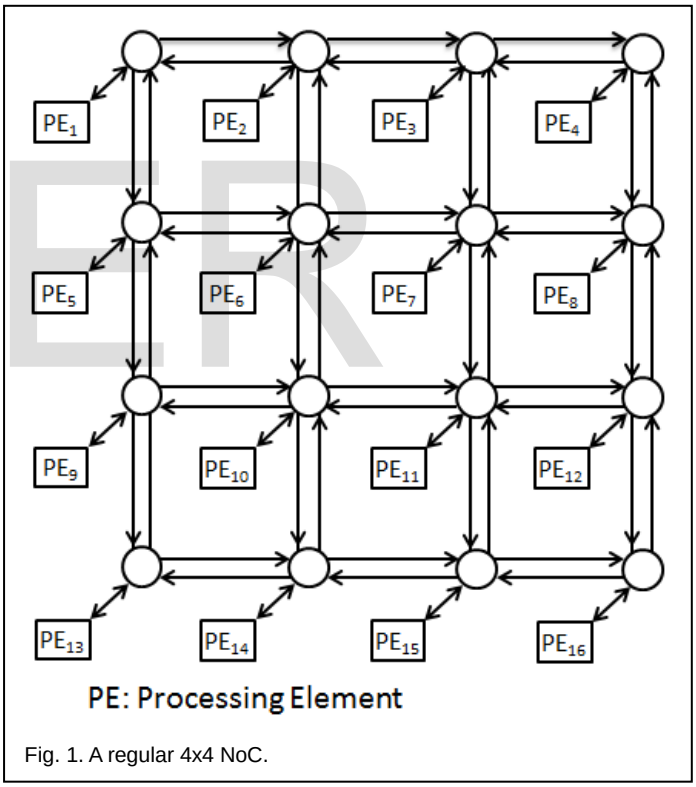

Fig. 1. A regular 4x4 NoC.

2 Proposed Fault Tolerant Link

The main idea behind our proposed fault tolerant link is to utilize the remaining functional wires in a broken link to maintain the routers that the link connects connected. We employ a splitter in the transmitter side to break down the data to be transmitted into several smaller packets and send them over the remaining functional wires. In the receiver side we employ an assembler to rebuild the original data sent by the transmitter. This process is illustrated in Fig. 2. In the example depicted in this figure, router_A is connected to router_B through a 4-bits wide link from which two of them that are marked

• *Yerkingali Tileugali, is with Shakarim State University of Semey, Kazakhstan. Email: erkin1207@gmail.com*

International Journal of Scientific & Engineering Research Volume 5, Issue 7, July-2014
ISSN 2229-5518

with x are failed. We arm all the links in the NoC with a novel failure detection circuit that is presented in [11]. The authors in [11] have proposed an FSM based simple but efficient failure detection mechanism for NoC links that is capable of monitoring the links and detecting and reporting the link failures as soon as they happen. The important point about this method is its low cost of implementation that makes it scalable to bigger networks. Once we detect the failed links, the transmitter first breaks down the data to be transmitted to the downstream router into smaller packets and then uses the remaining healthy links to transmit the message to $router_B$. When $router_B$ receives all of the packets, it uses the assembler to rebuild the original message.

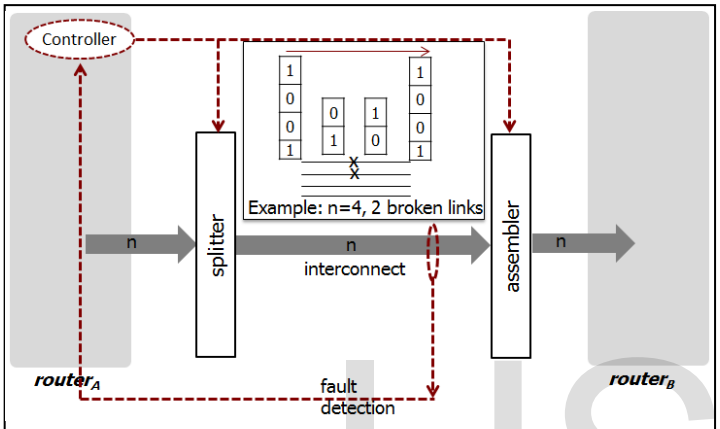

Fig. 2. Utilizing the remaining healthy links in a broken link to mainitain connectivity between $router_A$ and $router_B$.

3 NoC Router Architecture

Fig. 3 illustrates the conventional NoC router architecture [1]. The main blocks in this structure are as follows: 5 input and output ports. Usually input ports are formed by multiple virtual channels that are controlled by a virtual channel allocator unit. The routing computational unit determines the output port that the input data should be routed. The arbiter (inside switch allocator unit) arbitrates between different packets that try to access to a certain output port. Once the arbiter determines the winner, the corresponding input port will be granted access to that output port. The packets are transmitted to the output ports through crossbar switch that is controlled by switch allocator unit.

We modified the conventional router architecture explained above to provide hardware support for our proposed fault tolerant link. We integrated the fault detection mechanism [11] into routers that will inform any wire failure in the corresponding link. Initially when the system is fault free the router will work like a regular router. Once we detect a faulty link, the system will instantiate a splitter (also integrated in the router) that will break down the message/data (to be transmitted) in to smaller packets and use the remaining healthy wires to transmit the packets. The receiver side is also enhanced by integrating an assembler that will rebuild the original massage from the received packets.

4 Experimental Results

The proposed structure for the reconfigurable and fault tolerant NoC link is designed using Verilog HDL. The Xilinx ISE design tool [12] having a Spartan 6 FPGA as target device has been employed to synthesize and implement the proposed system. To validate the performance of the proposed link, we implemented a 4x4 regular NoC. In our implemented system and in our simulations, links were 64 bits wide; we assumed that we are after a system that can tolerate up to 8 failed wires per link.

Our implementation results shows that the area cost of out proposed system compared to a system that is not armed with any fault tolerant mechanism is 10%. Also our simulation and implementation results show that compared to the system that is not armed with any fault tolerant mechanism our proposed system has 15% higher power consumption.

5 Conclusion

In this paper, we proposed a new reconfigurable and fault tolerant structure for NoC links. The proposed structure is capable of maintain connectivity between the upstream and downstream routers even when the link connecting them fails. This is achieved by using the remaining functional wires in the broken link. In the transmitter side we design and integrate a splitter circuit that breaks down the message to be transmitted into smaller packets. Then the remaining functional wires are utilized to transmit the packets. In the receiver side we also design and integrate an assembler that rebuilds the original massage from the received packets. The main benefit of the proposed link structure is that it works in link level and therefore compared to dynamic routing algorithms does not create any deadlock or livelock problems.

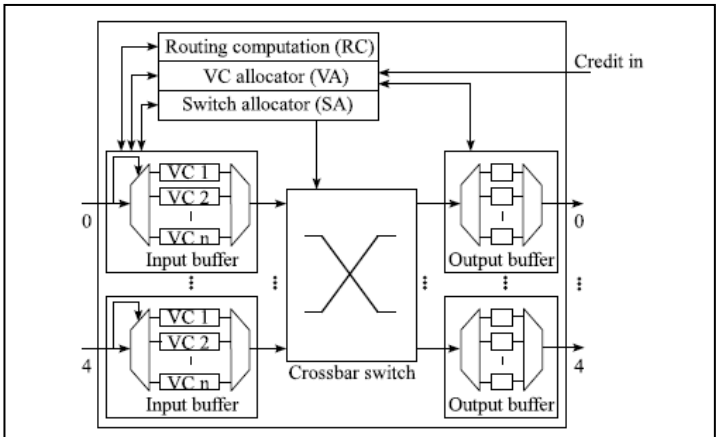

Fig. 3. Conventional router architecture.

References

[1] W. J. Dally, and B. Towles, "Principles and Practices of Interconnection Networks," Morgan Kaufmann, 2004.
[2] G.D. Micheli, and L. Benini, "Networks on Chips: Technology and Tools," Morgan Kaufmann, 2006.

International Journal of Scientific & Engineering Research Volume 5, Issue 7, July-2014
ISSN 2229-5518

[3] M. Agarwal, R. Dubey, N. Jain, and D. Raghuvanshi, "Comparative analysis of different topologies based on Network-on-Chip architectures," International Journal of Electronics and Communication Engineering, vol. 6, no. 1, pp. 29-40, 2013.
[4] R. Ho, K.W. Mai, and M.A. Horowitz, "The future of wires, " Proceedings of the IEEE, vol. 89, no. 4, pp. 490-504, 2001.
[5] H.S. Kia, C. Ababei, "A New Reliability Evaluation Methodology With Application to Lifetime Oriented Circuit Design," IEEE Transactions on Device and Materials Reliability, vol. 13, no. 1, pp. 192-202, 2013.
[6] F. Ge, N. Wu, and Y. Wan, "A Network Monitor based Dynamic Routing Scheme for Network on Chip," Asia Pacific Conference on Postgraduate Research in Microelectronics & Electronics, 2009.
[7] M. Ali, M. Welzl, and S. Hellebrand, "A Dynamic Routing Mechanism for Network on chip," NORCHIP Conference, 2005.
[8] H.S. Kia and C. Ababei, "A new fault-tolerant and congestionaware adaptive routing algorithm for regular Networks-on-Chi," IEEE Congress on Evolutionary Computation (CEC), 2011.
[9] M. Valinataj, P. Liljeberg, and J. Plosila, "Enhanced fault-tolerant network-on-chip architecture using hierarchical agents," IEEE Int. SymposiumSymposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2013.
[10] H.S. Kia, C. Ababei, "A new fault-tolerant and congestion-aware adaptive routing algorithm for regular Networks-on-Chip," IEEE Congress on Evolutionary Computation (CEC), 2011.
[11] H.S. Kia, C. Ababei, "Improving Fault Tolerance of Network-on-Chip Links via Minimal Redundancy and Reconfiguration," International Conference on Reconfigurable Computing and FPGAs (ReConFig), 2011.
[12] Xilinx ISE Tools, http://www.xilinx.com.