Fault Tolerant Network-on-Chip with Priority Based Arbiter

Priyanka V S
PG Student
Department of Electronics & Communication Engineering
Saintgits College of Engineering

Binu K. Mathew
Associate Professor
Department of Electronics & Communication Engineering
Saintgits College of Engineering

Abstract

The proposed NoC is a fault tolerant network-on-chip with a priority based arbiter. It is based on new error detection mechanisms suitable for dynamic NoCs and here the number and position of processor elements or faulty blocks vary during runtime. Also it provides online detection of data packet and adaptive routing algorithm errors. These mechanisms are able to distinguish permanent and transient errors and also it can localize accurately the position of the faulty blocks in the NoC routers. Also an arbiter is used here for managing multiple data's simultaneously.

Keywords: Adaptive Algorithm, Dynamic Reconfiguration, Network-On-Chip (Noc), Reliability, System-On-Chip (Soc).

I. INTRODUCTION

Due to technology scaling SoC systems are allowed to grow continuously in component count and also the complexity is increased. The commonly used shared-bus on-chip interconnect was a performance bottleneck for large systems due to global wiring delays. The Network-on-Chip (NoC) has therefore introduced as a means of providing scalable on-chip interconnects for SoC designs. Recently the trend of embedded systems has been moving toward multiprocessor systems-on-chip (MPSoCs) in order to meet the requirements of real-time applications. The complexity of these SoCs is increasing and the communication medium is becoming a major issue of the MPSoC.

The NoC relies on data packet exchange. The path for a data packet between a source and a destination through the routers is defined by the routing algorithm. NoC design adopts network-like communication that incorporates within the same chip a data-routing network consisting of communication links and routing nodes to provide a shared, segmented global communication structure. The required wiring is shortened as wires that form the communication links only have to span the local distance between routing nodes instead of global distances of an entire system. Therefore, such a data-routing network scales well with chip size and complexity, overcoming the previously mentioned limitation of complex global wiring. The NoC has many advantage over the traditional bus-based interconnect because of its features such as layered and scalable architecture, flexibility in network topology, and the decoupling of computation and communication. NoC design shares similarities with conventional off-chip networks such as those utilized by large-scale multiprocessors, distributed computing, and local or wide-area communication networks.

In the paper, a new reliable dynamic NoC is presented. The proposed NoC is having mesh topology structure of routers able to detect routing errors for adaptive routing based on the XY algorithm. The approach includes data packet error detection and correction. The originality of the proposed architecture is its ability to localize accurately error sources, allowing the throughput and network load of the NoC to be maintained. The considered routing algorithm is based on the adaptive turn model routing scheme and the well-known XY algorithm. This adaptive algorithm is live lock- and deadlock-free and allows data packets to pass around faulty regions.

II. LITERATURE REVIEW

A. Network-On-Chip Design:

Network-on-chip (NoC) design entails the provision of dedicated intra chip communication support that is similar to the more general and conventional off-chip networks that have evolved over several decades. The NoC approach is increasingly being adopted for system on- chip (SoC) design because of its ability to overcome the limitations of conventional bus-based communication. SoC systems that demand NoC support range from general purpose designs, application/platform-specific very large-scale integration (VLSI) designs, to flexible programmable-logic-based designs such as those in field-programmable gate arrays (FPGAs). The underlying architecture for providing NoC support for SoC design can vary depending on the application domain of the specific SoC platform.

B. NoC Architecture:

The overall architecture is in many aspects similar to large-scale networks, where its fundamental components consist of network interfaces, routing nodes, and communication links. On chip global network communication is supported by a set of interconnected routing nodes that are spread across the chip. Associated with each routing node is a network interface by which node elements (IP blocks) are connected, enabling access to the on-chip network. A NoC architecture is flexible so that it can support different network topologies based on system application requirements. The flexibility is reflected by the general network structure that is shown in Figure 2.1.

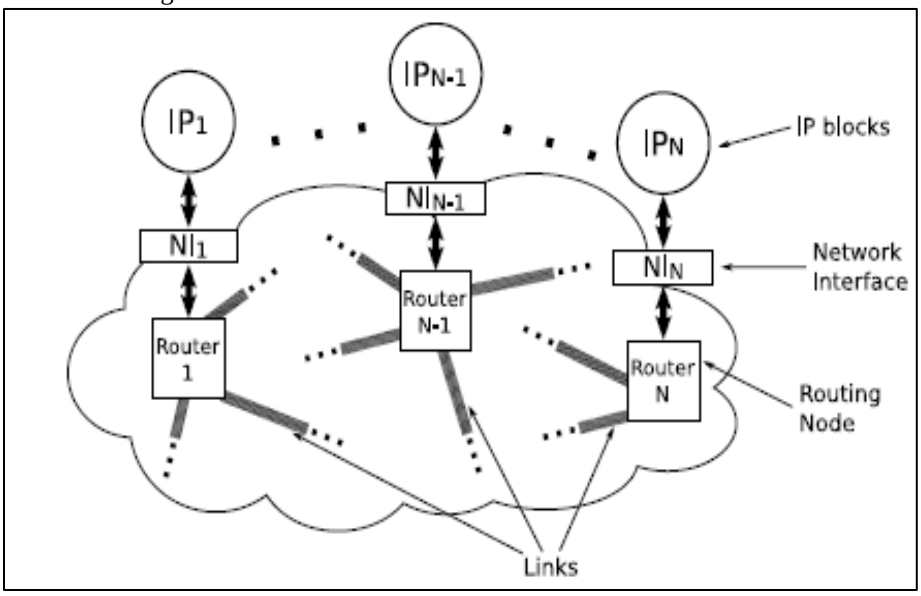

Fig. 1: Generic Noc Architecture

The role of a switch in a router is to connect the input buffers to the output buffers, as depicted in Figure 2.2. In most router designs, a crossbar switch is used to provide full connectivity between all of the available links.

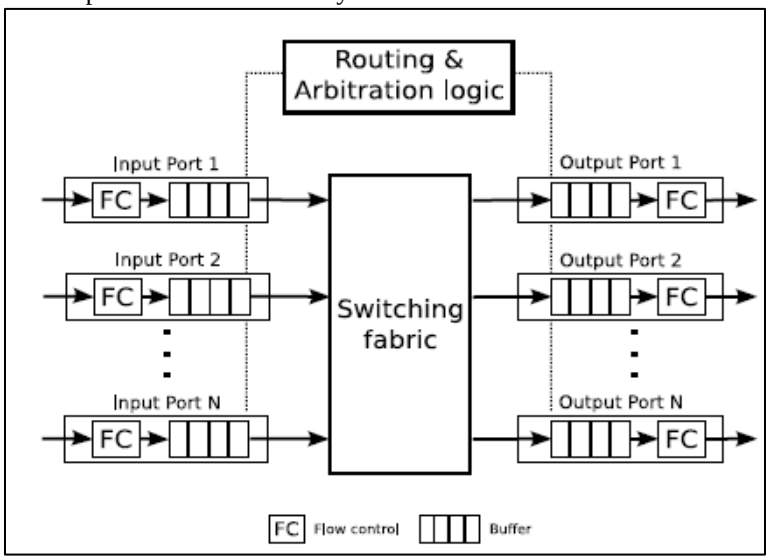

Fig. 2: Generic On-Chip Architecture

The switching technique defines how data flows through a switch in a router. There are two basic techniques: circuit switching and packet switching. In circuit switching, a fixed circuit is established between the sender and the receiver, and this direct connection is maintained for as long as it is needed. Although circuit switching can potentially ensure that the available bandwidth is fully utilized when the volume of communication is high, there is a high initial latency associated with the set-up of the direct circuit. Therefore, circuit switching is appropriate when communication patterns are well-understood, as in the case of application-specific SoCs and platform-specific SoCs.

Packet switching, on the other hand, transfers data by segmenting longer messages into smaller data packets, and forwarding these packets individually from the sender to the receiver, on a per-hop basis, possibly with different routes and delays for each

packet. Link reservation is not required, thus the network can support a large number of concurrent transmissions. Packet switching offers the potential for scalability and higher link utilization for general or unknown communication patterns.

C. Network Topology:

The topology of a network refers to the layout of components within a network, where it is concerned with the mapping of routing nodes and interconnects links. Topologies used for on-chip networks are adopted from large-scale networks and parallel computing (J. Kim et.al, (2006)) .Two-dimensional topologies are commonly adopted by on-chip networks because they can easily be mapped to the planar nature of a chip. The deciding factor is the system's required level of connectivity that is specified by the intended application. A few important NoC topologies are discussed below.

The mesh is a two-dimensional grid topology favoured in NoCs because of its regularity and linear area growth with the number of nodes. Meshes have a relatively large average network distance, which can increase power consumption. 2D-array is a type of mesh in which nodes form a two dimensional grid where each node is connected to the four adjacent routers. The routers at the edges have only two or three connections since they don't have more adjacent routers. Torus is a topology, which is similar to the 2D-array in which nodes form a regular cyclic 2-dimensional grid (W.J. Dally and B. Towels, (2001)). Here all routers have four connections since a torus basically is a mesh with wrap-around on the edges. Ring topology when the resources are connected to each other in a ring. Every resource is then connected to its two neighbour's communication with other resources then has to pass through the neighbours. Bus topology means that several resources use the same communication channel. In an ordinary local area network this can results in collisions, caused by two resources sending a packet at the same time. The tree is a hierarchical topology that begins with a top-level root router node with connections to one or more router nodes (known as children) at a lower level.

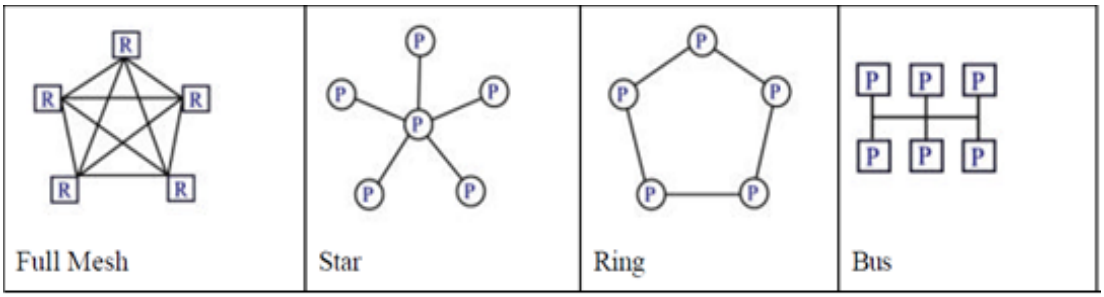

Fig. 3: Network Topologies

D. Routing Schemes:

Routing schemes can be classified as either deterministic or adaptive. Deterministic routing schemes have a predetermined traversal path between a given source and destination pair, independent of the current network status. The predetermined path allows packets to arrive in order, avoiding the overhead of reordering packets at the receiver. A popular deterministic routing scheme for meshes and tori is XY routing, which is simple and robust against deadlock (J. Duato et.al, (2003)). In XY routing, a packet is routed along a row first, then routed along the appropriate column to the destination.

In an adaptive routing scheme, routing decisions are made on a per-hop basis at each routing node, with consideration of current network status in the hope of avoiding congestion and balancing network load. Consequently, adaptive routing is more complex for implementation in order to offer flexibility in load balancing.

Routing schemes can be further distinguished as distributed routing or source routing. Distributed routing derives routing decisions locally at each router from local lookup tables or hardware routing functions. Hence, a packet only carries the destination address in its header, reducing the header overhead in a packet. In contrast, source routing derives its routing decision from a pre-computed global routing table, such that all the necessary routing decisions are embedded in the header to guide a packet to the receiving node.

E. Functionality of Router:

Router plays an important role in networks. Its main purpose is to receive packets from the sender, take decision by executing an algorithm and send the packet towards the destination. While going towards the destination, a packet may have to pass from several routers. Design of a router is very much depends on the routing algorithm used. A suitable routing algorithm makes the router less complex and faster. A NoC based router has the same functionality. Router architecture has three main components.

Input buffer is the temporary storage component. Main purpose of input buffer is to temporarily store the incoming packets if the router is busy or desire output is not empty. Input buffer is controlled by the Arbiter and Control block.

Cross Bar facilitates to connect an Input Port to any output port. Cross Bar is a combination logic which makes switching between different ports by receiving signals from Arbiter and Control block.

Arbiter and Control is the most important component of the router. It behaves like its brain. All routing decisions are taken inside this component. It resolves conflicts for simultaneous requests to an output port. It also organizes the flow of packets through the router. This component is actually the implementation of routing algorithm. It controls the input buffer by sending some signals to it. When a desired output is empty then it selects the desired output by sending the selection signals to the Cross Bar and sends a control signal to the input buffer to pass through the cross bar to the desire output.

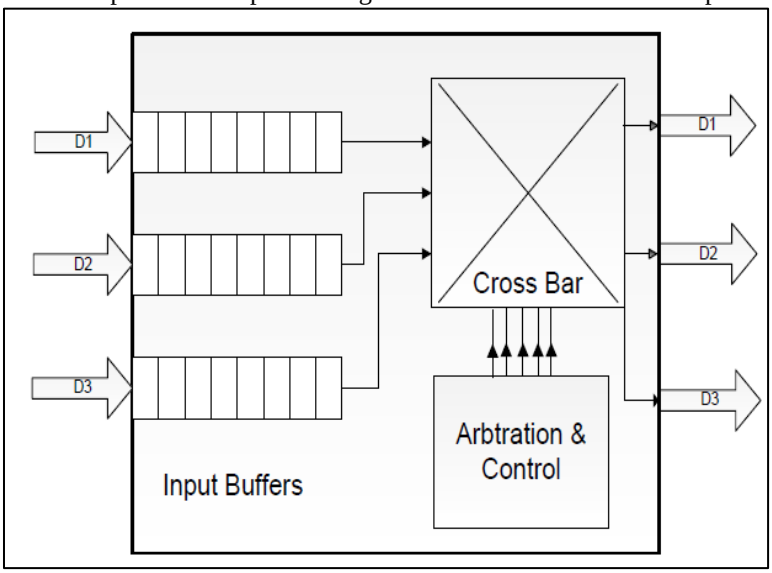

Fig. 4: Router Architecture

F. XY Routing Algorithms:

The XY routing algorithm is one kind of distributed deterministic routing algorithms. For a 2-Dimesion mesh topology NoC, each router can be identified by its coordinate (x, y). The XY routing algorithm compares the current router address (Cx, Cy) to the destination router address (Dx, Dy) of the packet, stored in the header flit. Flits must be routed to the core port of the router when the (Cx, Cy) address of the current router is equal to the (Dx, Dy) address. If this is not the case, the Dx address is firstly compared to the Cx (horizontal) address. Flits will be routed to the East port when Cx<Dx, to West when Cx>Dx and if Cx=Dx the header flit is already horizontally aligned. If this last condition is true, the Dy (vertical) address is compared to the Cy address. Flits will be routed to South when Cy<Dy, to North when Cy>Dy. If the chosen port is busy, the header flit as well as all subsequent flits of this packet will be blocked. The routing request for this packet will remain active until a connection is established in some future execution of the procedure in this router.

III. ANALYTICAL PROCEDURES

A. RKT-Switch:

It is a NoC-based communication approach. The RKT-NoC is a packet switched network based on intelligent independent reliable routers called RKT-switches. The RKT-switch architecture is having four directions i.e. North, South, East and West suitable for a 2-D mesh NoC. The PEs and IPs can be connected directly to any of the four sides of a router. Therefore, there is no specific connection port for a PE or IP. The proposed detection mechanisms can also be applied to NoCs using five port routers with a local port dedicated to an IP. But it has a drawback that when the local port has a permanent error and the IP connected to it is lost or needs to be dynamically moved in the chip because of the dynamic partial reconfiguration. Now, for the four-port RKT NoC, an IP can replace several routers by having several input ports and hence be strongly connected in the network (C. Bobda, (2005)). Each port direction is composed of two unidirectional data buses that are input and output ports Each input port follows to a first-input, first-output (FIFO) (buffers) and a routing logic block. The RKT-switch operation is based up on the store-and-forward switching technique. This technique is more suitable for dynamically reconfigurable NoCs. In the store and- forward technique, each data packet is stored only in a single router at any instant. Therefore the router is only need to empty its buffers when a router wants to be reconfigured. But with the wormhole switching technique (W. Dally and C. Seitz, (1987)), a single data packet can be spread over several routers. Hence, the time required to clear all the routers containing partial packet data (flits) and to reconstruct these packets before performing a reconfiguration is more important. The RKT NoC uses non bouncing routers (J. Raik et.al, (2006)), because if a router is surrounded by three unavailable neighbours, then it also becomes unavailable. But, if a data packet is sent to a router surrounded by three unavailable nodes, the packet cannot be routed.

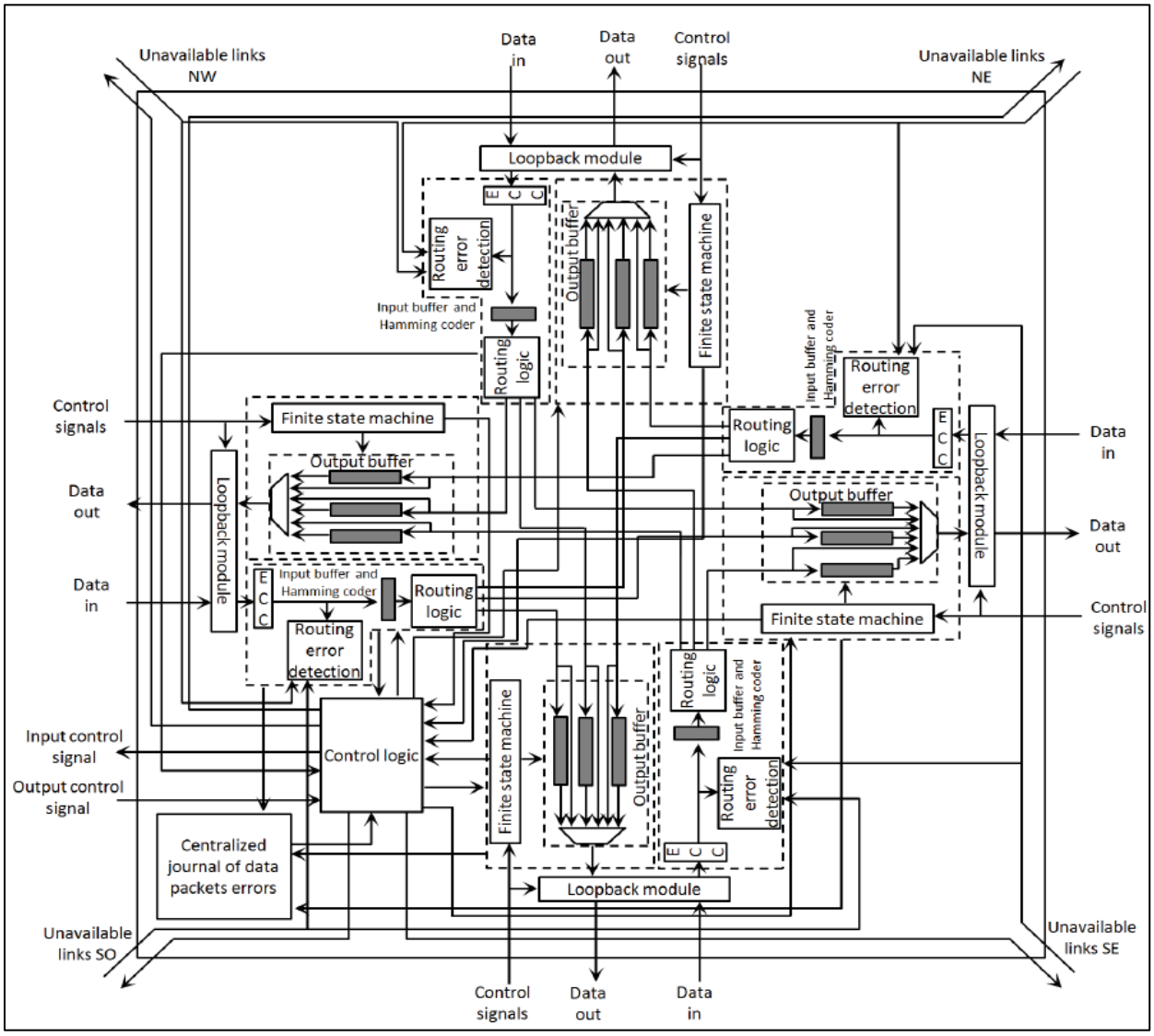

Fig. 5: Architecture of the Reliable Router RKT-Switch

The Hamming ECC is used in this RKT switch for error correction purpose .Hamming ECC is used here because it can provide a convenient trade-off between the error correction capacity and area overhead. This hamming ECC permits the correction of single event upset (SEU) errors, that means it corrects if there is only a single bit error in a flit and it also detects multiple event upset (MEU) errors ,that means it detects multiple bit error (two bit flips in a flit). On a single bit-flip error occurrence, error correction is possible with the Hamming ECC, whereas the single parity check would require packet retransmission and hence an increased transmission latency.

B. Routing Error Detection:

It is a reliable switch incorporating an online routing fault detection mechanism. This approach can operate with adaptive algorithms based on the well-known XY routing algorithm (G.M Chiu, (2000)), (S. Jovanovic et.al, (2009)), (W. Dally and C. Seitz, (1987)), (M. Majer et.al, (2005)). The main difficulty faced in routing error detection is to distinguish between the different types of errors. Main difficulty is to distinguish the bypass of an unavailable component in the NoC which occurs due to the use of the adaptive algorithm from a real routing error which causes due to a faulty component in the NoC. The consequence of the non-detection of routing errors is the loss of data packets that being sent either to an already detected faulty router or to an area performing a dynamic reconfiguration. The main elements required for routing error detection is diagonal availability indications, journal of routing error localizations and finally the structure of information fields in the data packets. The diagonal availability indication means the information links that are used to indicate its neighbour's availability status. There are eight direct neighbours for each RKT switch. It shares the diagonal availability indication links to these neighbours. If a router input is permanently faulty, then it will be disabled and make other ports active. So that we get a partially operating switch. If all the input ports are faulty, then the router will be considered as unavailable.

The basic concept of our approach is the following: Each router receiving a data packet checks the correctness of the routing decision made by the previous crossed switch. This routing error detection is performed in parallel after the Hamming ECC, as shown in Fig. 3.2. Consequently, this detection does not increase the data packet latency.

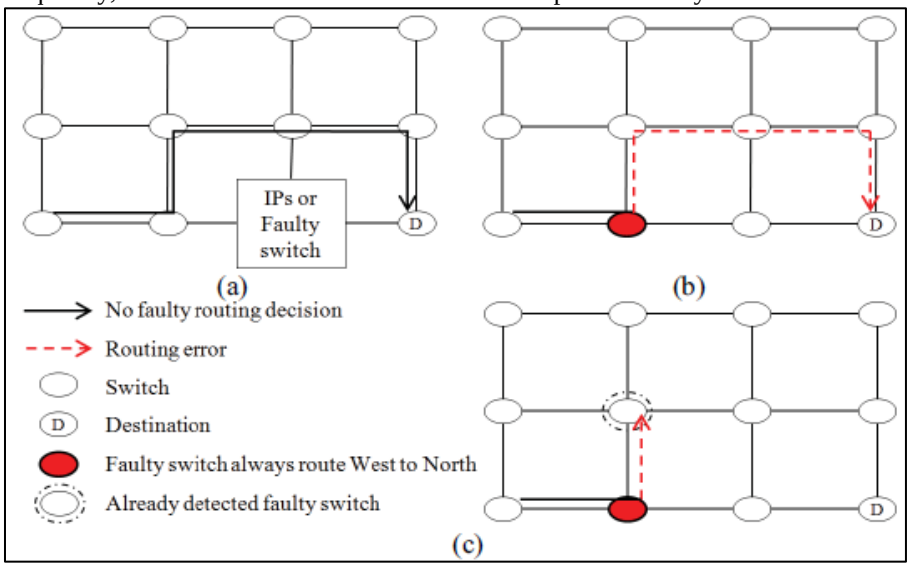

Fig. 6: Illustration Of The Routing Error Detection Problem (A) To Distinguish Dynamic Bypass (B) From A Routing Error And (C) To Avoid A Loss Of Data Packets

C. Localization of Data Packet Errors:

In order to locate and distinguish permanent and transient errors, a local history of data packet errors is stored locally in each router. This block is composed of journals contains the input and output ports. These journals are 3-bit-deep shift registers. The RKT-router uses the Ack/Nack data flow control signals for this task. Before transmitting a data packet to its neighbouring node, a copy of the same will be stored locally until an acknowledgement signal is received. A set of zero "0" will be added to the register if no error occurred during the transmission of data. If an error occurred during the transmission means retransmission of data is performed in response to a negative acknowledgement. If three negative acknowledgements are received, then the packet will be looped back and journal will be set to "1". The source of the error can be located anywhere on the input port, output port or on the data bus. The data packet will be checked by the input ECC after going through the loopback module. If no error is occurred means we can conclude that the error that is detected by the neighbour was occurred on the data bus. If the error occurred continuously for three times on the bus, then we can conclude that there is a permanent error on the data bus. The threshold is the number of consecutive errors required to flag an error source as permanent error. Here, we set a threshold of 3. When three consecutive errors occur on the same journal related to an input or output, the local historic of data packet errors concludes that a permanent error exists in the related direction.

The data packets after checked by the Error Correcting Code are being looped back and then checked by the routing error detection block. The faulty part of the NoC need to be isolated when a permanent fault is detected in a router. The part of the NoC which need to be isolated has been located accurately with the help of the local history. It can be located either in the input port, output port or in the data bus. The NoC –router activates the horizontal availability link of the faulty input port and two DAI links, if the error is located in the input port. Thus the neighbouring component connected to the faulty port cannot send any new data packets in this direction and DAI links helps to bypass the direction or position. The router detecting the permanent error must indicate to the neighbours to activate its availability indication link, if the error is on the data bus or in the output port.To indicate which port needs to be disconnected; the router detecting the permanent fault sends data packets to the destination of the neighbouring router. This one-flit data packet contains the address of the destination router and the direction of the port to be disconnected. However, the router must not send this special flit in the direction which was detected as faulty. As a result, the data packet is produced in the input port corresponding to the direction of the faulty neighbour. The routing logic block will then make a routing bypass and the packet will be sent to an available input of the faulty router.

IV. RESULTS AND DISCUSSION

Simulation refers to the verification of a design, its function and performance. It is process of applying stimuli to a model over time and producing corresponding responses from a model.

A. Simulation result of RKT Mesh:

The simulation result of a RKT switch mesh is shown in the figure 4.1.

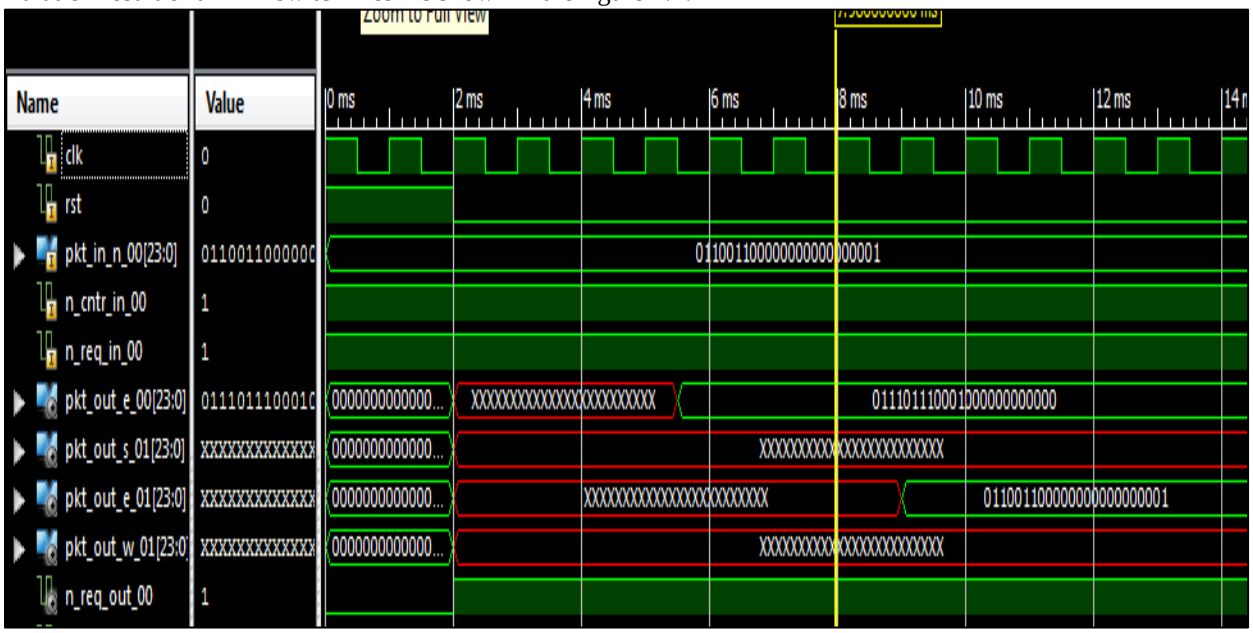

Fig. 7: Simulation Result of RKT Mesh

B. Simulation Result of RKT Mesh with an Arbiter:

An arbiter is added to the RKT switch in order to make it more efficient. It can help to manage simultaneous inputs at a time. The simulation result of a RKT switch with arbiter is shown below when more than one data is given as input at the same time.

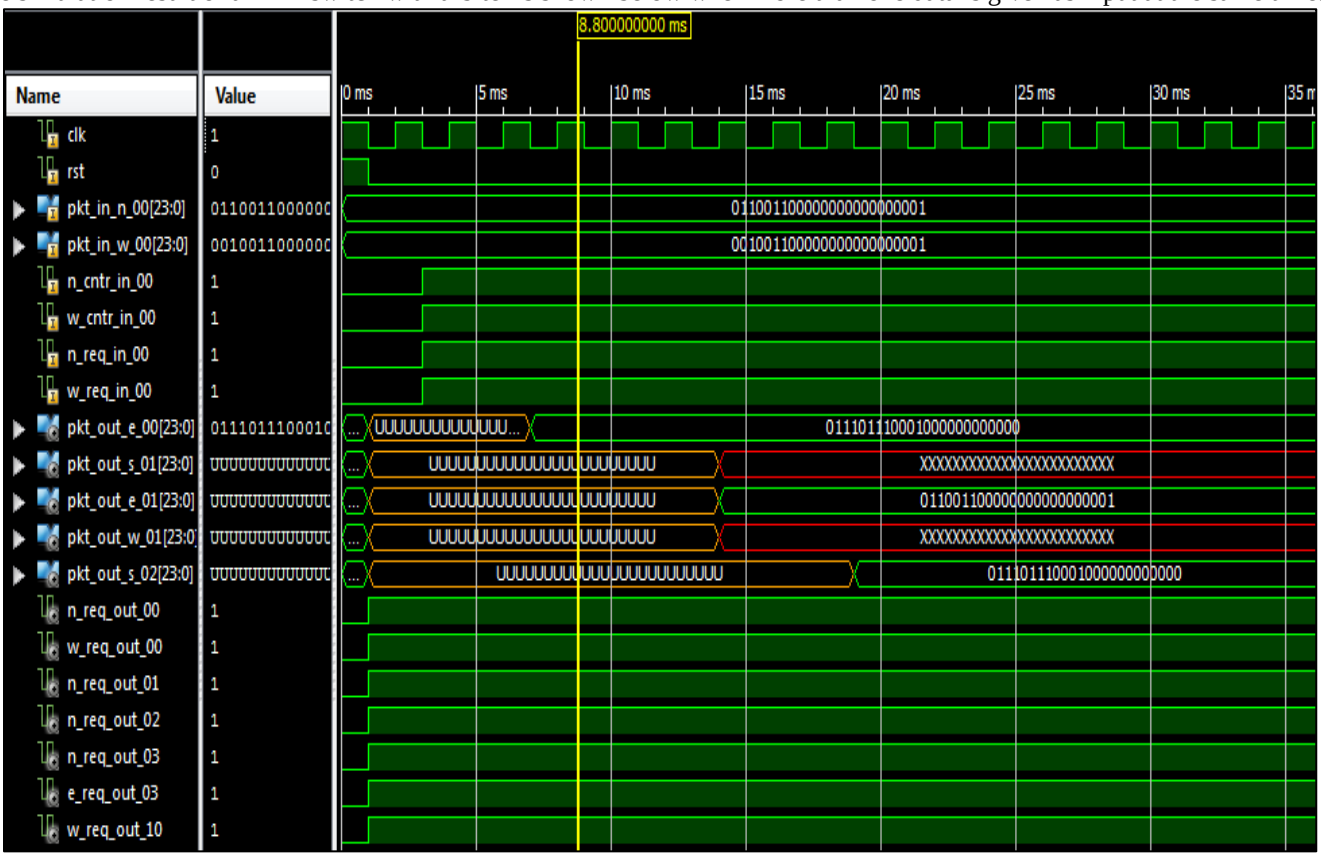

Fig. 8: Simulation Result of RKT Mesh with an Arbiter

C. Resource Utilization of Proposed System:

The resource utilization of a 4x4 RKT switch mesh is as given below:

top_mesh_4X4 Project Status (08/31/2015 - 14:47:17)			
Project File:	rkt_mesh.xise	**Parser Errors:**	No Errors
Module Name:	rkt	**Implementation State:**	Synthesized
Target Device:	xc3s500e-4fg320	**• Errors:**	
Product Version:	ISE 14.5	**• Warnings:**	
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	690	4656	14%
Number of Slice Flip Flops	486	9312	5%
Number of 4 input LUTs	1219	9312	13%
Number of bonded IOBs	229	232	98%
Number of GCLKs	12	24	50%

Fig. 9: Resource Utilization of Proposed System

V. CONCLUSION

The proposed routing error detection mechanisms finds the accurate localization of permanent faulty routing blocks in the network. They are suitable for adaptive routing algorithms based on XY algorithm .Here the main difficulty is to distinguish the bypasses of an unavailable component in the NoC which is due to the use of the adaptive algorithm from real routing errors which causes due to faulty components in the NoC. Regarding the proposed data packet error localization mechanisms, the simulations presented in this paper clearly show the efficiency of our techniques, which can localize permanent sources of errors more accurately than the switch-to-switch or code-disjoint mechanisms. Moreover, both presented techniques can distinguish permanent and transient errors, and show attractive performance as presented in the FPGA synthesis comparisons with a non-reliable NoC. The project focuses on evaluating accurately the impact of faulty detection blocks and improving the routing error detection mechanisms, by protecting the DAI links and routing detection blocks against errors.

REFERENCES

[1] Badri S. "Junction Based Routing: A Novel Technique for Large Network on Chip Platforms". Masters of Science thesis, Department of Electronics, School of Engineering, Jönköping University, Sweden. 2011.
[2] L. Benini and G. DeMicheli. "Networks on chips: a new SoC paradigm". Computer, 35(1):70–78, January 2002.
[3] C. Bobda, A. Ahmadinia, M. Majer, J. Teich, S. Fekete, and J. van der Veen, "DyNoC: A dynamic infrastructure for communication in dynamically reconfigurable devices," in Proc. Int. Conf. Field Program. Logic Appl., Aug. 2005, pp. 153–158.T.A.
[4] Y. M. Boura and C. R. Das, "Efficient fully adaptive wormhole routing in n-dimensional meshes," in Proc. 14th Int. Conf. Distrib. Comput. Syst., Jun. 1994, pp. 589–596
[5] G.-M. Chiu, "The odd-even turn model for adaptive routing," IEEE Trans. Parallel Distrib. Syst., vol. 11, no. 7, pp. 729–738, Jul. 2000.
[6] W.J. Dally and P. Song. "Design of self-timed VLSI multicomputer communication controller". In Proc. the International Conference on Computer Design, volume C-36 no. 5, pages 230–234, May 1987.
[7] W. Dally and C. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," IEEE Trans. Comput., vol. C-36, no. 5, pp. 547–553, May 1987.
[8] W.J. Dally and B. Towels." Route packets, not wires: On-chip interconnection networks". In Proc. 38th Design Automation Conference, pages 684–689, Las Vegas, NV, June, 2001
[9] J. Duato, S. Yalamanchili, and L. Ni." Interconnection Networks - An Engineering Approach". Morgan Kaufmann, 2003.
[10] A. Ejlali, B. Al-Hashimi, P. Rosinger, S. Miremadi, and L. Benini, "Performability/energy tradeoff in error-control schemes for on-chip networks," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 1, pp. 1–14, Jan. 2010.
[11] D. Fick, A. DeOrio, G. Chen, V. Bertacco, D. Sylvester, and D. Blaauw, "A highly resilient routing algorithm for fault-tolerant NoCs," in Proc. Design, Autom. Test. Eur. Conf. Exhibit., Apr. 2009, pp. 21–26
[12] A. P. Frantz, L. Carro, E. Cota, and F. L. Kastensmidt, "Evaluating SEU and crosstalk effects in network-on-chip routers," in Proc. 12th IEEE Int. Symp. On-Line Test., Jul. 2006, pp. 191–192
[13] R. Gindin, I. Cidon, and I. Keidar. "NoC-Based FPGA: Architecture and routing". In Proc. 2007 IEEE Int'l Symp. on Network-on-Chips, pages 253–264, Princeton, NJ, May, 2007.
[14] C. Grecu, A. Ivanov, R. Saleh, E. Sogomonyan, and P. Pande, "On-line fault detection and location for NoC interconnects," in Proc. 12th IEEE Int. On-Line Test. Symp., Jul. 2006, pp. 145–150.

[15] M. Hosseinabady, M. Kakoee, J. Mathew, and D. Pradhan, "Low latency and energy efficient scalable architecture for massive NoCs using generalized de Bruijn graph," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 8, pp. 1469–1480, Aug. 2011.
[16] S. Jovanovic, C. Tanougast, S. Weber, and C. Bobda, "A new deadlock-free fault-tolerant routing algorithm for NoC interconnections," in Proc. Int. Conf. Field Program. Logic Appl., Aug.–Sep. 2009, pp. 326–331.
[17] N. Karimi, A. Alaghi, M. Sedghi, and Z. Navabi, "Online networkon- chip switch fault detection and diagnosis using functional switch faults," J. Universal Comput. Sci., vol. 14, no. 22, pp. 3716–3736, 2008.
[18] J. Kim, C. Nicopoulos, and D. Park. "A gracefully degrading and energy efficient modular router arch. for on-chip networks". In Proc. 2006 IEEE Int'lSymp. on Computer Arch., pages 4–15, Boston, MA, June, 2006.
[19] M. Majer, C. Bobda, A. Ahmadinia, and J. Teich, "Packet routing in dynamically changing networks on chip," in Proc. 19th IEEE Int. Parallel Distrib. Process. Sym., Apr. 2005, p. 154b.
[20] D. Park, C. Nicopoulos, J. Kim, N. Vijaykrishnan, and C. Das, "Exploring fault-tolerant network-on-chip architectures," in Proc. Int. Conf. Depend. Syst. Netw., Jun. 2006, pp. 93–104.
[21] A. Pullini, A. Federico, D. Bertozzi, and L. Benini, "Fault tolerance overhead in network-on-chip flow control schemes," in Proc. 18th Symp. Integr. Circuits Syst. Design Conf., Sep. 2005, pp. 224–229.
[22] J. Raik, V. Govind, and R. Ubar, "An external test approach for network-on-a-chip switches," in Proc. 15th Asian Test Symp., Nov. 2006, pp. 437–442.
[23] K. Sekar, K. Lahiri, A. Raghunathan, and S. Dey, "Dynamically configurable bus topologies for high-performance on-chip communication," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 16, no. 10, pp. 1413–1426, Oct. 2008.
[24] J. Shen and P. Hsiung, Dynamic Reconfigurable Network-on-Chip Design: Innovations for Computational Processing and Communication, J. Shen and P. Hsiung, Eds. Hershey, PA, USA: IGI Global, 2010.
[25] J. Wu, "A fault-tolerant and deadlock-free routing protocol in 2d meshes based on odd-even turn model," IEEE Trans. Comput., vol. 52, no. 9, pp. 1154–1169, Sep. 2003.