PRODESIGN OF HIGH PERFORMANCE NOC ROUTER

Mitun S.

PG Scholar, Electronics & Communication Dept.
Hindusthan Institute of Technology, Coimbatore.

ABSTRACT

This is a high performance NoC Router that handles precise localizations of the faulty parts of the NoC. The proposed router is based on new error detection mechanisms suitable for dynamic NoCs, where the position of processor elements or faulty blocks varies during runtime. Indeed, I propose an online Error detection mechanism using CRC Algorithm. Proposed mechanism is able to discriminate permanent and transient errors and localize precisely the position of the faulty blocks in the NoC routers, while preserving the throughput, the network load, and the data packet latency.

KEYWORDS

Network on Chip, Loop Back Module, Error Detection Code, CRC Algorithm.

1. INTRODUCTION

As the volume and density of VLSI design increases, the intricacy of each component in a system increases rapidly. Recently the trend of embedded systems has been moving toward multiprocessor systems-on-chip in order to meet the requirements of real-time applications. Traditional bus-based communication methods are not able to keep up with the increasing requirements of future SoCs in terms of performance, power, timing closure, scalability, and so on. To satisfy the design productivity and signal integrity challenges of next-generation system designs, a structured and scalable interconnection architecture, Network on-Chip, has been proposed recently to reduce the complex on-chip communication problem.

Although NoCs can adopt concepts and methods from the well-established platforms of computer networking, it is impractical to blindly reuse features of existing computer networks and symmetric multiprocessors. In specific, NoC switches should be energy-efficient, small and fast. Neglecting these views along with proper comparison was typical for early NoC research but nowadays they are considered in more detail. The routing algorithms should be implemented by simple logic, and the number of data buffers should be less. Network properties and topology may be application-specific. NoCs should support quality of service (QoS), to achieve the various requirements in terms of end-to-end delays, throughput and deadlines. Real-time analysis, including video and audio playback, is one reason for providing QoS support. However, present system implementations like RTLinux, VxWorks or QNX are able to achieve sub-millisecond real-time analysis without special

hardware. This may point out that for many real-time applications the service quality of current on-chip interconnect infrastructure is adequate, and devoted hardware logic would be necessary to achieve microsecond accuracy, a degree that is rarely needed in practice for end users. Another inspiration for NoC-level quality-of-service is to support multiple simultaneous users sharing resources of a single chip multiprocessor in a public cloud computing foundation. In such occasions, hardware QOS logic enables the service provider to make agreement guarantees on the level of service that a user receives.

2. PROPOSED SYSTEM

To achieve a reconfigurable NoC, an efficient dynamic routing algorithm is required for the data packets. The goal is to maintain reliability and flexibility while providing high NoC performance in terms of throughput. Fig.1 illustrates a dynamic reliable NoC. Fig.1(a) shows the communications between several IPs and Fig.1(b) and (c) depicts the dynamic placement of an IP and the occurrence of a faulty node, respectively, both cases where bypasses determined by the dynamic routing algorithm are required. Furthermore, faulty nodes or even faulty regions make communications within the networks harder and even impossible with some routing algorithms, as shown in Fig. 1(c). Faulty nodes or regions dynamic component placement and are the main reasons why fault-tolerant or adaptive algorithms have been introduced and used in runtime dynamic NoCs.

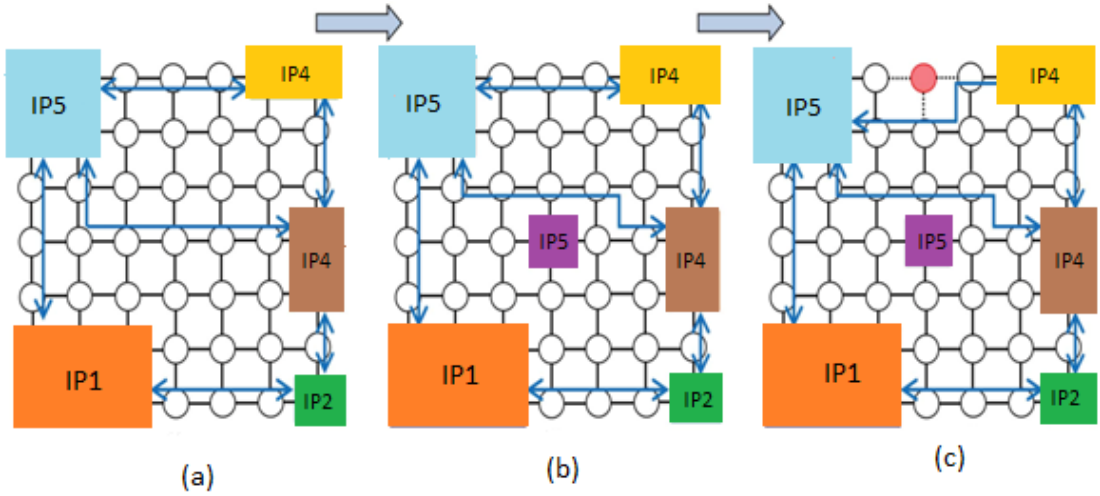

Figure.1.A Dynamic Reliable Noc Illustration.
(A) Normal Operation.
(B)Dynamic Implementation of An Ip.
(C) Online Detection of A Faulty Router.

2.1. Scenario

Network on chip is a prominent model for communications within large VLSI systems implemented on a single silicon chip. In a NoC system, modules such as memories, processor cores and specialized IP blocks exchange data using a network as a sub-system for the information traffic. A NoC is constructed from multiple point-to-point data chains interconnected by switches, such that messages can be relayed from any source module to any destination module over several links, by making

routing decisions at the switches. A NoC is similar to an advanced telecommunications network, using digital bit-packet switching over multiplexed links. Although packet-switching is sometimes demanded as necessity for a NoC, there are many NoC proposals exploiting circuit-switching techniques. This definition based on routers is normally interpreted so that a single crossbar switch, a single shared bus or a point-to-point network are not NoCs but practically all other topologies are. This is somewhat confusing since all above refered are networks (they enable communication between two or more devices) but they are not considered as network-on-chip approaches. Figure.2 shows a model of NoC.

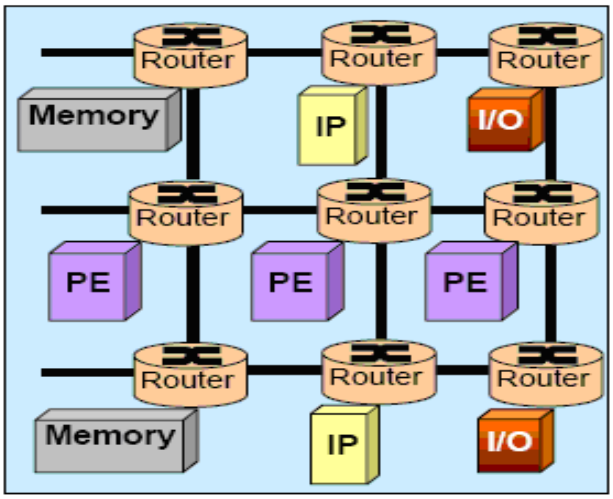

Figure.2 Model of NoC

2.2. Basic Concept of NoC Router

We propose a new reliable NoC-based communication approach called RKT-NoC. The RKT-NoC is a packet switched network based on intelligent independent reliable routers called RKT-switches. The architecture of the RKT switch is depicted in Fig.3. The RKT-switch is characterized by its architecture having four directions (North, South, East and West) suitable for a 2-D mesh NoC. The IPs and PEs can be directly connected to any side of a router. Therefore, there is no specified connection port for an IP or PE. The proposed detection mechanisms can also be applied to NoCs using five port routers with a local port dedicated to an IP. However, the major drawback of these architectures is 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. On the contrary, 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. Moreover, by using dynamic partial reconfiguration and IPs strongly connected in the NoC, no one fault location is more catastrophic than another. Indeed, an IP may have access to the network by being connected to several routers, or can be dynamically moved on the chip if this only access point becomes faulty. Each port direction is composed of two unidirectional data buses (input and output ports). Each input port is associated to a first-input, first-output (FIFO) (buffers) and a routing logic block.

The RKT-switch operation is based on the store-and-forward switching technique. This method is suitable for dynamically reconfigurable NoCs. Indeed, in our NoC, PEs and IPs can be implemented in place of one or several routers. At any instant with the store and- forward technique, each data packet is stored only in a single router. Hence, when a router needs to be reconfigured, the router is only required to empty its buffers. On the contrary, with the wormhole switching technique, a single data packet can be spread over several routers. As a result, 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, so that if a router is surrounded by three unavailable neighbors, it also becomes unavailable. In fact, if a data packet is sent to a router surrounded by three unavailable nodes, the packet cannot be routed.

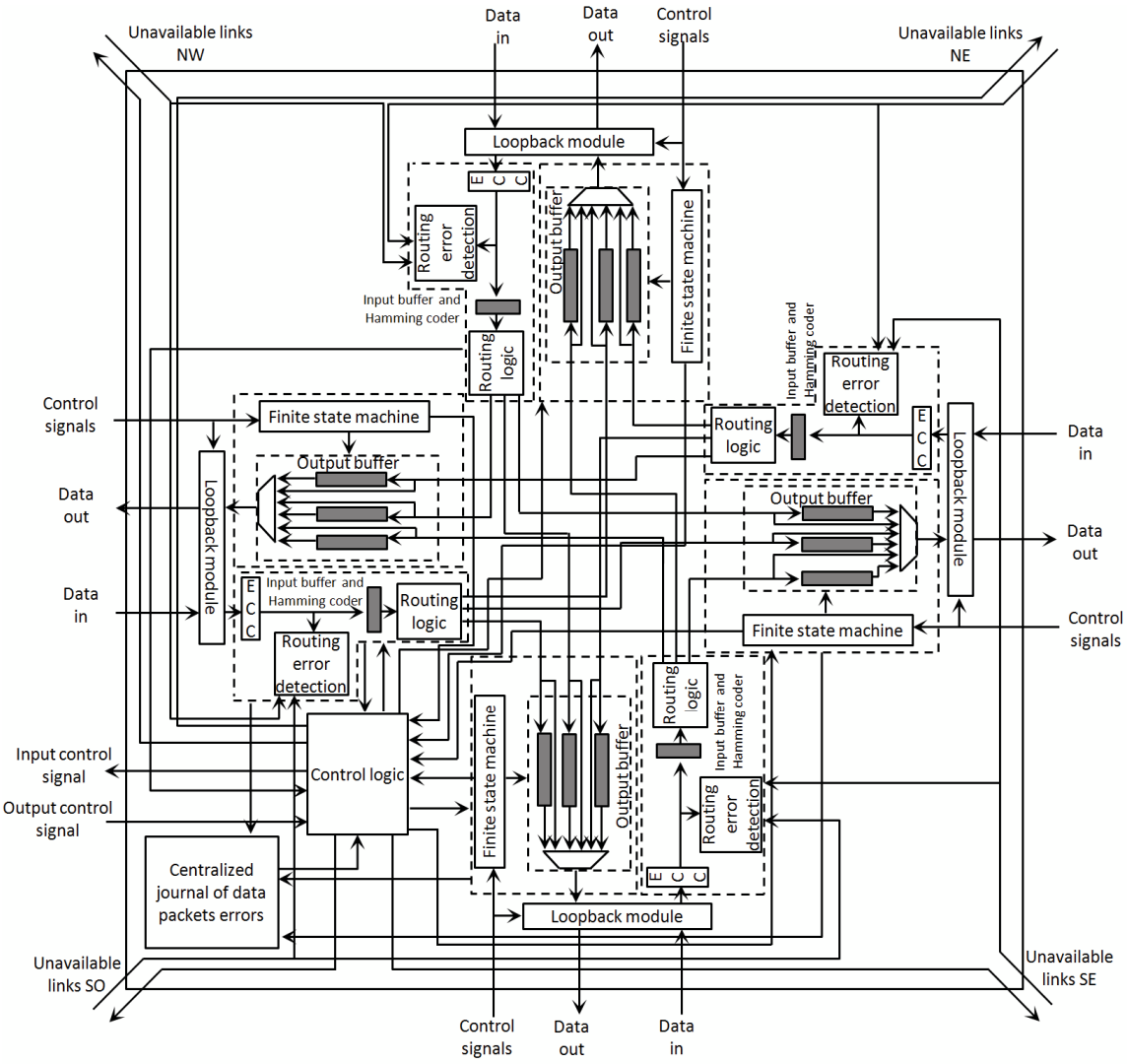

Figure.3. Architecture of RKT-Switch

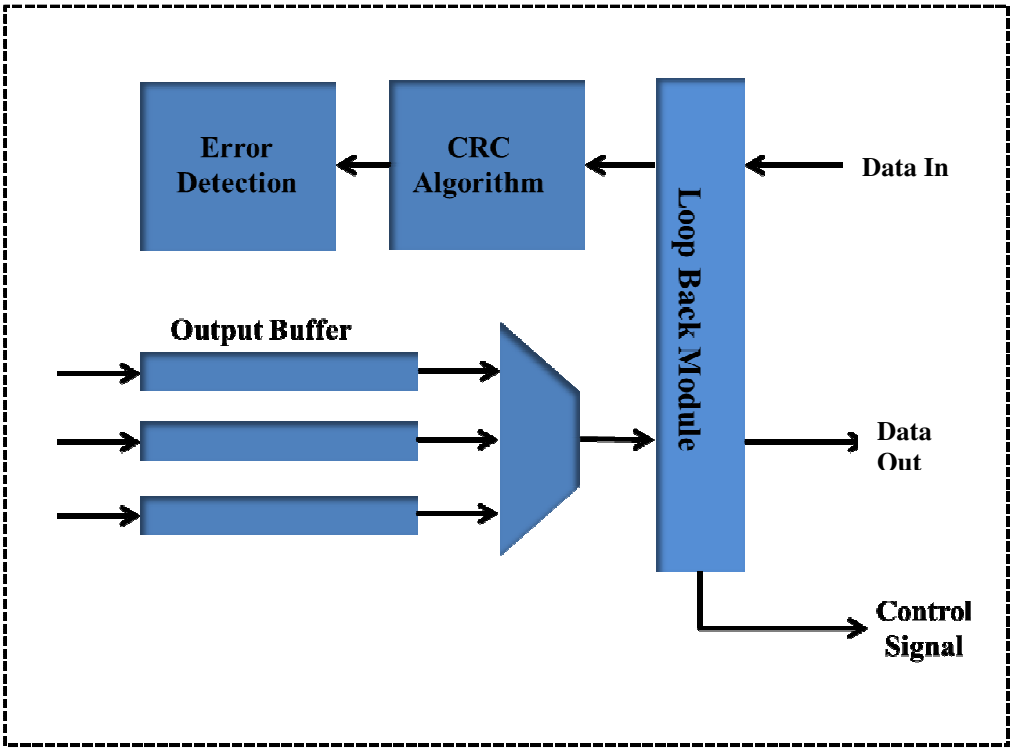

Figure.4. Block Diagram of Router

The data flow control used in our architecture is the Ack/Nack solution, which can handle fault-tolerant transmissions, although this does increase the energy consumption. This solution relies on the retransmission of packets being received as faulty by a neighboring node. Being able to perform a packet retransmission after it has been sent to a node requires that a copy of the packet be locally saved until an Ack or Nack is received. If a neighboring router receives a flit containing an error that cannot be corrected by the ECC, a Nack is sent back and the whole packet is retransmitted. Otherwise, at full packet reception an Ack is generated. More precisely, an Ack is generated only when all the flits of the data packet have been received and checked by the router, which reduces latency. The Hamming ECC is considered for our RKT-switch, in order to provide a convenient tradeoff between area overhead and error correction capacity. This choice permits the correction of single event upset (SEU) errors (one bit flip in a flit) and the detection of multiple event upset (MEU) errors (two bit flips in a flit). Moreover, the Hamming code is more suitable for NoCs based on Ack/Nack flow control than the parity bit check. Indeed, 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. The distinction between permanent and transient errors is granted thanks to a local historic, which saves the transmission results, and a loopback output mechanism. Moreover, the solution combined with the loopback mechanism and the original local historic allows the localization of errors, either on the bus connections or inside the switches, by localizing the faulty port.

2.3. Basic Principles of Loop Back Module

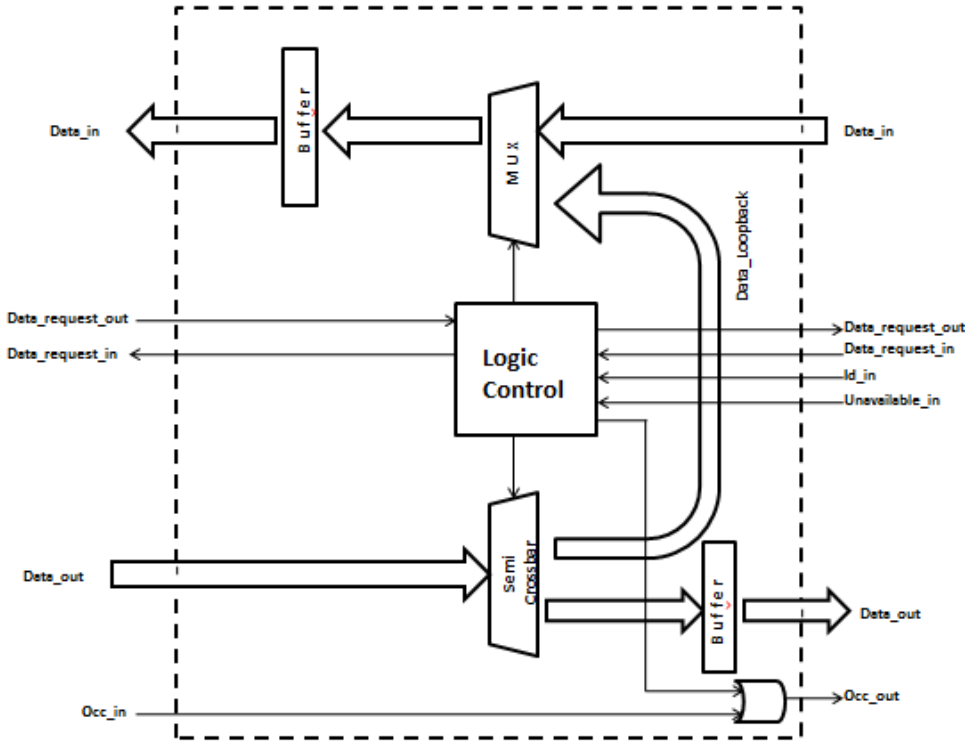

Figure.5. Block Diagram of Loop Back Module

In an active reconfigurable NoC, the number and the position of components in the network can change during operation, as illustrated in Fig.1. Actually, the position and number of the IP and PE in the NoC can be dynamically modified inorder to meet the requirements of the application. Partial reconfigurable regions (PRRs) must be defined inside the FPGA in order to achieve dynamic reconfiguration of the 2-D mesh NoC. These PRRs are the regions where partial reconfigurable modules (PRMs) can be implemented. PRMs represent electronic instantiations of functional units. They are defined by specific partial bit streams and can be placed according to the application needs. At present, these PRMs correspond to the IPs and PEs being placed and implemented inside the dynamic NoC, as shown in Fig.3. In a reliable NoC, faulty routers are isolated at runtime during the network operations. Let us consider a permanent faulty router that cannot be corrected. This router is permanently disabled. In the same way, during the reconfiguration of a PRR, no packet can be sent inside the area being reconfigured. Thus, these PRRs are dynamically isolated. However, these isolations can lead to data packet losses or increase packet transmission latency. More precisely, these drawbacks occur when routers containing data packets in their output buffers have their neighboring nodes unavailable due to a dynamic reconfiguration or permanent fault detection. Thereby, these data packets remain stored in the output routers until the end of the reconfiguration or are lost, in the case of detection of a permanent faulty node. To surmount these drawbacks, the proposed RKT-switch contains output buffer blocks associated with loopback modules, as portrayed in Fig.3. The role of each loopback module is to empty the buffers of each output port by looping

back the data packets in the input port of the router. The result is that the looped back packets are rerouted towards another output port of the router. This avoids data packets becoming trapped when a neighboring switch is detected as permanently faulty, and reduce latency when a neighbor has suffered a dynamic reconfiguration. Fig.6. illustrates the role of a loopback module. A PE or IP emitter sends data packets towards a destination IP according to the XY routing algorithm. If suddenly router (1, 3) becomes unavailable, the data packets remaining in the West output of router (2, 3) are looped back and rerouted towards its South output. This mechanism allows the stored data packets to be routed to the destination. Therefore, with router (1, 3) being indicated as unavailable, the subsequent data packets coming from the East input port are routed directly towards the South port by the dynamic routing algorithm. Furthermore, the main advantage of the combined use of the proposed loopback module, the local historic of data errors, and the switch-to-switch data error detection mechanism is the precise localization and distinction of the sources of data errors. Therefore, we can accurately locate whether the data errors are on the data bus, the input port, or output port, and whether the faults are permanent or transient.

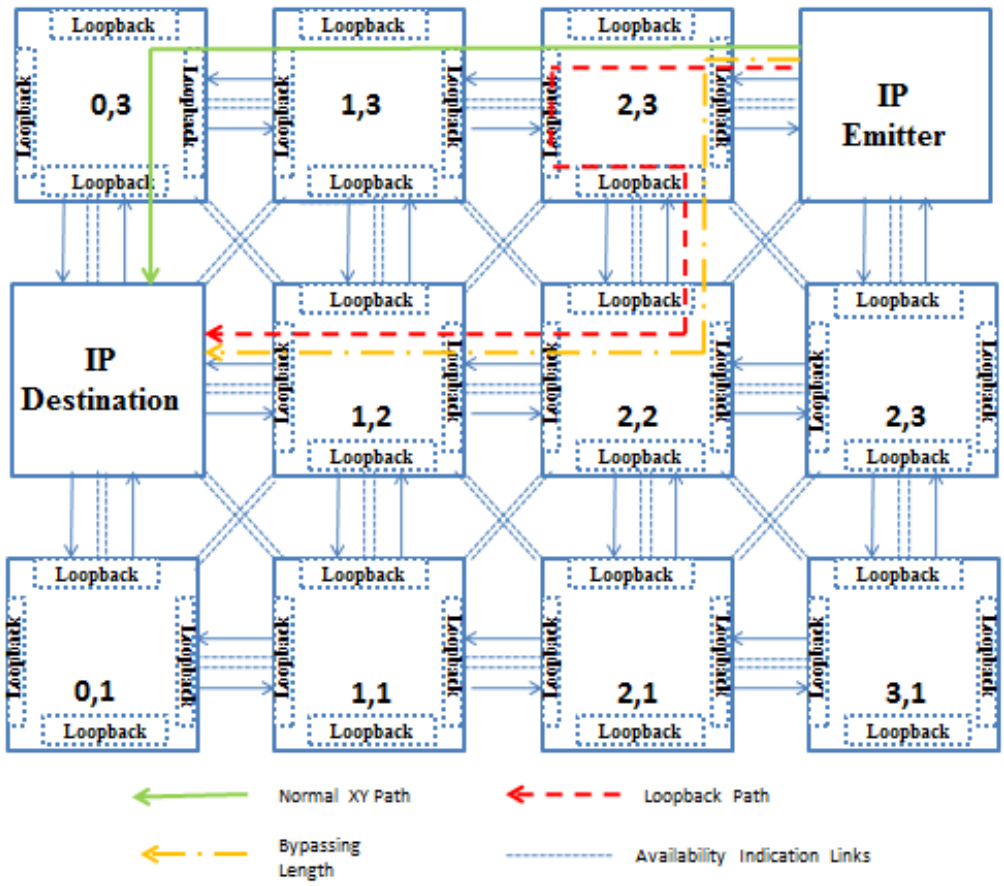

Figure.6. Illustrations of a Data Packet Loopback and A Dynamic Bypass Decision

2.4. Localization of Data Packet Errors

To locate and distinguish permanent and transient errors, a local historic of data packet errors is implemented locally in each router, as described in Fig.3. This block is composed of journals related to the input and output ports. These journals are 3-bit-deep shift registers. The RKT-router uses the Ack/Nack data flow control. When a data packet is transmitted to a neighboring node, a copy of the data is stored locally until the Ack is received. If no error occurred during the Transmission (reception of an Ack), a set to "0" is added to the register related to the input port of the get-in direction and the output port. If an uncorrectable error is detected by the neighbor, a retransmission is performed in response to a Nack. If three Nacks are received, the packet is looped back and a set to "1" is added into the journal related to the input and output ports taken by the data packet. Indeed, the error source can be located on the bus, in the input port, or in the Output port. After going through the loopback, the data packet is checked by the input ECC. If an uncorrectable error is detected by the ECC, the data packet is destroyed. If no error is detected, we can conclude that the errors detected by the neighbor occurred on the data bus. If the error occurred consecutively three times on the bus (i.e., three Nacks), we can conclude that there is a permanent error on the data bus. Table I shows the correlation between the data error detection results and the location of the errors (input block, output block, or data bus). The local historic has a threshold before disconnecting a part of a router. This threshold is the number of consecutive errors required to flag an error source as permanent. 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 being looped back, after being checked by the Error Correcting Code, are checked by the routing error detection block.

However, the routing error detection block finds in the SGD field that the previous router address is its own address and deduces a loopback. Consequently, it does not apply the routing error detection algorithm. When a permanent fault is detected in a router, the faulty part of the NoC has to be isolated. The part to be isolated has been located accurately by using the local historic and the loopback with the switch-to-switch error detection mechanism. It can be located in the input port, the output port, or the data bus. If the error is in the input port, the NoC-router activates the horizontal availability link of the faulty input port, and the two concomitant DAI links. In this way, the neighboring component connected to the faulty port cannot send new data packets in this direction, and the DAI flags indicate to the diagonal neighbors the possibility to bypass its position. If the error is on the data bus or in the output port, the router detecting the permanent error must indicate to the neighbor to activate its availability indications links. To indicate which port needs to be disconnected, the router detecting the permanent fault sends data packets to the destination of the neighboring router. This one-flit data packet contains the address of the destination router and the direction of the port to disconnect. However, the router must not send this special flit in the direction that was detected as faulty. As a result, the data packet is produced in the input port corresponding to the direction of the faulty neighbor. The routing logic block will then make a routing bypass and the packet will be sent to an available input of the faulty router.

Table.1. Localization of Errors

Results of the Data Transmission	Results of the Data Error Detection After Loopback	Input Port	Output Port	Data Bus
Ack	No loopback required	Not faulty	Not faulty	Not faulty
Three consecutive Nack	No error detected after loopback	Not faulty	Not faulty	Permanently faulty
Three consecutive Nack	Uncorrectable error detected after loopback	Suspect	Suspect	Suspect

2.5. Architecture of the Loopback Module

A loopback module is implemented in each of the four ports of the router, as illustrated in Fig.3. The architecture of the loopback module is depicted in Fig.5. The logic control block examine the accessibility of the neighboring router in order to transmit the data packets (data_request_in signal). If no loopback is demanded, a semi-crossbar connects the buffer to the data_out signal in order to send the data packets towards the neighboring router and activates the data_request_out signal. Then, a multiplexor connects the input data bus to the data_in bus. When a loopback is demanded, due to the inaccessibility of a neighboring router or an output block request occurrence after three Nack receptions, the logic control block configures the semi-crossbar block to send the assigned data packet on the data_loopback bus. Accordingly, the data packet is looped back inside the router and will be assigned as a new packet. During this step, in order to avoid the reception of a new data packet from the neighboring switch, the occ_out signal is activated. The loopback module requires one clock cycle to be crossed. Thereby, a data packet crossing a router has its latency increased by two clock cycles. Indeed, two loopback modules are crossed: one when arriving and one when leaving the switch.

3. RESULT

Experimentally obtained output are givrn below. It include the Empirical output, how much components are used, area used and the power status.

3.1. Empirical Output

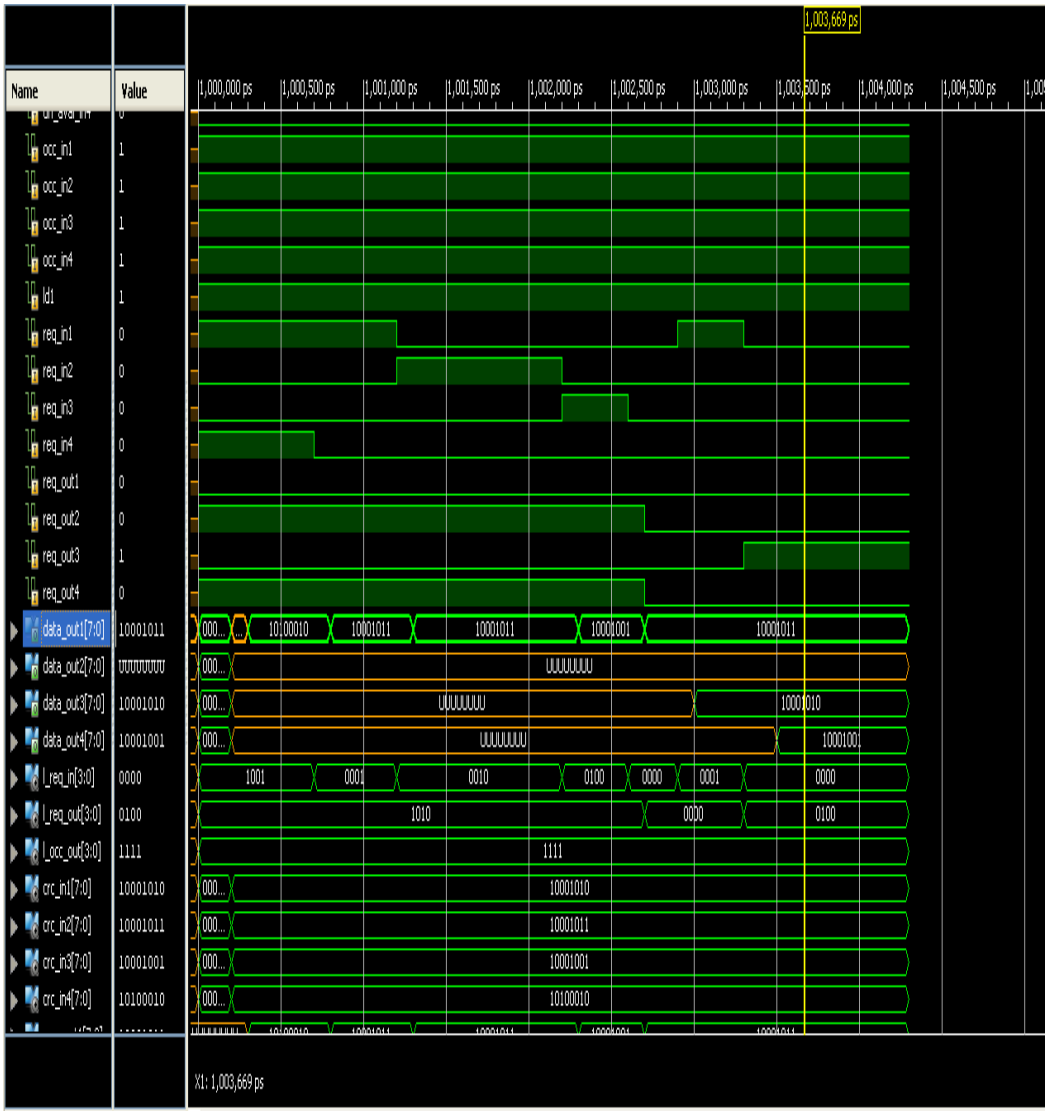

Figure.7. Simulation Result

3.2. Project Status and Area Used

output_buffer Project Status (10/10/2013 - 17:46:36)			
Project File:	Srnoc.xise	**Parser Errors:**	No Errors
Module Name:	rkt_switch	**Implementation State:**	Synthesized
Target Device:	xc3s250e-5pq208	**• Errors:**	No Errors
Product Version:	ISE 13.2	**• Warnings:**	547 Warnings (1 new)
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	157	2448	6%
Number of Slice Flip Flops	119	4896	2%
Number of 4 input LUTs	287	4896	5%
Number of bonded IOBs	78	158	49%
Number of GCLKs	5	24	20%

Figure.8. Project Status and Area Used

3.3. Power Status

Device	
Family	Spartan3e
Part	xc3s250e
Package	pq208
Grade	Commercial
Process	Typical
Speed Grade	-5

Environment	
Ambient Temp (C)	25.0
Use custom TJA?	No
Custom TJA (C/W)	NA
Airflow (LFM)	0

Characterization	
PRODUCTION	v1.2,06-23-09

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.000	9	...	...
Logic	0.000	282	4896	6
Signals	0.000	359	...	...
IOs	0.000	78	158	49
Leakage	0.052			
Total	0.052			

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	37.0	83.1	26.9

Supply Summary		Total	Dynamic	Quiescent
Source	Voltage	Current (A)	Current (A)	Current (A)
Vccint	1.200	0.015	0.000	0.015
Vccaux	2.500	0.012	0.000	0.012
Vcco25	2.500	0.002	0.000	0.002

	Total	Dynamic	Quiescent
Supply Power (W)	0.052	0.000	0.052

Figure.9. Power Status

4.CONCLUSIONS

The proposed routing error detection mechanisms allow the accurate localization of permanent faulty routing blocks in the network. They are suitable for adaptive routing algorithms based on XY where the main difficulty is to distinguish the bypasses of an unavailable component in the NoC (due to the use of the adaptive algorithm) from real routing errors (due to faulty components in the NoC). Validation simulations of our proposed routing error detection showed a routing error localization close to 96% for routing errors on an adaptive algorithm based on XY in a 6 × 6 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.

5.FUTURE WORKS

The faults during sharing of packets exceeding the routers efficiency can be more evaluated. This is explained through continuous fluid flow model. Network-on-chip designs are based on a compromise among latency, power dissipation, or energy, and the balance is usually defined at design time. However, setting all parameters, such as buffer size, at design time can cause either excessive power dissipation (originated by router underutilization), or a higher latency. The situation worsens whenever the application changes its communication pattern, e.g., a portable phone downloads a new service. Large buffer sizes can ensure performance during the execution of different applications, but unfortunately, these same buffers are mainly responsible for the router total power dissipation. Another aspect is that by sizing buffers for the worst case router, where the buffer slots are dynamically allocated to latency incurs extra dissipation for the mean case, which is much more frequent. In this paper we propose the use of reconfigurable increase router efficiency in an NoC, even under rather different communication loads. In the proposed architecture, the depth of each buffer word used in the input channels of the routers can be reconfigured at run time.

.

ACKNOWLEDGMENT

I would like to thank the Department of Electronics and Communication Engineering, HIT, Coimbatore for providing laboratory facilities and opportunity for experimental setup.

REFERENCES

[1] Cédric Killian, Camel Tanougast, Fabrice Monteiro, and Abbas Dandache ,2013 “Smart Reliable Network-on-Chip” Ieee Transactions On Very Large Scale Integration (VlSI) Systems.

[2] J. Shen and P. Hsiung, 2010, Dynamic Reconfigurable Network-on-Chip Design: Innovations for Computational Processing and Communication, J. Shen and P. Hsiung, Eds. Hershey, PA, USA: IGI Global.

[3] G.-M. Chiu, july 2000 “The odd-even turn model for adaptive routing,” IEEE Trans. Parallel Distrib. Syst., vol. 11, no. 7, pp. 729–738.

[4] Y. M. Boura and C. R. Das,june 1996, "Efficient fully adaptive wormhole routing in n-dimensional meshes," in Proc. 14th Int. Conf. Distrib. Comput. Syst., pp. 589

[5] C. Bobda, A. Ahmadinia, M. Majer, J. Teich, S. Fekete, and J. van der Veen,Aug 2005, "DyNoC: A dynamic infrastructure for communication in dynamically reconfigurable devices," in Proc. Int. Conf. Field Program. Logic Appl., pp. 153–158.

[6] S. Jovanovic, C. Tanougast, and S. Weber,july 2008 "A new high-performance scalable dynamic interconnection for fpga-based reconfigurable systems." in Proc. Int. Conf. Appl.-Specific Syst., Archit. Process., pp. 61–66.

[7] J. Wu, Sept 2003 "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.

[8] S. Jovanovic, C. Tanougast, S. Weber, and C. Bobda, Aug.–Sep. 2009 "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.

[9] W. Dally and C. Seitz, May 1987. "Deadlock-free message routing in multiprocessor interconnection networks," IEEE Trans. Comp.vol.C-36, no. 5, pp. 547–553.

[10] 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.

[11] 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.

[12] T. Pionteck, R. Koch, and C. Albrecht, "Applying partial reconfiguration to networks-on-chip," in Proc. Field Program. Logic Appl. Int. Conf.Aug. 2006,

[13] P. Lysaght and J. Dunlop, "Dynamic reconfiguration of FPGAs," in Proc. Int. Workshop Field Program. Logic Appl. More FPGAs. 1994, pp. 82–94.

[14] 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.

[15] 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.

[16] 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.

[17] C. Grecu, L. Anghel, P. Pande, A. Ivanov, and R. Saleh, "Essential faulttolerance metrics for NoC infrastructures," in Proc. Int. On-Line Test. Symp., 2007, pp. 37–42.

AUTHOR

Mr. Mitun S. Pursuing M.E. in VLSI Design and Embedded Systems, from Hindusthan Institute of Technology, Coimbatore under Anna University, Chennai. He Received B.Tech degree from Kannur university in Electronics and Communicatioin Engineering in 2012.
He is currently an intern in Nexegen Consultancy Services, Calicut.