Adding Fault-Tolerance to a Network-on-Chip

Thiago Felski Pereira[1], Cesar Albenes Zeferino[2]

Laboratory of Embedded and Distributed Systems
University of Vale do Itajaí
Itajaí, Brazil
[1]felski@univali.br, [2]zeferino@univali.br

***Abstract*— The constant reduction in the size of components of integrated circuits, as well as the growing operating frequency, increases the vulnerability to internal and external noise sources. These noises can cause a failure in any component, affecting the functioning of the system as a whole. Future Systems-on-Chip with dozens of cores will be based on Networks-on-Chip (NoCs), and will require networks that are able to detect a failure and avoid that this failure leads to a system failure and an application malfunction. In this context, this work aims at evaluating solutions to increase the reliability and availability of a SoCIN NoC, implementing mechanisms for error detection and correction on that network. The implemented mechanisms added a silicon overhead of 35,47% and a power dissipation overhead of 6,46% when compared to the original router.**

***Keywords*— System-on-Chip, Network-on-Chip, Fault-tolerance, Error model, Single-event Upset**

I. INTRODUCTION

New technologies of circuits manufacture have allowed the implementation of complete systems on a single chip, including processors, controllers, peripherals and memories. Such systems are called Systems-on-Chip (SoCs). The interconnection between the cores of a SoC is usually done through buses, because they are interconnection structures of easy reuse, reducing costs and design time. However, SoCs with several tens of cores demand architectures with better performance than those offered by buses. To meet the requirements of these SoCs, Networks-on-Chip (NoCs) [1] emerged as a solution to interconnect cores offering scalable performance and communication parallelism.

A NoC consists of routers, transferring packets between the cores, and of links that interconnect these routers and connect the cores to the network. The constant reduction in the size of the components and the increased operating frequency make the NoC (as well as other components of a SoC) increasingly susceptible to internal noise sources (power supply, crosstalk) and external (heat, electromagnetic interference induced by alpha particles) [2]. These noises can result in faults in components of the NoC, and a fault can produce an error, as, for example, the inversion of a bit. When there is a fault in a router or a network link, the routes based on the component that suffered the fault can be compromised, affecting the functioning of the system as a whole.

Several types of SoCs require that its components are tolerant to different types of faults, depending on the operating environment to which they are subjected. A fault-tolerant NoC should be able to detect a fault and prevent the resulting error to lead to a system failure and a malfunction of the application.

Providing reliability to a NoC implies direct consequences on the network performance, as well as in the silicon costs and in the power consumption of the application [2]. This increasing occurs because the provision of reliability is based on redundancy [2], which is done by replicating existing circuits and/or adding new circuits.

In this context, this article presents a low cost solution to protect the router of SoCIN*fp* NoC [3] ensuring the continuity of operation even in the presence of a SEU (single-event upset) transient fault in one of its registers, as well as protection of the routing bits of the packet header. This solution ensures the correct operation of the router without having to retransmit a packet when a SEU fault occurs.

This article is organized as follows. Section II describes the architecture of SoCIN*fp*. Section III describes how faults tolerance techniques have been applied in SoCIN network in previous works. Section IV presents and discusses experimental results. Finally, in Section V, the conclusions of the work are presented.

II. ARCHITECTURE OF SOCIN*FP*

SoCIN (System-on-Chip Interconnection Network) [3] is a parameterizable NoC developed to provide low-cost scalable communication for SoCs. This NoC uses a 2-D mesh topology and a basic building block, the RASoC router (Routing Architecture for SoC) [4], which is implemented as a synthesizable and parameterizable hardware block. The RASoC router parameterization allows the system designer to configure the width of the communication channels and the depth of its FIFO buffers to generate a network with the best cost and performance trade-off required by the target application. However, the communication mechanisms that define how data is transferred over the network are fixed (i.e., non-parameterizable). The RASoC router uses a Handshake flow control, FIFO buffers, XY deterministic routing, and Round-Robin arbitration. The network addressing is based on X and Y coordinates, and the packet header includes the amount of displacement to be made in each direction on the route between the source and the destination (X_{offset}, Y_{offset}). This information is updated at each router and analysed by the next router to determine the output port to forward the packet. When it equals (0, 0), the packet is delivered to the core attached to the router.

Aiming at giving greater flexibility to the network and expand its design workspace to find the trade-offs, in [3], a new router that extends the features of RASoC parameterization was developed. This new router was named ParIS (Parameterizable Interconnect Switch) and differs from RASoC by supporting the set-up of communication mechanisms, offering alternatives for flow control (Handshake or Credit-based), FIFO buffering (at the input and output channels), distributed routing (deterministic or partially adaptive) and arbitration (Round-Robin, static or random), and other techniques can be easily implemented. To support this degree of parameterization, modifications were made in the way that the SoCIN routing is done, resulting in changes in the network header. Instead of the distance (*offset*) until the destination node, the header includes its coordinates (X_{dest}, Y_{dest}). At each router, the routing algorithm compares their coordinates with the one of the destination node. If they are equal, the packet is delivered to the core attached to the router. Otherwise, it is forwarded to a neighbour router, using a port selected by the routing algorithm after comparing the coordinates. This new version of SoCIN network was called SoCIN*fp* (*fully parameterizable*) and the architecture of its router is described below.

A. Architecture of ParIS router

ParIS router has up to five bi-directional communication ports (L, N, E, S and W), each one composed of two opposite simplex channels: input (*in*) and output (*out*), as can be observed in Figure 1. Each channel consists of data (*din* or *dout*), data validation (*val*) and return (*ret*) signals.

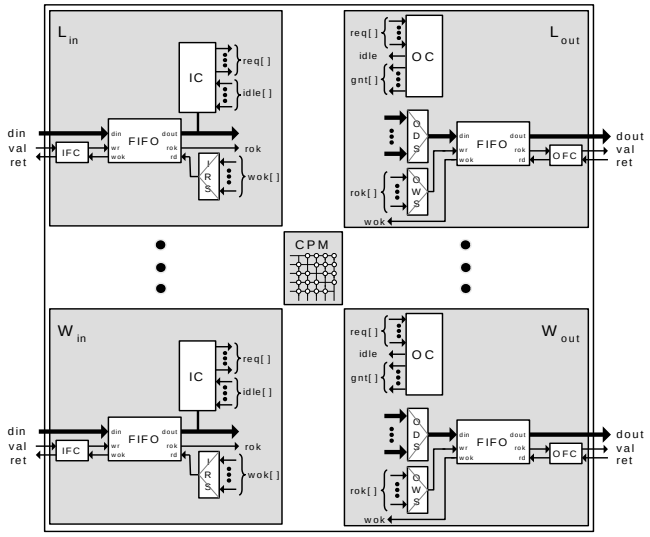

Figure 1. Block diagram of the ParIS router [3]

Each input channel is an instance of a module composed of blocks responsible for the following mechanisms: flow control (IFC – Input Flow Control), buffering (FIFO – First-In, First-Out), routing (IC – Input Controller) and switching (IRS – Input Read Switch).

The output channels are instances of a module composed of blocks responsible for the following mechanisms: switching (ODS – Output Data Switch and OWS – Output Write Switch), arbitration (OC – Output Controller), buffering (FIFO) and flow control (OFC – Output Flow Control).

This work aimed at providing fault-tolerance to ParIS router in order to ensure the correct operation of the routing, arbitration and switching mechanisms. The techniques were also applied to protect the IC and OC router blocks, as well the routing information of the packet header.

III. PREVIOUS WORKS

Some previous studies about fault-tolerance for SoCIN were carried out. In [5], techniques were implemented to protect the router of the original SoCIN (RASoC router) against SEU transient faults (soft-errors) and crosstalk. The Author proposed the use of a hybrid hardware and software approach to provide fault-tolerance to the routers. The solution consists of hardware modifications with the inclusion of a CRC in the packet trailer to detect errors in the packet payload and a bit of acknowledge/not acknowledge (Ack/Nack) signal to request retransmission. To solve the problem in routing, a triple sampling technique was used. The Author compared this solution with a fully hardware-based in which the links are protected by a triple sampling of the received. In this solution, FIFOs are protected by Hamming encoders/decoders and other components of the router (e.g. FSMs – Finite State Machines) are protected via TMR (Triple Modular Redundancy). Results showed that the hybrid solution offers the same error coverage rate of the solution entirely based on hardware and has a lower overhead in power and area.

In [6], fault-tolerance techniques were implemented based on redundancy of information (parity and CRC) to protect SoCIN links against crosstalk transient faults. The Authors have implemented a saboteur module responsible to inject crosstalk faults in a SystemC simulation model of SoCIN, and thus assessed the ability of the network to detect and recover from such failures. The Authors did not evaluate the impact of the used techniques in the cost and performance of the network.

In [7], the TMR technique is used to protect the state and controlling registers of routers against SEU faults and a selected crosstalk avoidance code (SCAC) was used to endure crosstalk effects and soft errors on communication channels and data buffers.

This work derives from the studies done in [5], however, applying a spatial redundancy technique in SoCIN*fp* in order to protect its router and arbitration blocks. The work differs also by applying a technique of information redundancy to protect the routing field (RIB) of the header. The implementation of these techniques in the ParIS router is described below.

A. Adding Spatial Redundancy

TMR spatial redundancy technique [5] was applied to protect the registers of the routing (IC) and arbitration (OC) blocks of ParIS router against SEU faults that could change the state of these registers, which could result in the

establishment of a wrong connection or in the cancellation of an existing connection. The first error would result from the setting of a bit in the register of IC block (generating a new request). The second error could be caused by resetting a bit in the register of the IC block (cancelling a request) or a bit in the register of the OC block (cancelling an already established connection).

The registers of these blocks were protected against SEU transient faults by applying the spatial redundancy TMR technique. This technique is based on the replication of the modules to be protected, by submitting its outputs to a voter module, which is responsible for defining a valid output to the other blocks of the router.

1) Spatial Redundancy in the IC Block:

Internally, the IC block consists of a combinational function that performs the routing algorithm and selects an output module. This selection is stored in a register (called *req_reg*) that maintains the state of all signs of request (one for each output module that can be requested). At any given moment, only one request can be set (ex: "0001"). In addition to the flip-flops, the register has a control logic that detects the presence of the header in the output of the FIFO and the time the packet trailer is forwarded. The first condition identifies the time that *req_reg* register should store the output of routing function (by setting up a request), whereas the second identifies the time that the register should be cleaned (resetting the request) and, therefore, cancelling the connection. A SEU type fault in a flip-flop of *req_reg* can activate or cancel a request, which can cause the system to fail, given that communications carried out by means of that input channel would be compromised.

The TMR solution adopted consisted in the instantiation of two additional copies of *req_reg* register and inclusion of a voter circuit. When a SEU fault occurs in a flip-flop from one of the three registers, the voter will select the state of the flip-flops of the same index of the other two registers and avoid the error. However, the error is propagated if the same fault occurs in more than one flip-flop of the same index of at least two registers. The structure of IC block with TMR is depicted by Fig. 2

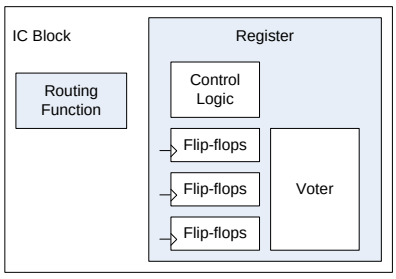

Figure 2. IC block with TMR

2) *Spatial Redundancy in the OC Block:*

Internally, the OC block consists of a priority generator and a programmable priority encoder. The first defines the priority criteria to be applied in an arbitration cycle, indicating which request will be the highest priority. The priority encoder implements a selection logic applying the current priority criteria to select one of the requests and grants the connection to the IC block with the highest priority.

Similarly to what was done in the OC block, all flip-flops were replicated and connected to a voter to protect the OC block against SEU faults, resulting in the structure depicted in Fig. 3.

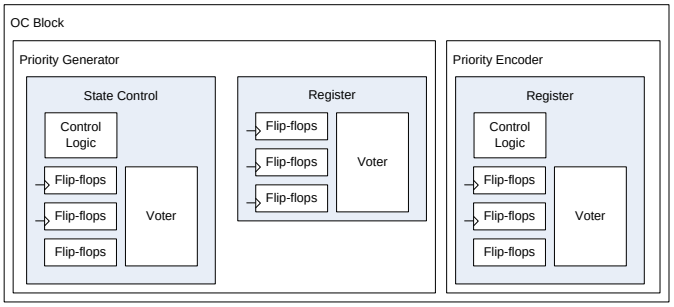

Figure 3. OC block with TMR

B. Adding Information Redundancy to the Packet Header

The work also proposed to protect the routing information contained in the packet header (RIB – Routing Information Bits), when it is stored in the FIFO, in order to avoid that a SEU fault in one of its bits (which could change its value) lead the packet to an incorrect route. The RIB field is replicated twice and the three instances are evaluated by a voter, which is responsible to generate a valid RIB when there is a single SEU fault in a given bit or multiple faults in different bits.

IV. Implementation and Results

In order to evaluate the impact of the proposed solutions, VHDL models were synthesized to a 90 nm ASIC technology (SAED) using Synopsys Design Compiler tool. The obtained metrics were: (*i*) the silicon area (expressed in μm^2); (*ii*) the maximum operating frequency (in MHz); and (*iii*) the total dissipated power (expressed in mW) for a given operating frequency.

Each implementation was compared with similar results of the ParIS router without fault-tolerance, using the same configuration of communication mechanisms, and operating at the same clock frequency. The configuration used on the router ParIS was as follows: (*i*) XY routing; (*ii*) Credit-based flow control; (*iii*) Round-Robin arbiter; (iv) 4-flit FIFO buffers at the input channels; (*iv*) 32-bit data; (*v*) 8-bit RIB; and (*vi*) 5-port router.

A. Silicon Area

Table I presents the silicon costs, including the following metrics for the target technology: (*i*) combinational area; (*ii*) non-combinational area; (*iii*) nets area; (*iv*) total area; and (*v*) overhead.

The redundancy of information implemented added a small extra cost in combinational and net area, without any non-combinational overhead, given that its implementation is purely combinational.

TABLE I. SILICON RESULTS

Implementation	Combinational area (µm²)	Non-combinational area (µm²)	Net area (µm²)	Total area (µm²)	Overhead (%)
Original	24,389.22	21,136.90	3,535.16	49,061.28	
Information redundancy	26,912.56	21,136.90	3,689.99	51,739.45	5.46
Spatial redundancy	32,263.37	26,915.33	4,475.15	63,653.85	29.74
Combined redundancy	34,900.99	26,915.33	4,645.38	66,461.70	35.47

The Spatial redundancy implementation presented overhead in all areas, but the combinational area had a more significant overhead than the others. The Combined (Information + Spatial) implementation presented the same increasing in non-combinational area (sequential), since the information redundancy technique is purely Combinatorial.

B. Power Dissipation

Table II presents the power results for the proposed solutions in comparison with the original implementation of ParIS router. It is observed that the Combined implementation presents a reduced power overhead (6.46%), while the area overhead was of 35.47%.

TABLE II. POWER DISSIPATION RESULTS

Implementation	Total power (µW)	Overhead (%)
Original	2,134.70	
Information redundancy	2,204.80	3.28
Spatial redundancy	2,224.30	4.20
Combined redundancy	2,272.50	6.46

V. CONCLUSIONS

In this paper, fault-tolerance techniques have been applied to protect ParIS router of SoCIN*fp* network against SEU transient faults. The protection of the routing and arbitration mechanisms was done by applying a spatial redundancy technique (TMR). The implementations consisted in the replication of the modules vulnerable to SEUS faults and in the addition of a voter module. As the technique of spatial redundancy does not protect a word whose header has already arrived with error, it was also adopted a technique of information redundancy to protect the routing header field (RIB).

The information redundancy technique applied in this study had a very small cost compared to the spatial redundancy technique. The overhead of the combination of the two techniques was of 35.47% in silicon area and 6.46% in power dissipation. This extra cost represents the impact required to ensure the operation continuity even in the occurrence of faults.

As future work, it is proposed to integrate the techniques used in this work with the techniques adopted by [6] to protect the network links so as to produce a more robust implementation of SoCIN*fp* in the presence of faults. It is also intended to add the fault-tolerance techniques in a SystemC simulation model of the SoCIN*fp* network in order to assess the impact of the fault-tolerance mechanisms in the network performance metrics (throughput and latency).

ACKNOWLEDGMENT

This work was supported by CNPq National Program of Microelectronics (Process No. 551483/2010-5).

REFERENCES

[1] JANTSCH, A.; TENHUNEN, H. (Eds.). Networks on Chip. Boston: Kluwer Academic Publishers, 2003. 303p.

[2] BERTOZZI, D. The data link layer in NoC design. In: DE MICHELI, Giovani; BENINI, Luca. Networks-on-Chip: technology and tools. 2006. Amsterdam: Boston: Elsevier: Morgan Kaufmann Publishers, 2006.

[3] ZEFERINO, Cesar Albenes; SANTO, F. G. M. E.; SUSIN, A. A. . ParIS: A Parameterizable Interconnect Switch for Networks-on-Chip. In: SYMPOSIUM ON INTEGRATED CIRCUITS AND SYSTEMS, 17., 2004, Porto de Galinhas. Proceedings... New York : ACM Press, 2004. p. 204-209

[4] ZEFERINO, Cesar Albenes; KREUTZ, M. E.; SUSIN, A. A. . RASoC: A Router Soft-Core for Networks-on-Chip. In: (INT) DESIGN AUTOMATION & TEST IN EUROPE (DATE) - DESIGNER´S FORUM, 2004, Paris. Proceedings... Piscataway: IEEE Computer Society, 2004. p. 198-205.

[5] FRANTZ, Arthur Pereira. Designing fault tolerant NoCs to improve reliabitity on SoCs. 2007. Dissertação (Mestrado) – Programa de Pós-Graduação m Computação. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2007.

[6] VEIGA, Fabrício; ZEFERINO, Cesar Albenes. Implementation of techniques for fault tolerance in a Network-on-Chip. In: SYMPOSIUM ON COMPUTING SYSTEMS (WSCAD-SCC 2010), 11., 2010, Petrópolis. Proceedings... Los Alamitos : IEEE Computer Society, 2010. p. 80-87.

[7] YING, Zang; HUAWEI, Li; XIAOWEI, Li; Reliable Network-on-Chip Router for Crosstalk and Soft Error Tolerance. In: ASIAN TEST SYMPOSIUM (ATS 2008), 2008. Proceedings… Sapporo: IEEE Computer Society, 2008.