

FPGA BASED HIGHLY RELIABLE FAULT TOLERANT APPROACH FOR NETWORK ON CHIP (NOC)

Jehosheba Margaret.M¹, Mary Susanna. M², Rajapirian.P³

¹Student, VLSI Design, Kings college of Engineering, Thanjavur, Tamilnadu, India. E-mail:jehosh17@gmail.com

²Faculty, Department of Electronics and Communication Engineering, PRIST University, Kumbakonam, Tamilnadu, India

³Faculty, Department of Electronics and Communication Engineering, Kings College of Engineering, Thanjavur, Tamilnadu, India

Abstract

Network on chip is an interconnection between several processing elements and routers. There are several possibilities for the occurrence of faults within the network. These faults degrade the performance of the network. In order to increase the performance several fault tolerant methods has been used. They involve themselves in rerouting and hence take longer paths. To make the path shorter, the router architecture has to be modified. The introduction of minimal routing algorithm for faults in the network increases the overall performance of the network. When we use the algorithm, it takes shortest path regardless of the presence of faults. The proposed algorithm is much simpler than the previous existing algorithm. It provides link among the surviving routers in the network. It proves to be more efficient even in the presence of multiple faults. It has the ability to connect the routers both horizontally and orthogonally even in the presence of faults. It has reduced delay over the network. It has proved to be more reliable of 99.5% when multiple faults are found in the network. It can be used in 4x4 mesh topology with six faulty routers to analyse the reliability of network. It also helps to estimate the various parameters such as reliability, latency, speed, area and power.

Keywords- Network on Chip, faulty router, fault tolerant algorithm, chip network, faulty link.

1. INTRODUCTION

The recent technology in VLSI has made the chip to be much smaller in size. To make it possible, the inner modules and their interconnections are brought into small size. Networks-on-Chip (NoC) has emerged as a promising solution for on-chip interconnection in Multi Core NoCs due to its scalability, reusability, flexibility, and parallelism [1] [2] [3]. NoC consists of Network Interfaces, Routers, set of links interconnecting the Routers and a defined communication protocol for IP core interaction [4]. NoC architecture design follows the computation architecture design, which partitions a behaviour model and maps it onto an computation architecture model [5]. The first step is communication modelling and analysis. The second step is topology and protocol design based on the communication requirements. The aim of this step is to use the least network resource to fulfil the communication requirements showed by the communication pattern. To estimate the delay on each interconnection in term of clock cycles and analyse power and area of a NoC design, the chip floor plan should be estimated in the third step to get some design requirements for interconnections. The fourth step is the performance analysis which simulates the NoC architecture. In the fifth step, logic design can help increase the accuracy of power and area analysis, but the

interconnection design is the centre of the analysis.

Faults refer to the failure of the system. Transient and permanent faults are two different types of faults that can occur in on-chip networks [6]. Transient faults are temporary and unpredictable. They are often difficult to be detected and corrected. Permanent faults are caused by physical damages such as manufacturing defects and device wear-out. These faults should be recovered or tolerated in a way that the network continues functioning. A deterministic routing algorithm uses a fixed path for each pair of nodes resulting in increased packet latency especially in congested networks [7]. In adaptive routing algorithms, packets are not restricted to a single path when traversing from a source to a destination router [8]. Moreover, non-minimal methods are usually more complex with a larger number of virtual channels. These algorithms are used in order to reroute packets around faults [9].

Some fault-tolerant algorithms are proposed to support special cases of faults, such as one-faulty routers, convex or concave regions. These algorithms either disable the healthy components or require a large number of virtual channels to avoid deadlock. There are other fault-tolerant approaches [10] [11] [12] which do not require any virtual channels. A common behaviour in fault-tolerant approaches

is that packets are routed normally in the network until they face a fault. At this time, turn models or other techniques are used to reroute packets around the faults such that no cycles be formed in the network.

In this paper, we present a new approach to tolerate multiple faults in the network. This approach is termed as Connection Retaining Fault tolerant approach which works more effectively under faults. When a router becomes faulty, the router is considered as a link to send the data from source router to destination router. If a link is faulty, it changes its direction so that the data reaches its destination router.

The rest of this paper is organized as follows: Section II reviews the related work. Basic details of the router are given in Section III. The proposed fault-tolerant routing algorithm is presented in Section IV. Multiple faulty cases are discussed in Section V. The results are investigated in Section VI. Finally the paper is concluded with a summary in the last section.

2. RELATED WORK

There are few papers discussed to deal with faults in links [12][13] and routers[11][14]. Few methods never use virtual channels[9][10][11][15]. Some methods like FTDR[17] and HARAQ[18] use routing table. Hence they require large area [16][17]. Few approaches do not use routing table [18] but they take non minimal path. The routing algorithm scans the position in descending order and selects its position. BFT-NoC[2] tolerate faulty links by dynamic sharing of surviving channels. The algorithm [12] tolerate single fault. Another algorithm [14] tolerates two faulty links in the network. DBP [19] approach uses a lightweight approach to maintain the network connectivity among non-faulty routers. Here the connections are embedded inside the chip. The turn based model is used to avoid deadlock conditions [12][15][16]. The turn-based routing methods have lower hardware costs but also lower performance because of having less adaptivity.

The connection retaining algorithm has the following advantages:(i) It provides shortest path in the network. (ii) It requires knowledge about neighbouring router. (iii) It consists few lines of codes. (iv) requires only two virtual channels. (v) When router is faulty, it is considered as link. (vi) High reliability. (vii) Does not convey extra information about the routing process. Reachability is another important factor as it consists of reachable faulty neighbours.

2.1 Basic Router

Routers are considered to be the most important blocks in network on chip. Routers consist of few main components like buffers, ports, crossbar switch, flow control and arbiter.

There are four ports indicating the four directions namely North, South, East and West. A Local port is always connected to the processing element. In a XY network, each input pair is connected to the output pair.

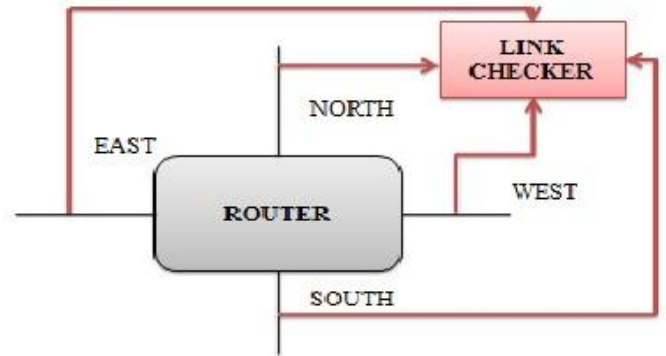


Fig.1. Basic Router

The type of router used in this method is double-Y network. It provides adaptiveness with minimum number of virtual channels. The network is partitioned into two sub networks namely +X and -X. Eastward channel route through the +X part in Y-direction whereas Westward channel route through the -X part in Y-direction.

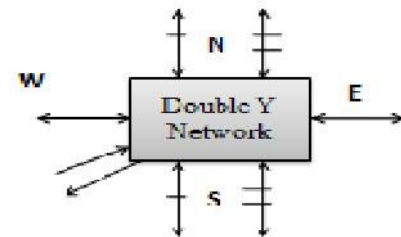


Fig 2. Router in Double-Y network

Flow control is a mechanism decides how the network resources are allocated in the network. Buffers hold the data in the router which are stored temporarily. Buffers are provided for the four different directions namely North, South, West and East.

3. ARCHITECTURE

3.1 Router Architecture

Faults are considered to be failure of a system. There are several possibilities for the occurrence of faults in the network. Since NoC consist of link, core and router, it may lead to link fault, core fault and router fault respectively. The network continues to work in case of link fault and core fault but fail to work when the router fails. Faults in the link can be rectified by the dynamic sharing of the

remaining channels. When a router fails, it cannot send or receive the data from one location to another. It causes entire system to fail leaving the core and link as faulty. The core can perform its regular operation only when the fault is recovered.

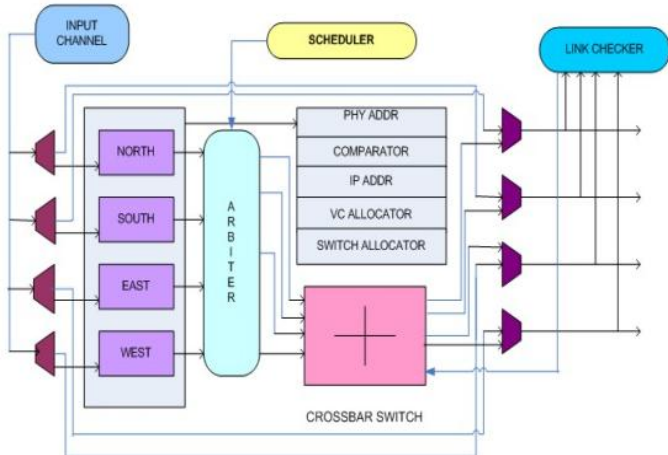


Fig.3. Modified router architecture

In this proposed method, the router architecture can still be used even in the presence of faults. In the normal router architecture, it consists of input buffers, routing unit, virtual channel allocator, switch allocator and crossbar switch. In this method, the architecture is modified with reference to the normal architecture. It needs extra multiplexer, demultiplexer, link checker and wiring connections. When the router is faulty, the west input channel is directed towards the east output channel while the east input channel is directed towards the west output channel. In the same way, the north input channel is directed towards the south output channel while south input channel is directed towards the north output channel. It needs extra wiring and costs overhead.

The above architecture is used in 4x4 mesh topology. It can tolerate multiple number of faults in the network by acting as a link with its neighbouring routers.

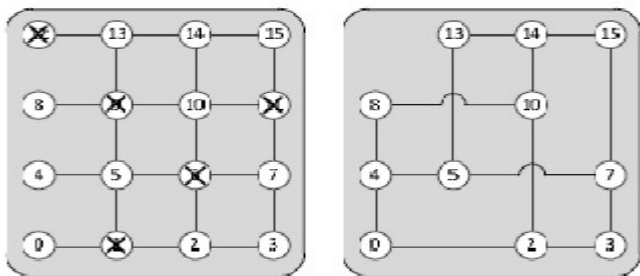


Fig 4 4x4 Mesh topology network with fault and their result after applying algorithm

3.2 Fault Tolerant Algorithm In Router Architecture

In this section, the fault tolerant algorithm is used to tolerate the number of faults present in the entire network. It supports the faulty router by taking non minimal paths. Due to the characteristics of adaptiveness, the data chooses different path by bypassing the faulty router to reach the destination router from the source router. By default all the data are routed in the Y-direction of the network. If the neighbouring routers are faulty in the Y-direction then the data is routed in the X-direction.

If there are three routers in the network with one as faulty then the data is transferred by bypassing the faulty router. In case of six routers in the network with one faulty router in the Y-direction of the source router then the data is sent in the X-direction. After selecting the path, it further sends the data to the next router to reach destination. If the next hop router is faulty, it is considered as a link. Finally, it reaches its destination router. If the destination is two hops away from the source then the data is routed to the non faulty router first. It tries the best to avoid routing through the faulty router.

Suppose, if both the direction are non faulty then it moves through the less congested area. If the network is designed to tolerate a few numbers of faults, packets can be routed adaptively in the network as long as the remaining distance along both directions is equal or greater than two hops. To sum up, it is guaranteed that all situations of a single faulty router are covered by the retaining algorithm taking only the shortest paths between each pair of source and destination routers. In addition, fully adaptiveness is provided when the packet is not close to the destination router (i.e. the packet is close to the destination when the distance along one dimension is at most one hop). However, for better reliability, the adaptiveness of algorithm can be limited. If the distance is equal in both directions then the data is sent in either way.



Fig 5 Tolerating one fault

Moreover, this algorithm requires only the information about the adjacent neighbouring routers. So, it requires low area overhead. The presence of faults occurs mostly in additional resources and less used resources. This makes the reliability to be more efficient.

3.3 Link Checker

Link checker is used to check the connection (i.e., link) between the two adjacent routers. In a network, all routers are well connected to each other in a well established manner. There are several possibilities for the failure of link. Care must be taken so that these failures do not affect network performance.

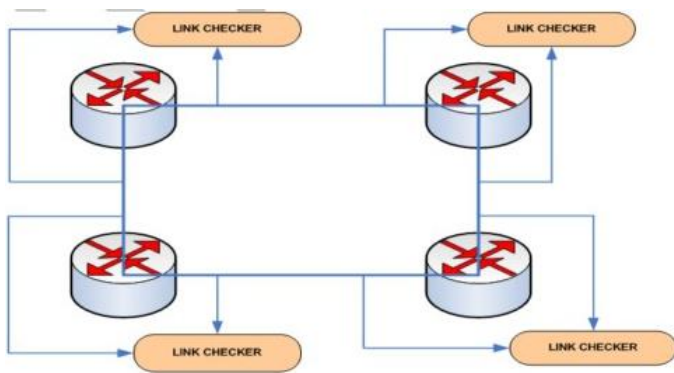


Fig 6 Link checker with router

If a link fails, the transfer of data is stopped. It is necessary to have the information about the status of the links in order to complete the data transfer from source router to destination router. The usage of link checker prevents performance degradation in the network.

4. RELIABILITY OF ROUTERS

In this section, it deals how the faulty routers are tolerated in the presence of two or more faults. When the number of faults is increased, it undergoes several problems in the network. The solution can be brought by analysing the probabilities of faults occurrence.

4.1 Reliability for Two Faulty Routers

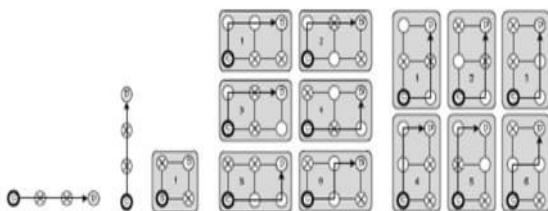


Fig. 7. Tolerating two faulty routers

Reliability estimation is carried out under two possible conditions in the network.

Reliability1:

By using the fault tolerant approach how the data is successfully sent under the existence of faults in the network. At first, we have to calculate the number of total combinations of two faulty routers in the network. Then, we have to measure the number of combinations in which two faults occur in diagonal positions. By dividing these two numbers, the reliability value is obtained. The number of different combinations of two faulty switches in an nxn mesh network can be measured by using the formula,

N(all combinations):

$$N = (n^2(n^2 - 1)) / 2 \tag{1}$$

N(diagonal combinations):

$$N = 2(n - 1)^2 \tag{2}$$

$$\text{Reliability1} = 1 - 4((n - 1)^2 / n^2(n^2 - 1)) \tag{3}$$

Reliability2:

The estimation of reliability considered when the data is sent over the faulty router. Only the healthy routers send the data to another router. Any faulty router cannot send the data but only can receive them. The total number of data per combination is given by,

$$N(\text{delivered per combination}) = (n^2 - 2)(n^2 - 3) \tag{4}$$

Total = N(delivered per combination) * N(all combinations).

Total number of data defeated,

$$\text{Defeated} = 2 \times N(\text{diagonal combinations}) = 4(n - 1)^2 \tag{5}$$

Reliability2, $R_2 = 1 - (\text{Defeated} / \text{Total})$.

4.2 Reliability for Multiple Routers

Reliability check is made for multiple router faults. The number of faults in the routers cannot be increased for small size networks. At least one non faulty router must present in the network. As the number faulty routers increases, the performance decreases. Even under these severe fault conditions, the rest of the packets can be routed to their destinations through the shortest paths. The focus of this work is to tolerate faults by using the shortest paths without any performance loss. However, non minimal paths or virtual channels can be used to support the remaining cases.

Based on the above two methods the reliability of the router is estimated.

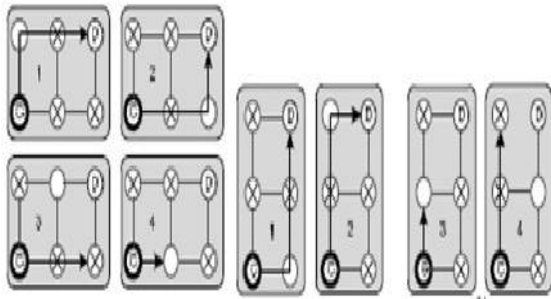


Fig. 8 Tolerating multiple faulty router

5. EXPERIMENTAL RESULTS

To evaluate the performance of the network, an NoC simulator is used along with the VHDL codes is applied over the entire components of the Network on Chip. The data width is set to 32 bits for all routers. Each input buffer can accommodate 8 flits in each virtual channel. As a performance metric, we use latency defined as the number of cycles between the initiation of a message issued by a Processing Element (PE) and the time when the message is completely delivered to the destination PE.

The performance estimation of this connection retaining approach is compared with two methods, called Reconfigurable Switch algorithm [13] and Reconfigurable Links algorithm [14]. In the other methods, packets may take unnecessary longer paths to reach destinations. Reconfigurable Switch algorithm requires one virtual channel along each dimension and is able to tolerate all single faulty routers. Reconfigurable Links utilizes two virtual channels to support all two faulty routers.

5.1 Reliability for Uniform Traffic

In the uniform traffic profile, each processing element (PE) generates data packets and sends them to another PE using a uniform distribution [20]. The mesh size is considered to be 4x4. To evaluate the reliability of connection retaining algorithm, the number of faulty routers increases from one to six. All faulty routers are selected using a random function and its performance is evaluated.

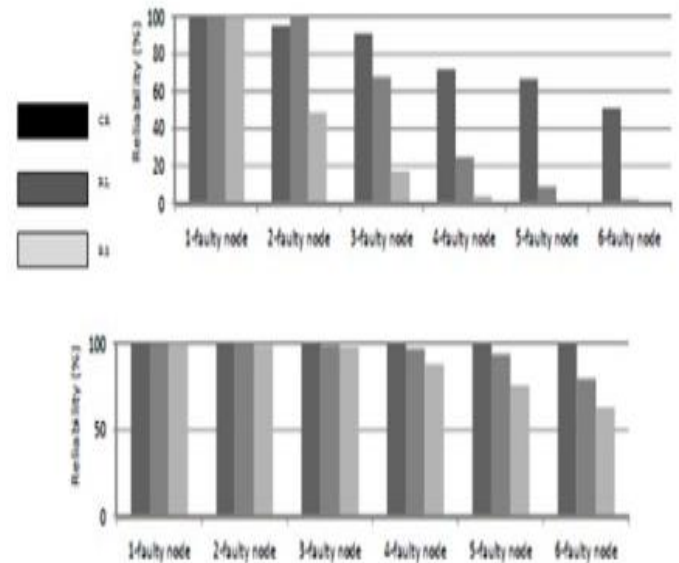


Fig 9 Reliability Analysis

Reliability is based on the above two performance metric measurement. When the faults are considered to be minimum, the reliability is good for all the three methods. As the number of faults increases, the other two methods degrade in their performance. Unlike the other methods connection retaining method holds good under multiple faults. It is proved to be more efficient with the reliability estimation of about 99.5% under multiple faults in the network.

5.2 Performance Analysis for Uniform Traffic

The average communication latency of Reconfigurable Switch algorithm and Reconfigurable Links are obtained under the cases of single and two faulty routers as more faulty routers are not well supported. The average communication latency of Connection retaining fault tolerant algorithm is measured under one to six faulty routers in the network. To have a fair performance comparison we use two virtual channels in all three methods. The extra virtual channels are used for the performance purposes. In a fault-free network, the performance of all methods is comparable while Reconfigurable Links outperforms others because of its better adaptiveness. As the number of faults increases, the performance of Reconfigurable Switch algorithm and Reconfigurable Links significantly decreases. We increase the number of faulty routers to six faults and measure the performance of connection retaining method. Surprisingly, the performance gradually starts growing under the same traffic load.

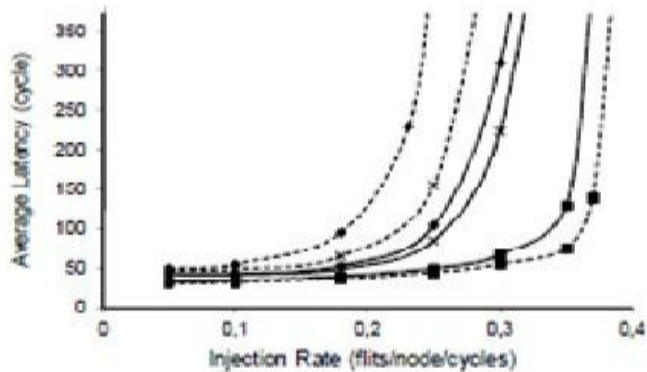


Fig 10 Performance Analysis under uniform traffic

5.3 Performance Analysis for Hotspot Traffic

Under the hotspot traffic pattern, one or more routers are chosen as hotspots receiving an extra portion of the traffic in addition to the regular uniform traffic. The performance Analysis is for Connection retaining algorithm, Reconfigurable Link and Reconfigurable Switch. Connection retaining algorithm proves to more promising under hot spot condition.

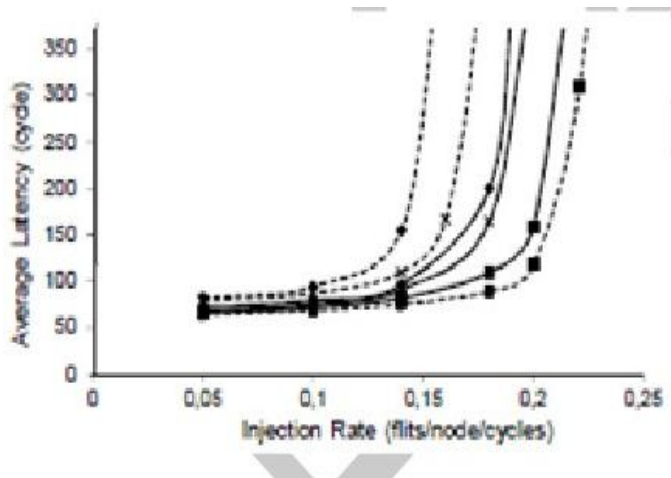


Fig 11 Performance Analysis under hotspot traffic

6. CONCLUSIONS

In this paper, we proposed a fault-tolerant approach to retain the connectivity of the network. In the presented approach, data are routed through the shortest paths, maintaining the performance of NoC even in the presence of multiple faults. The router architecture is slightly modified to route the data safely between the routers. The main purpose of the modification in the router architecture is to maintain the connectivity among the remaining routers. For this to happen, when a router becomes faulty, the router links are connected to each other along the

horizontal and orthogonal directions. This method is a very simple, lightweight, and adaptive approach. It takes only one and two virtual channels along the X and Y dimensions respectively. Obtaining high reliability by a simple approach is the conclusion of this work.

REFERENCES

- [1] Xu, Jiang, W. Wolf, J. Hankel, S. Charkdhar, "A Methodology for design, modeling and analysis for networks-on-Chip," in Proc. of IEEE International Symposium on Circuits and Systems, pp. 1778-1781, 2005.
- [2] W. Tsai, D. Zheng, S. Chen, and Y.H. Hu, "A fault-tolerant NoC scheme using bidirectional channel", in Proc. DAC, pp.918-923, 2011.
- [3] E. Rijpkema et al., "Tradeoffs in the design of a router with both guaranteed and best-effort services for networks on chip," in Proc. Of DATE'03, pp. 350-355, 2003.
- [4] Kangmin Lee, Se-Joong Lee, Hoi-Jun Yoo, "Low-power network-on-chip for high-performance SoC design", press, 2008.
- [5] L. Benini, G. De Micheli, "Networks on chip: a new paradigm for systems on chip design", Design, Automation and Test in Europe Conference and Exhibition, 2002.
- [6] M. Ali, M. Welzl, S.Hessler, "A Fault tolerant mechanism for handling Permanent and Transient Failures in a Network on Chip," in Proc. Of international conference on Information Technology, pp.1027-1032, 2007.
- [7] J. Duato, S. Yalamanchili, L. Ni, "Interconnection networks: an engineering approach", Morgan Kaufmann Publishers, 2003.
- [8] M. Daneshlab, M. Kamali, M. Ebrahimi, S. Mohammadi, A. Afzali-Kusha, and J. Plosila, "Adaptive Input-output Selection Based On-Chip Router Architecture," Journal of Low Power Electronics (JOLPE), Vol.8, No. 1, pp. 11-29, 2012.
- [9] F. Chaix, et al., "A fault-tolerant deadlock-free adaptive routing for On Chip interconnects," in Proc. of DATE, pp. 1-4, 2011.
- [10] S. Jovanovic, C. Tanougast, et al., "A new deadlock-free fault-tolerant routing algorithm for NoC interconnections", in Proc. of FPL, pp.326-331, 2009.
- [11] J. Wu and D. Wang, "Fault-tolerant and deadlock-free routing in 2-D meshes using rectilinear-monotone polygonal fault blocks", in Proc. Of Parallel Algorithms., pp.99-111, 2005.
- [12] Z. Zhen, A. Greiner, S. Taktak, "A reconfigurable routing algorithm for a fault-tolerant 2D-Mesh Network-on-Chip", DAC, pp. 441-446, 2008.
- [13] M. Ebrahimi et al., "MAFA: Adaptive Fault-Tolerant Routing Algorithm for Networks-on-Chip," in Proc.

- of DSD, pp. 201-206, 2012.
- [14] M. Valinataja, S. Mohammadi, J. Plosila, P. Liljeberg, H. Tenhunen, "A reconfigurable and adaptive routing method for fault-tolerant meshbased networks-on-chip," International Journal of Electronics and Communications (AEU), v. 65, I.7, pp. 630-640, 2011.
- [15] D.Fick, A. DeOrio, G. Chen, V. Bertacco, D. Sylvester and D. Blaauw, "A highly resilient routing algorithm for fault-tolerant NoCs," Design, Automation and Test in Europe (DATE), pp. 21-26, 2009.
- [16] M. Valinataj, S. Mohammadi, S. Safari and J. Plosila, "A link failure aware routing algorithm for Networks-on-Chip in nano technologies," IEEE Conference on Nanotechnology, pp. 841-844, 2009.
- [17] Ch. Feng, Zh. L, A. Jantsch, J. Li, M. Zhang, "A Reconfigurable Fault tolerant Deflection Routing Algorithm Based on Reinforcement Learning for Network-on-Chip", in Proc. of NoCArc, 2010.
- [18] M. Ebrahimi, M. Daneshtalab, F. Farahnakian, P. Liljeberg, J. Plosila, M. Palesi, and H. Tenhunen, "HARAQ: Congestion-Aware Learning Model for Highly Adaptive Routing Algorithm in On-Chip Networks," in Proc of NOCS, pp. 19-26, 2012.
- [19] M. Koibuchi, H. Matsutani, H. Amano, T.M. Pinkston, "A Lightweight Fault-Tolerant Mechanism for Network-on-Chip", in Proc. of NoCS, pp.13-22, 2008.
- [20] C.J. Glass et al., "The Turn Model for Adaptive Routing", in Proc. Of 19th Int'l Symp. Computer Architecture, pp. 278-287, 1992