

# Exploring Multistage Interconnection Network Implementations for FPGA-based MPSOCs

Yassine AYDI, Ramzi TLIGUE, Mohamed ABID

CES Laboratory

National Engineering School of Sfax

Email: [yassine.aydi@oous.rnu.tn](mailto:yassine.aydi@oous.rnu.tn), [ramzitligue@ieee.org](mailto:ramzitligue@ieee.org), [mohamed.abid@enis.rnu.tn](mailto:mohamed.abid@enis.rnu.tn)

Jean-Luc DEKEYSER

LIFL and INRIA-Lille,

University of Lille, France

Email: [dekeyser@lifl.fr](mailto:dekeyser@lifl.fr)

**Abstract**— Modern embedded systems integrated a variety of complex and heterogeneous components communicating with each other at high-speed rates. The interconnection architecture employed in such systems has an important impact to their overall performance. Multistage interconnection network has emerged as a promising alternative to ensure communication for Multi Processor System on Chips. In this paper, we describe a design methodology of MIN-based MPSOC on programmable circuits (FPGAs). As a case study, we propose an evaluation methodology based on performance metrics that include latency, execution time and area requirements for two MIN-based MPSOC topologies, i.e. the Omega and the Butterfly Fat Tree (BFT). Moreover, we show in our experimental results that multistage interconnection networks will become a future general purpose interconnect for FPGA-based MPSOCs much like today's standard bus architectures.

## I. INTRODUCTION

A large number of applications require more and more intensive computations, especially real-time and multimedia applications, and are integrated onto the same chip. In order to improve the performance of current embedded systems, Multiprocessor System on Chip (MPSOC) is a recent solution for increasing computational system performance. MPSOC have to be energy efficient, cheap, reliable, and must offer sufficient computing power for advanced and complex applications [1]. So an efficient interconnection network intra-chip [2][3], with low latency and high bandwidth, is fundamental for high communication throughput among the components. As a promising alternative, Networks on Chip (NoC) have been proposed by academia and industry to handle communication needs for the future multiprocessor systems on chip [4].

Multistage Interconnection Network (MIN) has been used in classical multiprocessor systems. As an example, MINs are frequently used to connect the nodes of IBMSP [5] and CRAY Y-MP series [6]. Further on, MINs are applied for networks on chip to connect processors to memory modules in MPSOC. A MIN is defined by its topology, switching strategy, routing algorithm, scheduling mechanism, fault tolerance, and dynamic reconfigurability [7].

An essential step in the design of an MPSOC is the verification of the whole system, and especially of the selected communication architecture. Traditionally, this verification is synonym with simulation which consists on the performance evaluation of the system [8]. However, such

technique provides partial verification, so it cannot cover all design errors or detect undesirable situations (deadlock, starvation). The trend is then to adopt formal verification, which is based on using methods of mathematical proof to ensure the quality of the design, improve the robustness of the system, and speed up the development [9]. To compare and contrast different communication architectures, a standard set of performance metrics should be evaluated, such as area, energy consumption, execution time and latency. Therefore, the easy programmability and the large integration capacity of FPGA provide a faster performance evaluation through emulation which complements the formal verification process of the communication architecture.

FPGA manufacturers propose environments to design multi processor systems on chip. Generally, their tools adopt standard FPGA communication architecture. This is apparent from Xilinx FPGAs adopting IBM's PLB bus architecture to be compatible with their embedded PowerPCs. Complex MPSOCs require higher bandwidths and to be more area efficient than a bus can offer. With technology scaling, if several NoC implementations are reported up-to-date, a few NoC-Based FPGA implementations are proposed. A design methodology of MIN-based MPSOC on FPGA is investigated in this paper. Section 2 outlines some related work in this area. Next, MIN architecture is introduced. Section 4 details the Modeling and implementation of MIN's architecture on FPGA. Then an integration of our NoC in MPSOC architecture is also described in section 4. Hardware synthesis and Simulation results are detailed in section 5. Finally, we conclude the paper and we give directions for future work.

## II. RELATED WORK

Several interconnect architectures on FPGA ranging from busses [10], crossbars [11], to networks on chip [12] [13] have been reported in the literature. We briefly describe below some communication architecture designs proposed recently. Kumar [14] has proposed a reconfigurable Multi Processor Network on Chip architecture in which both the network and the processing nodes are fully configurable, and the network is designed to match the application requirements. A design of an adaptable FPGA-based Network on Chip dynamically reconfigurable has been proposed by pionteck et al[15]. Bobda and Ahmadian present two approaches addressing the problem of online reconfiguration

of Network on Chip. First, they present a circuit-routing solution based on the concept of a reconfigurable multiple bus on chip (RMBoC). The second solution denoted a generic 2D dynamic model (DyNoC), targets devices with unlimited reconfiguration capabilities. The feasibility of the approaches has demonstrated through analysis and examples [16]. In [17], a reprogrammable interconnect is implemented based on a LUT-based bus macro and is used to dynamically reconfigure the attached IPs.

From a complementary design consideration, we proposed a design methodology of MIN-based MPSOC on FPGAs. As a case study, we propose an evaluation methodology based on performance metrics that include latency, throughput and area requirements for two MIN-based MPSOC topologies.

### III. MIN OVERVIEW

In this section, we present an overview of the networks used for the design of the MIN-based platform for MPSOC.

#### A. MIN Components

The common multistage interconnection networks (MINs) used, have  $N$  inputs and  $N$  outputs nodes and are built using  $r \times r$  switches. Such MINs have  $N/r$  switches at each stage, and  $\log N$  stages of switches denoted  $d$ . The different stages are connected by links generated by applying permutation functions. In a MIN, a path between a source and a target is obtained by operating each corresponding switch of the stage  $i$  in straight mode if the  $i^{th}$  bit of the destination address is equal to 1, otherwise in exchange mode.

#### B. Banyan MIN

Banyan MIN is a multistage interconnection network characterized by one and only one path between each source and destination. A banyan MIN of size  $N \times N$  consists of  $r \times r$  crossbars. An interesting subclass of Banyan MINs is composed of Delta networks. Let denote by:  $o_i$  the  $i^{th}$  output of a crossbar in a MIN, and by  $C_j$ , a crossbar belonging to the stage  $j$ . So, the Delta property can be defined as follows: if an input of  $C_j$  is connected to the output  $o_i$  of  $C_{j-1}$ , then all other inputs of  $C_j$  must be connected to the stage  $(j-1)$  on outputs with the same index  $i$ .

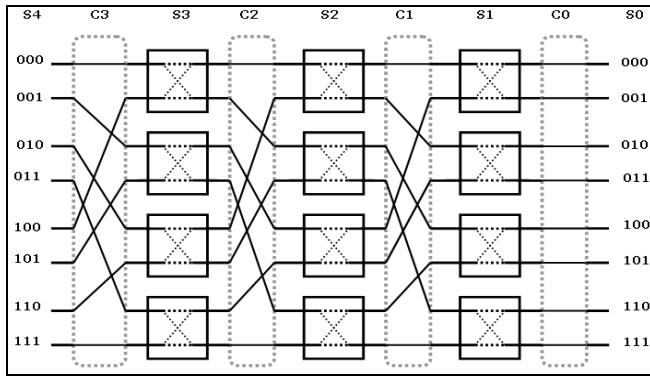


Fig.1. A Delta network (8,2)

The difference between each of the existing MINs is the topology of interconnection links between the crossbar

stages. A study of equivalence of a variety of Delta MINs has been detailed in [18]. An example of Delta networks (a subclass of MINs) illustrated in figure 1 connects 8 processors to 8 memories by means of 3 stages of 4 switches each. Processors and memories are represented by 3-bit number  $(d_2 \ d_1 \ d_0)_2$ . The interconnection stages denoted  $C_i$  ( $0 \leq i \leq 3$ ) are generated by applying permutation functions.

#### C. Butterfly Fat Tree MIN

The Butterfly Fat Tree (BFT) topology is a particular subset of multistage interconnection networks (MINs). As shown in figure 2, the network is modeled in the form of a tree. The components (processor, IP, memory...) are placed at the leaves and switches placed at the vertices. A pair of coordinates labels each node,  $(l, p)$ , where  $l$  denotes a node's level and  $p$  denotes its position within that level. In general, at the lowest level, there are  $N$  components with addresses ranging from 0 to  $(N-1)$ . Each switch, denoted by  $S(l, p)$  has four child ports and two parent ports. The IPs are connected to  $N/4$  switches at the first level. The number of levels depends on the total number of IPs, i.e. for  $N$  IPs, the number of levels will be  $\log_4 N$ . In the  $j^{th}$  level of the tree, there are  $N/2^{j+1}$  switch. At each next higher level of the tree the number of required switch reduces by a factor of 2, which illustrates that the number of switch tends to  $N/2$  as  $N$  grows arbitrarily large [19].

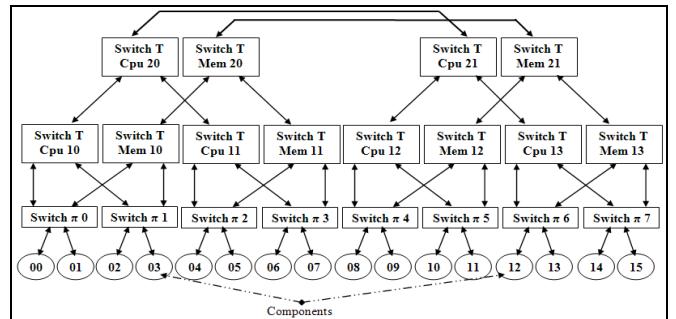


Fig.2. A Butterfly Fat Tree network

### IV. DESIGN OF MIN NETWORKS FOR FPGA-BASED MPSOCS

As previously hinted, this work provides a comparison in terms of performance and implementation costs between networks; Delta and Butterfly fat tree in order to test the reliability and the effectiveness of the MIN and to compare the performances of the networks. This section details the implementation of the MINs as well as completes MIN-based MPSOC on programmable circuits.

#### A. MPSOC architecture

The proposed architecture contains mainly  $N$  processors miniMIPS,  $N$  sharing data memories, MIN, and  $N$  instructions memories (Fig.3). The clock input is global to all components. All processors operate in parallel; each one is linked directly to an instruction memory. On the other hand, each processor can access to any data memory across the

MIN to read information or write data. The role of MIN is to manage access to data memories and avoid conflicts.

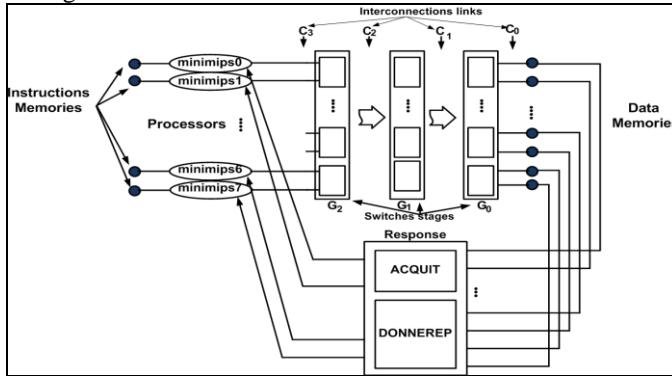


Fig.3. Proposed MPSoC architecture

### B. Test application

To test ease of use and performance of the networks a FIR-Filter and IIR-Filter applications were implemented.

The difference equation of an IIR Filter is:

$$y[n] = \sum_{k=0}^n b_k * C[n - k] - \sum_{k=1}^m a_k * y[n - k]$$

If we put  $a_i=0$  we get difference equation of the FIR.

Processors are used to illustrate functionality. The impulse response used in this example is:  $H(n) = \{ h(0), \dots, h(p) \}$ .

The input vector is:  $C(n) = \{ C_0, \dots, C_{16} \}$  and the generated output is:  $Y(n) = \{ y(0), \dots, y(23) \}$

$H(n)$  is stored in registers and used by the PEs when it is needed in the algorithm.

Each processor sends requests requiring the access to the data memories to bring inputs  $C_i$ , in order to carry out arithmetic operations. Finally it returns to data memory to store the obtained results. The distribution of the data in memories is random; consequently it is possible to use different distributions and to compare the performances of the network each time.

### C. MIN implementations on FPGA

We describe below the design of the networks. Our communication architectures are primarily composed of the following modules: Routing, Arbitration, Memorizing and Connection Module. We detail respectively the packet structure, and the switch architecture of Delta and Butterfly fat tree networks.

#### Delta MIN implementation:

- Packet structure: Traffic passed through the networks composed of fixed size packets. The packet format has three parts (Fig.4).

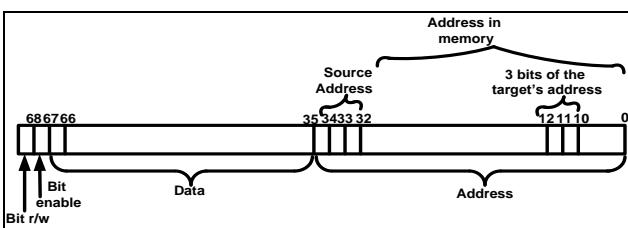


Fig.4. Packet format of Delta MIN

- Switch architecture: The switch is composed of  $2 \times 2$  crossbars, a Scheduler, a couple of input and output ports (Fig.5). Each input port of the router has dedicated buffer storage. Packets are buffered in input port until the output port of the next stage is ready to accept the packets; the width of each buffer is equal to the packet length in order to facilitate the routing strategy.

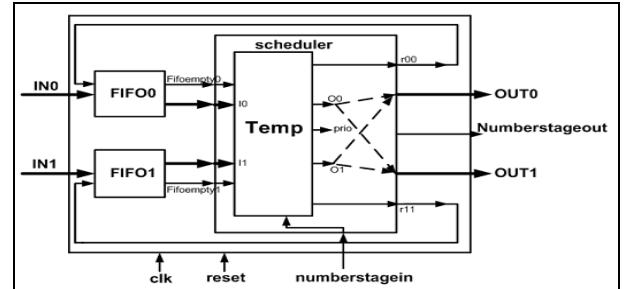


Fig.5. Switch architecture of Delta MIN

#### BFT MIN implementation:

- Packet structure: The data exchanged in BFT network are fragmented into packets. There are two kinds of packets: request and answer. Every packet is composed of three parts: (Fig.6, Fig.7). The request packet is generated by the processor and transferred towards the memory. The answer packet is generated by the memory and transferred towards the processor.

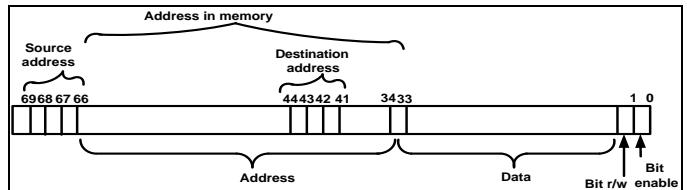


Fig.6. Request packet format of BFT MIN

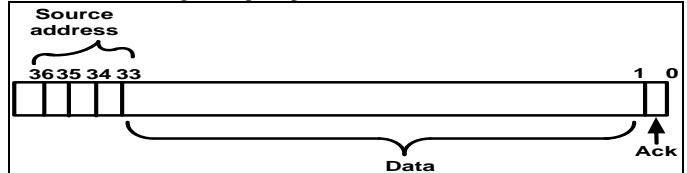


Fig.7. Respond packet format of BFT MIN

- Switch architecture: the bloc is primarily composed of routing Modules of  $\pi$  and T type which connect a processor with a memory, and FIFO as memorizing Modules. The router  $\pi$  (Fig.8), according to destination address and the routing algorithm, redirects the packets towards the memory (if the destination addresses is equal to stage addresses) or towards neighbor Switch in order to approach the final destination. The router T is composed of 3 bidirectional ports (left, right and down). All ports are connected to neighbors Switch (Fig.9). Each input port is preceded by a buffer (FIFO) to store the data temporarily. The scheduling is ensured by an arbiter implemented in the switch.

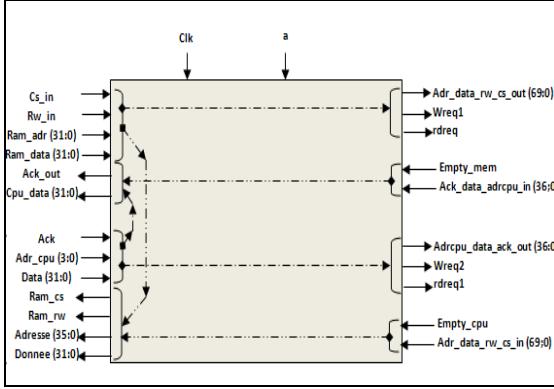


Fig.8. Router  $\pi$

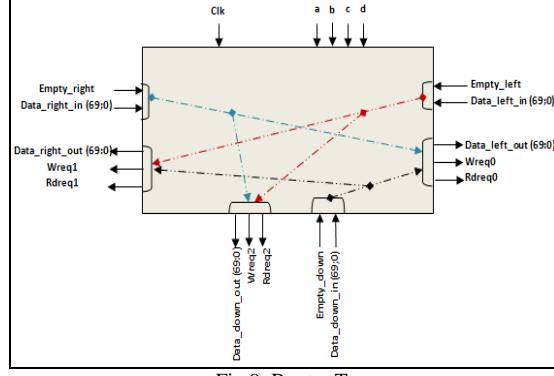


Fig.9. Router T

## V. PERFORMANCE ANALYSIS

Our intention in this paper is to design and evaluate different topology of dynamic network through case study of *FIR-Filter* and *IIR-Filter* applications with different number of cores. Subsequently, NOC evaluation metrics, such as area and average latency became essential aspects for optimizing the implementation of networks in MPSOC design. We have used VHDL and Xilinx ISE 9.1i to synthesize and target our components for Xilinx virtex4 XC4VLX200-FF1513 device.

### A. Switch performances

Three types of switch were conceived. Moreover we varied the depth of the FIFO used (2, 4 and 8 by FIFO) to evaluate the switch area consumption. It was noted that the most adequate solution for our switch is to use a FIFO of depth 8 because it has less FPGA area. This is explained by the use of FIFO blocks prefabricated by XILINX.

TABLE I  
Synthesis and latency of Delta MIN Switch

Depth of FIFO	Number of Slices	Logic Utilization		Number of FIFO16/ RAMB16s	Latency
		Number of Slice Flip Flops	Number of 4 input LUTs		
2	812	713	1182	0	3 cycles
4	826	719	1208	0	
8	568	587	929	4	

Target: xc4vlx200-11ff1513

TABLE II  
Synthesis and latency of Switch T (BFT)

Depth of FIFO	Number of Slices	Logic Utilization		Number of FIFO16/ RAMB16s	Latency
		Number of Slice Flip Flops	Number of 4 input LUTs		
2	804	664	1009	0	4 cycles
4	809	670	1018	0	
8	352	465	601	6	

Target: xc4vlx200-11ff1513

TABLE III  
Synthesis and latency of Switch  $\pi$  (BFT)

Depth of FIFO	Number of Slices	Logic Utilization		Number of FIFO16/ RAMB16s	Latency
		Number of Slice Flip Flops	Number of 4 input LUTs		
2	498	477	638	0	2 cycles
4	501	481	643	0	
8	245	345	362	4	

Target: xc4vlx200-11ff1513

As shown in TABLE I,II,III The Delta MIN switch is characterized by minimal latency and important area consumption on FPGA compared by the BFT switches which have less FPGA area and more latency.

### B. MIN performances

The performances estimation of the NOC is based on the three types of switch which we already mentioned. The results shown in TABLE IV and V were obtained by varying the number of the processors (4, 8 and 16).

TABLE IV

Synthesis of Delta MIN

Number of processor	Number of Slices	Target: xc4vlx200-11ff1513		Logic Utilization				
		%	Number of Slice Flip Flops	%	Number of 4 input LUTs	%	Number of FIFO16/ RAMB16s	%
4	2281	2	2364	1	3993	2	16	4
8	6839	7	7092	3	12257	6	48	14
16	18220	20	18912	10	33057	18	128	38

TABLE V

Synthesis of BFT network

Number of processor	Number of Slices	Target: xc4vlx200-11ff1513		Logic Utilization				
		%	Number of Slice Flip Flops	%	Number of 4 input LUTs	%	Number of FIFO16/ RAMB16s	%
4	3213	3	3196	1	5055	2	12	3
8	8019	9	8066	4	12418	6	36	10
16	18109	20	17818	10	28136	15	84	25

As the number of processor increase, the area utilization increase due to the increase of network size, also the

difference between the Delta MIN and BFT network in term of area consumption increase. This is because for the same number of processor BFT consumes fewer switches than Delta MIN.

### C. MPSOC performances

We saw in last section the performances of BFT and DELTA MIN which present a good solution to connect N processors to N memories. The objective of this section is to implement our MINs in a multiprocessor architecture, and then compare the performances of both networks.

TABLE VI  
Logic utilisation for 4x4 to 16x16 architecture

Number of processor	Delta MIN			BFT		
	4	8	16	4	8	16
Number of Slices	12697	30251	68432	12551	27372	55466
%	14	33	76	14	30	62
Number of Slice Flip Flops	10057	22909	49466	11212	23323	47623
%	5	12	27	6	13	26
Number of 4 input LUTs	23667	56534	127564	22857	48429	99553
%	13	31	71	12	27	55
Number of FIFO16/ RAMB16s	20	56	144	8	52	130
%	5	16	42	2	15	38

As it's shown through the two implementations of BFT and Delta MIN systems in TABLE VI, BFT networks consumes more than Delta MIN in term of area utilization, and as the number of processors increases (from 4 to 16) the differences between the two networks become bigger. These values confirm that the area of MIN implementations is proportional to  $N \log_2 N$ ; compared to  $N^2$  for full crossbar network (N defines the number of processors) [20].

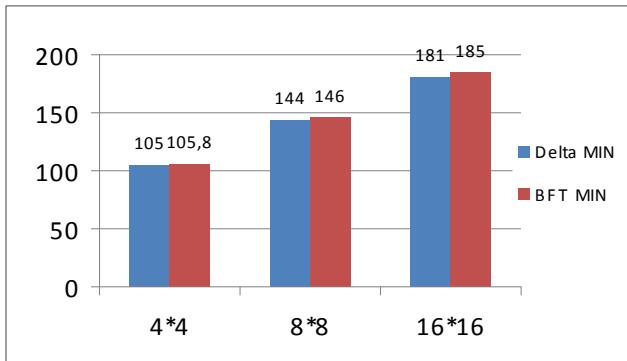


Fig.10.Latency of Filter FIR from 4 to 16 processors (in ns)

In order to test the functionality of our architecture, latency and execution time measurements were taken by varying the topologies of the MINs for the application: FIR-Filter. System clock rate is 50 MHz. Fig.10 shows that latency increases as the number of processors increases from

4 to 16 processors. This can be explained as follow: it is evident when increasing the number of nodes, the probability of conflict to occur in switches increases and message passes through networks slowly. Also, Delta MIN has shorter latency than BFT MIN at all processor sizes simulated. Furthermore, we find that execution time increases as the number of processors increases, and Delta MIN topology is the best in term of execution time for this application (Fig.11) for the operating modes. Finally, we concluded that changing MIN topologies affect slightly latency and execution time.

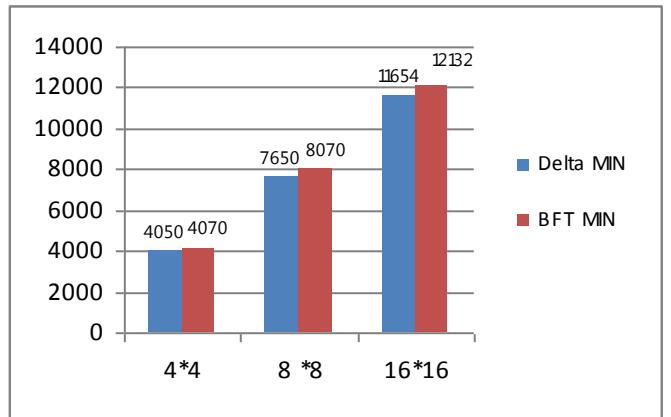


Fig.11. Execution time of Filter FIR from 4 to 16 processors (in ns)

## VI. CONCLUSION AND FUTURE WORK

In this paper, we have proposed the design methodology and the implementation of MPSOC architecture based on multistage interconnection networks on FPGA. Our objective in this work is to design and evaluate different topology of dynamic network through case study of FIR-Filter and IIR-Filter applications with different number of cores.

Design and modeling of the MINs were realized in three phases. We proposed two models of MIN networks, a DELTA MIN and a BFT MIN and we detailed the characteristics and the advantages of each one. In the second phase we implemented the MINs in a multiprocessor architecture. The last phase deals with the application in order to evaluate and test the reliability of our interconnection platform and to estimate its performances. For that purpose, two types of Filters were implemented: a Filter FIR and Filter IIR. The performances estimation was focused on FPGA area, the latency as well as the execution time.

The framework presented in this paper opens promising trend for further development as complement to verification techniques through emulation. We plan to extend this work to employ formal notations to validate the implementation of the communication architecture based on multistage interconnection network.

## VII. REFERENCES

[1] A. Jerraya, and W. Wolf, "Multiprocessor Systems-on-Chips", Morgan Kaufmann Publishers, San Francisco, 2004.

- [2] W. Wolf, “The future of multiprocessor systems-on-chips”, Proc. of the 41st annual conference on Design automation, ACM Press, New York, 2004, pp. 681–685.
- [3] S. Pasricha, “Floorplan-aware automated synthesis of bus-based communication architectures”, Proc. of the 42nd annual conference on Design automation, ACM Press, New York, 2005, pp. 565–570.
- [4] L. Benini, “Networks on chips: A new SoC paradigm”, Computer, Vol. 35, N° 1, IEEE Computer Society Press, Los Alamitos, California, 2002, pp. 70–78.
- [5] C. B. Stunkel, “The SP2 High-Performance Switch”, IBM Systems Journal, Vol. 34, N° 2, IBM Corp., Riverton, USA, 1995, pp. 185–204.
- [6] T. Cheung, “A simulation study of the CRAY X-MP memory system”, IEEE Transactions on Computers, Vol. 35, N° 7, IEEE Computer Society, Washington, 1986, pp. 613–622.
- [7] Y. Aydi, “Design and Performance Evaluation of a Reconfigurable Delta MIN for MPSOC”, In 19th International Conference on Microelectronics (ICM '07), 2007.
- [8] Y. Aydi, “Dynamicity Analysis of Delta MINs for MPSOC Architectures”, STA'07, 2007.
- [9] M. Elleuch., “Formal Specification of Delta MINs for MPSOC in the ACL2 Logic”, in Proceedings of Forum on Design and specification Languages - FDL '08, 2008.
- [10] P. Sedcole, “Modular dynamic reconfiguration in Virtex FPGAs”, IEEE Proc . Comp. and Dig. Tech., 153(3), 2006.
- [11] S. Wee, “A Practical FPGA based Framework for Novel CMP Research”, In FPGA, 2007.
- [12] N. Kapre, “Packet switched vs time multiplexed FPGA overlay networks”, In FCCM, 2006.
- [13] T. Marescaux, “Networks on chip as hardware components of an OS for reconfigurable systems”, In FPL, 2003.
- [14] A. Kumar, “An FPGA design flow for reconfigurable network-based multi-processor systems on chip”, In DATE 2007, pp. 117-122.
- [15] T. Pionteck, “A Dynamically Reconfigurable Packet-Switched Network-on-Chip”, In DATE 2006, pp. 136-137.
- [16] C. Bobda and A. Ahmadinia, “Dynamic Interconnection of Reconfigurable Modules on Reconfigurable Devices”, IEEE Design & Test, Vol.22, N° 5, IEEE Computer Society Press, 2005, pp. 443–451.
- [17] M. Huebner, “Scalable Application-Dependent Network on Chip Adaptivity for Dynamical Reconfigurable Real-Time Systems”, In FPL, 2004.
- [18] Kruskal, “A unified theory of interconnection network”, Theoretical Computer Science, Vol.48, N°1, Elsevier Science Publishers Ltd., essex, 1986, pp. 75-94.
- [19] Brett Stanley Feero, and Partha Pratim Pande, “Networks-on-Chip in a Three-Dimensional Environment: A Performance Evaluation”, IEEE Transactions on computers, Vol. 58, N°1, 2009, pp. 32-45.
- [20] B. Neji and Y. Aydi, “Multistage Interconnection Network for MPSOC: Performances study and prototyping on FPGA”, The 3rd International Design and Test Workshop, pp. 11-16.