IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 6, Issue 6, Ver. I (Nov. - Dec. 2016), PP 85-91
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Low Power 2-D Mesh Network-on-Chip Router using Clock Gating Techniques

Vijaykumar R Urkude[1], Dr. P. Sudhakara Rao[2]
[1]*Associate Professor, Dept of ECE, Vignan Institute of Technology and Science, Hyderabad, India*
[2]*Principal, Vignan's Institute of Management and Technology for Women, Hyderabad, India*

Abstract: *Network-on-Chip (NoC) is the platform of interconnection platform and is requirements of the modern on-Chip design. Area overhead, power consumption, and NoC performance is influenced by the router buffers. Resource sharing for on-chip network is critical to reduce the chip area and power consumption. An area efficient routing node for a NoC is presented in this paper. Out of the four components of routing node, the input block (mainly consisting of buffers) and scheduler have been modified to save area requirements. The other two components of the routing node take up negligible area in comparison. Custom SRAM is used in place of synthesizable flip flops in the input block. It has resulted in a saving of nearly 25% of the silicon area. Power is optimized by 65% when operated at 16 ns clock. Clock gating is high-level technique for decreasing the power consumption of a design. Clock gating reduces the clock network power dissipation, relaxes the data path timing, and decreases routing congestion by eliminating feedback multiplexer loops. For designs that have large multi-bit registers, clock gating can save power and reduce the number of gates in the design.*

Keywords: *Clock Gating, Network-on-Chip, Router, RTL, SRAM,*

I. Introduction

Increasing requirements on electronic systems are one of the key factors for evolution of the integrated circuit technology. Multiprocessing is the solution to meet the requirements of upcoming applications. Multiprocessing over heterogeneous functional units require efficient on chip communication [1, 2]. Network-on-Chip (NoC) is a general purpose on-chip communication concept that offers high throughput, and it is the basic requirement to deal with complexity of modern systems. NoC architecture is as shown in Fig 1.

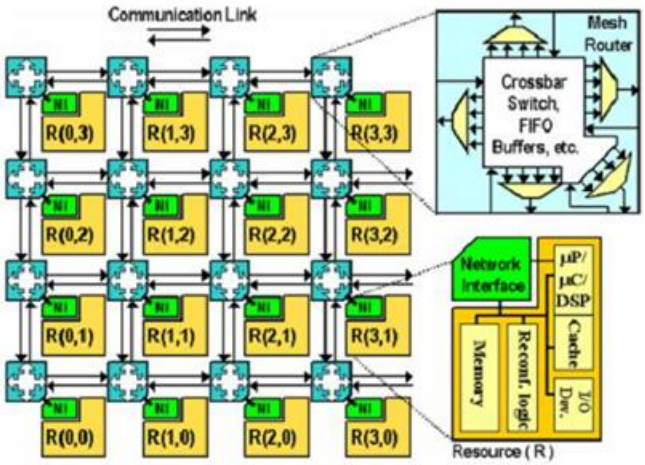

Fig 1. NoC Architecture

All links in NoC can be simultaneously used for data transmission, which provides a high level of parallelism and makes it attractive to replace the typical communication architectures like shared buses or point-to-point dedicated wires NoC platform is scalable and has the potential to keep up with the pace of technology advances [3]. But all these enhancements come at the expense of area and power. In the RAW multiprocessor system, interconnection network consumes 36% of the total chip power [4]. A typical NoC system consists of processing elements (PEs), network interfaces (NIs), routers and channels. The router further contains scheduler, switch and buffers. Buffers consume the 65% of the total node (router + link) leakage power for all process technologies, which makes it the largest power consumer in any NoC system [5]. Moreover, buffers are dominant for dynamic energy consumption [6].

II. Noc Architecture

Network-on-Chip has been proposed on various topologies [7-10]. A simple NoC architecture consists of three components: the routing nodes, the links, and network interfaces (or network adapters in some literature), as shown in Fig 2.

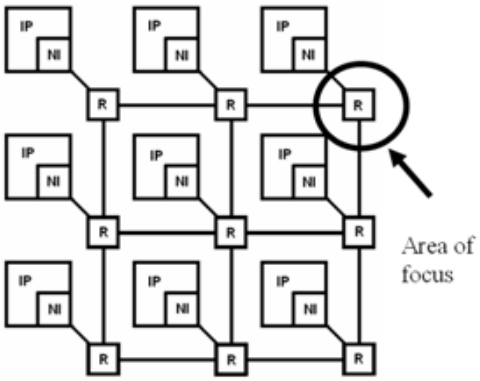

Fig 2. NoC Overview

Routers direct data through several links (hops). Topology defines their logical lay-out (connections) whereas floorplan defines the physical layout. The function of a network interface (adapter) is to decouple computation (the resources) from communication (the network). Routing defines the path taken from source to the destination whereas switching and flow control policies define the timing of transfers. Task scheduling refers to the order in which the application tasks are executed and task mapping defines which processing element (PE) executes certain task. IP mapping, on the other hand, defines how PEs and other resources are connected to the NoC [11]. The major goal of communication-centric design and NoC paradigm is to achieve greater design productivity and performance by handling the increasing parallelism, manufacturing complexity, wiring problems, and reliability. The three critical challenges for NoC are: area, power, latency, and CAD compatibility [12].The key research areas in network-on-chip design [13, 14].are as:

- Communication infrastructure: topology and link optimization, buffer sizing, floor-planning, clock domains, power.
- Communication paradigm: routing, switching, flow control, quality-of-service, network interfaces
- Application mapping: task mapping/scheduling and IP component mapping.

All of these challenges result in area, power, and performance tradeoffs [13].Area and power can be estimated from hardware requirements. Performance is generally estimated using analytical model. This paper proposes the area and power efficient design of the router as it is the most redundant component which is equal to the no. of PEs on one kind of NoC, as shown in Fig 2.

III. Problem Statement

The implementation of network-on-chip presents certain challenges. Two of the most critical design metrics for networks-on-chip are area requirements and power consumption. Due to the fact that die area per wafer of silicon is limited, the NoC implementation should be carried out using an approach that minimizes area requirement. Also due to likelihood of most SoCs being implemented in battery powered devices, power consumption of the NoC should also be as low as possible. Usually, reduction in area results in a saving in power requirements due to the fact a smaller area is achieved using fewer components on-chip. Fewer components on-chip will consume less power compared to architecture requiring more components on-chip.
Performance of digital systems can be enhanced by making use of custom IP cores to replace some of the standard-cell components. This performance enhancement comes at the price of increased design time and effort, but is preferred for maximizing performance.

IV. Design And Implementation Of Proposed Task

The routing node consists of four basic components: the input ports, the output ports, the crossbar switch, and the scheduler. The components arranged in decreasing order of size are the input blocks, the scheduler, the output blocks, and the crossbar switch as shown in Fig 3. The primary function of the input block is to store incoming packets before they can be routed to their respective output ports. Hence, the majority of the area of the input blocks is used by memory elements. The existing design employs DFF (D flip-flop) elements for memory storage. The modified input block will be based on SRAM memory cells. SRAM memory cells provide the fastest and most compact means of on-chip storage.

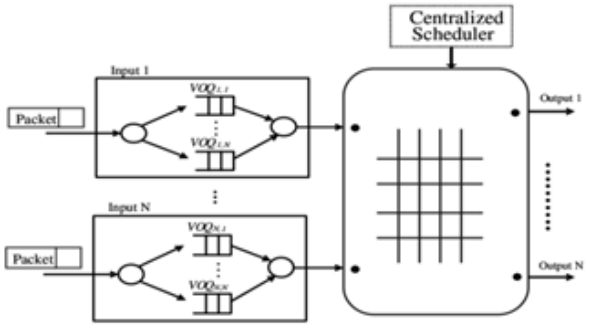

Fig 3 Router Components

The function of the scheduler is to arbitrate between conflicting requests for access to the crossbar switch shared medium. The existing scheduler architecture is based on a symmetric implementation of round-robin like algorithm requiring one set of grant arbiters and one set of accept arbiters to perform arbitration. The modified design uses the concept of folding to reduce the area of the scheduler by removing one set of arbiters and using the remaining set of arbiters to perform both grant and accept arbitration in a time multiplexed fashion. The design of the modified routing node is implemented using standard cell based VLSI flow with provision for custom IP core inclusion. The Cadence tool chain is used to implement the design from RTL coding to synthesis and place and route. Design verification is carried out using hierarchical functional simulation at each level of the design flow. Also, static timing analysis is used to verify timing closure in the final design layout. Area and power are two important parameters which need to be optimized for better NoC performance. The NoC consists of three basic components which are the routing node, the routing links, and network interfaces. Optimization of the routing nodes will lead to improvement in the area and the power requirements of the NoC, as it is the most redundant component which lies in association with every processing element in SOC. Thus, the aim of this work is to present a modified architecture of the routing node to achieve higher area and power efficiency using changes at the RTL architecture level and use of custom IP to boost the performance of standard-cell based ASIC design.

V. The Proposed Router Architecture

The routing node configuration shown in Fig 4 is 4x4. It is based on a 2D mesh NoC topology where each routing node is connected to four other routing nodes.

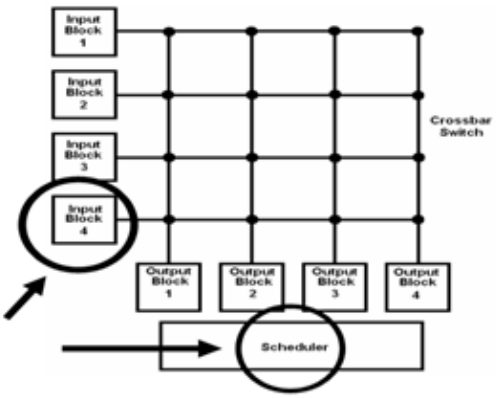

Fig 4. 4x4 Routing Node

The NOC infrastructure includes components responsible for packetization, transmission, and de-packetization of data. These components, respectively, are the NI, the VC router, and the links. These components are repeated for every grid element in NOC. So, if we consider a NOC with 3×3 mesh network, then it will have nine sets of components of NI, VC router and links. It can be clearly seen that these components will occupy a significant amount of silicon space on the chip and therefore the cost and the power consumption of the chip would increase. However, it must be noted that serial packet-based communication will still remain an optimum solution as compared to a bus-based system in terms of the power consumption and will reduce the cost of system design in the longer run due to the potential for reuse.
The design of the proposed router has been carried out as follows:

5.1 Proposed switching technique

Switching techniques can be classified based on network characteristics. Circuit switched networks reserve a physical path before transmitting the data packets, while packet switched networks transmit the packets without reserving the entire path. Packet switched networks can further be classified as Wormhole, Store and Forward (S&F), and Virtual Cut through Switching (VCT) networks as shown in Fig 5. In Wormhole switching networks, only the header flit experiences latency. Other flits belonging to the same packet simply follow the path taken by the header flit. If the header flit is blocked then the entire packet is blocked. It does not require any buffering of the packet. Therefore, the size of the chip drastically reduces. However, the major drawback of this switching technique is a higher latency. Thus, it is not a suitable switching technique for real-time data transfers.

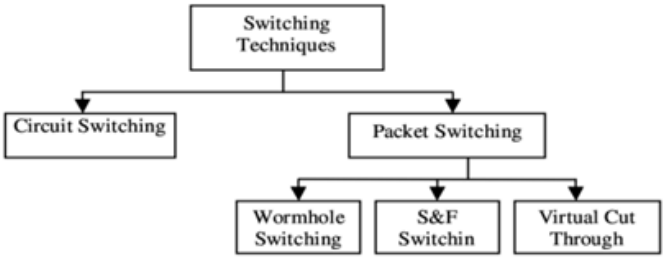

Fig 5. Switching Techniques

S&F switching forwards a packet only when there is enough space available in the receiving buffer to hold the entire packet. Thus, there is no need for dividing a packet into flits. This reduces the overhead, as it does not require circuits such as a flit builder, a flit decoder, a flit stripper and a flit sequencer. Store and forward is the easiest policy in terms of implementation complexity. So this implementation is based on store and forward switching.

5.2. Proposed Flow Control Mechanism

Flow control determines how network resources, such as channel bandwidth, buffer capacity, and control state, are allocated to a packet traversing the network. The flow control may be buffered or buffer less as shown in Fig 6. The Buffer less Flow Control has more latency and fewer throughputs than the Buffered Flow Control. The Buffered Flow Control can be classified further as:

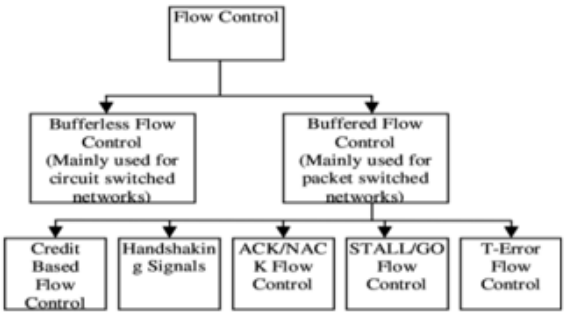

Fig 6. Flow Control Techniques

In Credit Based Flow Control, an upstream node keeps count of data transfers, and thus the available free slots are termed as credits. Once the transmitted data packet is either consumed or further transmitted, a credit is sent back and used [15, 16]. To minimize the chances of dropped packets at the receiving end, the credit based flow control mechanism has been incorporated wherein only those output IP blocks take part in the scheduling that has some credit. In addition to this, every input block maintains packet array and the linked list array to maintain the proper flow so as to avoid the out of order delivery.

5.3. Proposed buffer implementation in the design of router

A higher buffer capacity and a larger number of virtual channels in the buffer will reduce network contention, thereby reducing latency. However, buffers are area hungry, and their use needs to be carefully directed [17, 18] therefore proposed a simple implementation of a buffer architecture for NoC buffers using 180nm technology to estimate the cost and area of buffers needed for NoC. Also proposed the trade-off between buffer size and channel bandwidth to secure constant latency and concluded that increasing the channel bandwidth is preferable to reducing the latency in NoC.

The input block consists of six major components: the packet array, the linked list array, the destination head array, the destination tail array, the free-list FIFO, and a shift register. Four of these six components are conventional memory elements. A more area efficient implementation of memory is through the use of SRAM cells. Each SRAM cell is implemented using 6 transistors. SRAM design is carried out using full custom approach to ASIC design as shown in Fig 7.

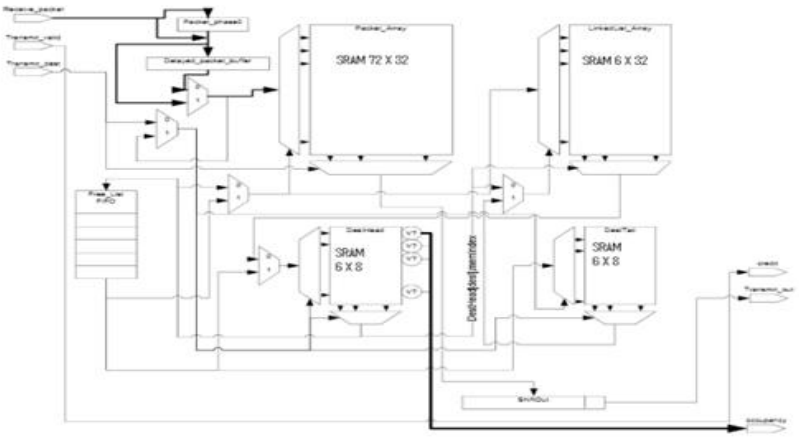

Fig 7. Input Module with SRAM based Arrays

5.4. Proposed Scheduler in the Design

The scheduler was modified using a folding approach due to the regular structure and placement of the arbiters. The modified scheduler is as shown in Fig 8. Each arbiter in the modified scheduler now has to generate both grant and accept signals in a time multiplexed fashion. The arbiter is modified to hold both grant and accept pointers for successive time slots. The proposed scheduler belongs to a Router in 2D Mesh NOC design. So here the value of N is 4.

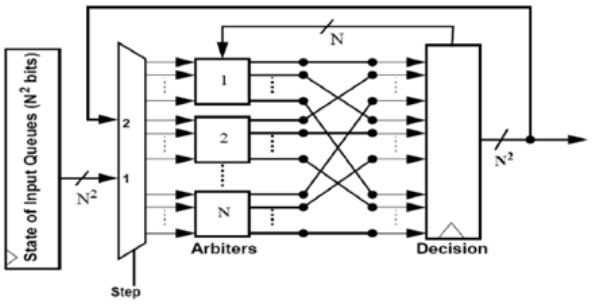

Fig 8. Modified Scheduler

VI. Introduction To Clock Gating

Clock gating [19-21] applies to synchronous load-enable registers, which are groups of flip-flops that share the same clock and synchronous control signals and that are inferred from the same HDL variable. Synchronous control signals include synchronous load-enable, synchronous set, synchronous reset, and synchronous toggle. The registers are implemented by Design Compiler by use of feedback loops. However, these registers maintain the same logic value through multiple cycles and unnecessarily use power. Clock gating saves power by eliminating the unnecessary activity associated with reloading register banks. Designs that benefit most from clock gating are those with low-throughput data paths. Designs that benefit less from RTL clock gating include designs with finite state machines or designs with throughput-of-one data paths.

Power Compiler allows performing clock gating with the following techniques [19]:

1. RTL-based clock gate insertion on unmapped registers. Clock gating occurs when the register bank size meets certain minimum width constraints.
2. Gate-level clock gate insertion on both unmapped and previously mapped registers. In this case, clock gating is also applied to objects such as IP cores that are already mapped.
3. Power-driven gate-level clock gate insertion, which allows for further power optimizations because all aspects of power savings, such as switching activity and the flip-flop types to which the registers are mapped, are considered.

Without clock gating, Design Compiler implements register banks by using a feedback loop and a multiplexer. When such registers maintain the same value through multiple cycles, they use power unnecessarily. Fig 9 shows a simple register bank implementation using a multiplexer and a feedback loop.

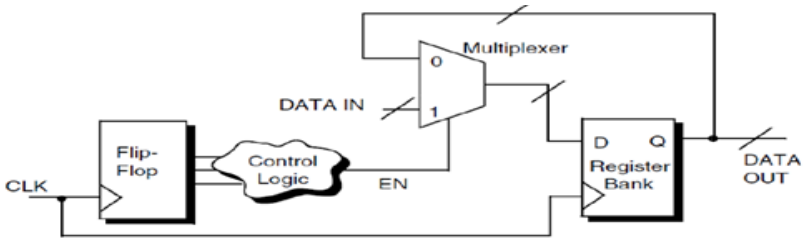

Fig 9. Synchronous Load-Enable Register With Multiplexer

The multiplexer also consumes power. Clock gating eliminates the feedback net and multiplexer shown in Fig 9 by inserting a 2-input gate in the clock net of the register. Clock gating can insert inverters or buffers to satisfy timing or clock waveform polarity requirements. The 2-input clock gate selectively prevents clock edges, thus preventing the gated-clock signal from clocking the gated register.

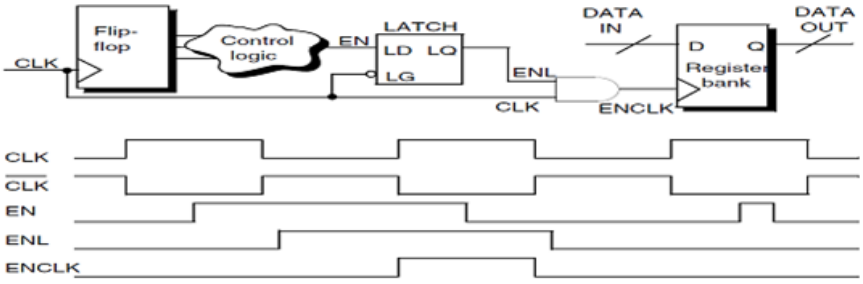

Fig 10 shows a latch-based clock-gating style using a 2-input AND gate, however, depending on the type of register and the gating style, gating can use NAND, OR, and NOR gates instead. Clock gating reduces the clock network power dissipation, relaxes the data path timing, and reduces routing congestion by eliminating feedback multiplexer loops. For designs that have large multi-bit registers, clock gating can save power and reduce the number of gates in the design. However, for smaller register banks, the overhead of adding logic to the clock tree might not compare favourably to the power saved by eliminating a few feedback nets and multiplexers.

VII. Conclusion

2D-Mesh has been an area efficient implementation of a routing node for an NOC is demonstrated. Of the four components of routing node, the input block (mainly consisting of buffers) and scheduler have been modified to save area requirements. The other two components of the routing node take up negligible area in comparison. The use of custom SRAM in place of synthesizable flip flops in the input block has resulted in a saving of over 25% of the silicon area and power optimization is 65% when operated at 16 ns clock.

Table 1. Comparative Results of Proposed Router with Existing Router Design

Network	Topo-logy	Flit Size in bits	Ports	Buf Size in flits	Tech in nm	L in Clk	A in Sq.mm	F in MHz
Teraflops	Mesh	32	4	16	65	5	0.34	4270
Xpipes	Custom	32	4	--	100	7	--	--
Dally	Torus	256	5	4	100	3	--	200 - 2000
HIBI	Bus	32	2	2,8	130	4	0.03 -0.05	435
Octagon	Ext. Ring	32	4	2,8	130	4	0.04 - 0.09	435
SPIN	Fat- T	16	8	8	130		0.24	200
Aethereal	Mesh	96	5	8	120		0.26	500
ANoC	Mesh	32			130		0.25	500
Mango	Mesh	32	5	1			0.19	795
Hermes	Mesh	32	5	2,8	130	10	0.05-0.11	435
SoCBus	Custom	16	3	1	180			
ASoC	Mesh	32	4	2	180		0.04-0.08	400
Avg.		**50.1**	**4.8**	**6.4**	**170**	**5.2**	**0.14-0.22**	**328-596**
Present Work	**Mesh**	**32**	**4**	**8**	**90**	**4**	**0.15**	**500**

Legends used in above Table:-, Buf.-Buffer, Tech-Technology, L-Latency, A-Area, F-Frequency, Ext-Extended, R-Ring, , T-Tree Cus- Custom, Avg-Average,Pre-Present. Clock gating is an important high-level technique for reducing the power consumption of a design. Clock gating reduces the clock network power dissipation, relaxes the data path timing, and reduces routing congestion by eliminating feedback multiplexer loops. For designs that have large multi-bit registers, clock gating can save power and reduce the number of gates in the design.

References

[1] K. Latif, T. Seceleanu and H. Tenhunen, "Power and Area Efficient Design of Network-on-Chip Router through Utilization of Ideal Buffer", Proc.17th IEEE International Conference and workshop on Engineering of Computer based System**, (2010).**

[2] L. Benini and G. De Micheli, "Networks on Chips, Morgan", Kaufmann Publishers, **(2006).**

[3] W. Hangsheng, L. S. Peh and S. Malik, "Power- driven design of router micro architectures in on-chip networks", Proc. of the 36th Annual IEEE/ACM International Symposium on Micro architecture (MI-CRO), **(2003)**, pp. 105-116**.**

[4] X. Chen and L. S. Peh, "Leakage power modeling and optimization of interconnection networks", Proc. of International Symposium on Low Power Electronics and Design, **(2003)**, pp. 90-95**.**

[5] N. Banerjee, P. Vellanki and K. S. Chatha, "A Power and Performance Model for Network-on-Chip Architectures", Proc. of the conference on Design, automation and test in Europe (DATE), vol. 2, **(2004)**, pp. 1250-1255.

[6] ARTISAN, "A comparison of network-on-chip and busses", white paper, **(2005).**

[7] W. J. Dally and B. Towles, "Principles and practices of interconnection networks", Morgan Kaufmann Publishers, **(2004).**

[8] T. Bartic, "Topology adaptive network-on-chip design and implementation", IEEE Proc. Comput. Digit Tech., vol. 152, no. 4, **(2005)** July, pp. 467–472.

[9] E. Beigne, "An asynchronous NOC architecture is providing low latency service and its multi-level design framework", in ASYNC, **(2005)** March, pp. 54–63.

[10] L. Benini and G. de Micheli, "Networks on chips: A new SoC paradigm", IEEE Computer, vol. 35, no. 1, **(2002)** January, pp. 70–78.

[11] J. Owens, et. al., "Research challenges for on-chip interconnection networks", IEEE Micro, vol. 27, no. 5, **(2007)** September-October, pp. 96–108.

[12] T. Bjerregaard and S. Mahadevan, "A survey of research and practices of network-on-chip", ACM Computing Surveys, vol. 38, no. 1, **(2006)** June, Article no. 1.

[13] T. Bjerregaard and J Sparose, "A router architecture for connection oriented service guarantees in the MANGO clockless network-on-chip", *in DATE,* vol. 2, **(2005)** March, pp. 1226-1231.

[14] C. Wang, et. al., "Area and power efficient innovative NoC Architecture", Proc. of 18thEuroMicro International Conference, PDP 2010, Pisa, Italy, **(2010).**

[15] E. Bolotin, I. Cidon, R. Ginosar and A. Kolodny, "QNoC: QoS architecture and design process for network on chip", Journal of Systems Architecture, vol. 50, Issue 2-3 (Special Issue on Network on Chip), **(2004)** February, pp. 105-128.

[16] E. Bolotin, I. Cidon, R. Ginosar and A. Kolodny, "Cost considerations in Network on Chip Integration", The VLSI Journal, no. 38, **(2004)**, pp. 19-42.

[17] H. Zimmer, S. Zink, T. Hollstein and M. Glesner, "Buffer-architecture exploration for routers in a hierarchical network-on-chip", Proc. 19th IEEE International Symposium on Parallel and Distributed Processing, **(2005)** April, pp. 1-4.

[18] E. Bolotin, A. Morgenshtein, I. Cidon, R. Ginosar and A. Kolodny, "Automatic hardware-efficient SoC integration by QoS Network-on-Chip", Proc. 11th International IEEE Conference on Electronics, Circuits and Systems, **(2004)**, pp. 479-482**.**

[19] Power Compiler User Guide, SYNOPSIS Version F 2011.9, **(2011)** December**.**

[20] M. Keating, "The Future of Low Power", SNUG, **(2007),**

[21] M. Keating, D. Flynn, R. Aitken, A. Gibbons and K. Shi, "Low Power Methodology Manual: For System-on-Chip Design", Springer, **(2007).**