2009 First International Conference on Computational Intelligence, Communication Systems and Networks

A Communication-Centric Embedded System Architecture (ACCESA)

Radha Guha
CSE Dept. of Amrita University, Coimbatore, India 641105

Abstract

In the era of application convergence, the embedded systems are required to support multiple applications with different traffic patterns and Quality of Service (QoS) requirements. But continual process technology's shrinking trend makes the communication network of the embedded system the performance bottleneck for its increased latency. To ameliorate this problem, a communication-centric embedded system architecture (ACCESA) is proposed in this paper which is modular and block based. The modular and block based architecture of ACCESA helps in performance improvement of the embedded system by means of flexible and efficient QoS provisions and power management. The pre-fabricated or pre-configured hybrid and hierarchical communication network of ACCESA saves on run-time reconfiguration time and on interconnect cost by preventing under utilization of communication resources without sacrificing on processing speed and throughput of the applications. This reduced and efficient resource usage in ACCESA in turn saves on power consumption of the chip.

1. Introduction

The Integrated Circuits (ICs) will soon accommodate billions of transistors running at 10 to 15 GHz speed by the end of this decade with 90 nm and smaller process technology [1]. Because of the enormous capacity of the ICs, a whole system of several heterogeneous processor cores such as soft microprocessor cores, digital signal processor (DSP) cores, application specific integrated circuit (ASIC) processor cores, thousands of reconfigurable processing elements (PEs) as hardware accelerator cores and memory blocks all will be packed in the same chip, increasing the architectural complexity of the chip.

Along with the increased architectural complexity of the IC chip, the application requirements are also changing. In the era of application convergence, multiple applications with different traffic patterns and QoS requirements will be mapped onto the same chip for concurrent executions. As for example, with the advancement of the wireless technology such as WiMAX and Wi-Fi etc., embedded systems in small hand-held battery powered portable devices will support standard voice functions of a telephone as well as text messaging, sending and receiving photos and videos, gaming, speech recognition and security provisions for sensitive data exchanges [2], [3], at the same time. This trend will increase the System on Chip (SoC) design complexity.

ACCESA architecture design is based on the current and future trend of different levels of communication concurrency requirements at different parts of the chip by multiple applications running concurrently on the same chip. The next sections describes the design principles behind ACCESA.

1.1. ACCESA Architecture

The accommodation of multiple functionalities on a single chip implies different communication load on different parts of the chip. As for example data transfer from memory to PEs and data transfer within the same task mapped onto a local group of PEs need more communication bandwidth than that of inter-tasks data transfer. Thus provision of a regular two dimensional grids of Network on Chip (NoC) means under utilization of interconnects.

Moreover due to the intense competition amongst the embedded systems developers, the success of a product depends on its shorter time to market requirement. These challenging design goals of the embedded systems can be met if the vast architectural exploration, verification and testing of the processing elements can be done easily and efficiently.

Thus a modular and block based flexible configuration architecture named a communication centric embedded system architecture (ACCESA) has been proposed in this paper for the reconfigurable FPGAs. This modular and block based architecture will facilitate partial block based reconfiguration at run-time, efficient resource management and power savings by advanced technologies like clock gating and voltage scaling etc. on a specific module of the chip.

Secondly a hybrid and hierarchical communication network has been proposed which can be pre-fabricated or pre-configured to reduce reconfiguration time of the chip at run time. The hybrid and

978-0-7695-3743-6/09 $25.00 © 2009 IEEE
DOI 10.1109/CICSYN.2009.26

hierarchical communication infrastructure supports different levels of concurrency differently and saves on interconnect cost and power consumption. This basic communication infrastructure can be modified during run-time partial reconfiguration as needed by the applications. This reduces the run-time reconfiguration cost of the chip.

The rest of the paper is organized as follows. Sub-section 1.2 introduces the resource management scheme in ACCESA configuration architecture. Section 2 presents the related work on NoC based communication architecture as proposed by other researchers. Section 3 proposes the communication infrastructure of ACCESA architecture. Section 4 shows the simulation result of power savings due to the hybrid and hierarchical communication architecture of ACCESA. Section 5 concludes the paper.

1.2. Resource Management in ACCESA

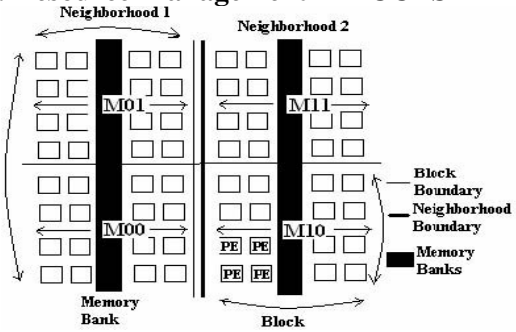

Figure 1. ACCESA Configuration Architecture

As shown in Figure 1, the ACCESA configuration architecture for FPGA is designed to be modular and block based, to facilitate efficient power management and customizable QoS provisions per module during run-time partial reconfiguration. In reference [4] their "Synchroscalar" architecture is column based, but here a block based configuration architecture has been designed for modern FPGAs like Xilinx's Virtex-4 and Virtex-5 etc. [5], [6], [7], [8], where block based partial reconfiguration is possible.

In each module there is a balance between the amount of logic elements and memory, so that the PEs in each block can sustain their speed by concurrent data access from on-chip memory blocks. As different applications may have different memory requirements, even if all the SRAM cells are not used as memories all the time, they can be used for implementing the control units or the finite state machines (FSMs) [9] of the processing elements (PEs) more efficiently. FSMs implemented with SRAMs have better power and area savings than the same implementation by Flip Flops (FFs) and programmable Look Up Tables (LUTs).

In Figure 1, a partial ACCESA configuration architecture is shown for a Xilinx's Virtex-4 XCVSX35 device [5], which is an array of 96*40 configurable logic blocks (CLBs) with maximum capacity of 240 Kb of distributed RAM or shift registers and total 3.4 Mb of block RAMs, where each block is 18 Kb and operating at 500 MHz. Block RAMs can be cascaded to form larger memory blocks. It is assumed that an array of 5*5 CLBs are required to implement a PE. Since ACCESA is coarse grained, the basic elements before configuration are PEs. During configuration, data-path widths of each PE or connecting PEs are changed for function specific optimization purposes.

In ACCESA architecture, a group of PEs is called a block (BL), and a group of blocks is called a neighborhood (NH). ACCESA architecture is modular and a hardware resource manager can keep track of all the resources in a hierarchical fashion, first by a neighborhood number, then by a block number in a neighborhood and finally by a PE number in each block. Suppose ACCESA has 8*8 array of PEs. First the PEs are grouped together as blocks and then as neighborhoods.

Suppose each block has 4*4 PE array and each neighborhood has 2*1 block array or 8*4 PE array. So the whole chip consists of 2 neighborhoods, 4 blocks and 64 individual PEs. The block boundaries are indicated by thin lines and the neighborhood boundaries are indicated by thick lines in Figure 1. This hierarchical resource organization, along with the hybrid and hierarchical communication infrastructure of ACCESA provide efficient resource and power management and flexible QoS provisions for various tasks with different communication bandwidth and latency requirements.

ACCESA's communication architecture is presented in Section 3. Before that the current approach taken by the researchers and commercial developers for a communication centric embedded system design is discussed in Section 2.

2. Related Work

Continual shrinking trend of the process technology makes it possible for the chip to operate at an increased frequency. Increased frequency of operation of the SoC makes communication network more critical in determining system performance [10], [11], [12], [13], [14] than the processing elements themselves. Thus the requirement of a communication centric architecture design has emerged.

The efficiency of a communication system depends on its latency, redundancy, flexibility, cost and

scalability. Because of the increased frequency of operation and narrower wire size, dedicated point to point and shared bus based communication systems are no longer suitable for their clock-skew problems for a FPGA chip where a large number of heterogeneous IP cores are integrated.

Apart from solving the clock-skew problems, multiple processors executing concurrently on the same chip demand increased concurrency of communications. But to reduce the cost of the chip, increased communication resource utilization is needed. Moreover in order to reduce the manual efforts and to shorten the time to market of the product there is a need for a communication architecture which can decouple [14] the communication protocols into transaction, transport and physical layers to adapt to the rapid changes of process technology and system architecture design differently. Buses do not support this decoupling of communication protocols effectively, what the NoC system can do effectively.

As communication network has become more critical than the processing elements, the SoC architecture is evolving towards a generic network of distributed routers connected with short length interconnecting wires. This is called Network on Chip (NoC) [10], [11], [12], [13], [14] and is the latest paradigm in embedded system design. Current researchers see it as a potential on-chip communication system.

In reference [14], the Arteris' NoC is the first commercial implementation of NoC. There the authors have compared a complete bus based system with a complete NoC based system in terms of frequency of operation, maximum throughput, minimum latency, area usage and power consumption. Their bus based system is hierarchical to avoid clock skew problem for a chip of size 9 mm^2. In all parameters of comparisons the NoC based system is found to be more efficient than the bus based system except for the area usage.

Other researchers who have proposed some deviations from the regular mesh architecture to provide application specific interconnects are as follows. In reference [15], the authors have proposed a design methodology for automating application specific NoC synthesis. In reference [16], the authors have proposed a NoC based Hierarchical Graph (HG) as a flexible communication architecture which can be modified for application specific need. HG employs less number of routers than mesh network and thus occupies less chip area and can be implemented easily. In reference [17], the authors have presented an application specific high-level scheduling and interconnect topology synthesis technique. Their interconnect topology provides low-hop links between processing elements to reduce power, latency and to avoid deadlock for the scheduling algorithm. In reference [18], the authors have prosed a hierarchical network dividing the global 2D network into smaller sub-networks and connecting them hierarchically.

In comparison to the above mentioned research work for performance improvement of the communication network of the embedded systems, ACCESA communication architecture is a hybrid system using best of point to point, bus and NoC communication standards. For a small area, if clock skew is not a problem, point to point or bus is still used. Also as in a local area, more communication concurrency is required than the global area a hierarchical mesh network is used in providing more concurrency in a local area than in the global area to reduce interconnect cost. The details of ACCESA communication network is described next.

3. ACCESA Communication Architecture

Instead of a monolithic two dimensional grids of NoC, ACCESA communication infrastructure is hybrid and hierarchical in nature where the motto is to provide locally dedicated and globally shared interconnections. The hybrid and hierarchical communication architecture in ACCESA incorporates the best of point to point, bus and NoC communication standards to create a communication centric embedded system design. By having the task mapping strategy to confine a task within a block of adjacent PEs, the traffic pattern can be controlled. As a task will have more data communications within itself, a dedicated point-to-point connection within the PEs in a block is provided. Whereas shared communication paths can be provided for inter-block communications.

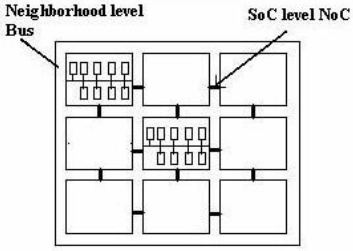

Figure 2. Option 1: Neighborhood Level Bus

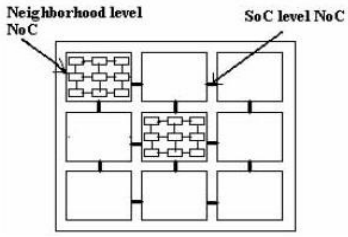

Figure 3. Option 2: Neighborhood Level NoC

As bus based system uses less resources than NoC, bus can be used in a small area to connect a few blocks in a neighborhood, if the bus length is within the limit of clock-skew problems, as shown in Figure 2 as Option 1. In Option 1 the neighborhoods are connected with a 2D NoC.

Alternatively if more concurrency is needed then a two dimensional grids NoC can be provided in a neighborhood to connect a few blocks. In that case there are two levels of NoC, one at neighborhood level and another at SoC or chip level providing different levels of concurrency as shown in Figure3 as Option 2.

The monolithic NoC architecture is not used for the entire chip for some major disadvantages of NoC. First major disadvantage of NoC is its higher silicon usage than the bus system. Whereas in bus based communication system there is one central arbiter, in NoC based communication system every processor is connected with a router. As the number of processors in the chip increases the arbitration logic in each router takes more silicon area. Secondly when traffic load varies from one part of the chip to another, providing a regular two dimensional grid network means underutilization of resources.

Thirdly NoC employs packet switching mechanism, where data is decomposed into packets or flits and the head flits contain the destination address. The data first has to be decomposed into packets at the producer end and then again reassembled at the receiver end. For large amount of data transfer between IP blocks, such as from memory to PE this mechanism takes more time. This problem can be ameliorated by adding local block memory for a group of PEs so that this data does not have to travel the NoC. A dedicated route with circuit switch option can be provided where a path between the producer and the consumer can be formed for the entire length of data transmission time.

Because of the above mentioned disadvantages of NoC, a hybrid and hierarchical communication network has been proposed in this paper to save on communication resources. How much reduction of communication interconnects is possible without slowing down the operating speed of the processors or reducing the throughput of the applications is analyzed next.

When a sub-NoC system is used in a neighborhood, communication resources can still be saved because of the hierarchical nature of the NoC in the entire chip. As for example if 81 IP blocks, such as processing elements and memories were connected in one level NoC of 9*9 two dimensional grid, there would be 144 interconnects. Instead if 81 IP blocks are connected in two levels of NoC as shown in Figure 3, then there will be 9 clusters and in each cluster there will be 9 IP blocks. In each cluster the IP blocks are connected by first level of NoC of 3*3 two dimensional grid. The 9 clusters in the chip are connected by the second level of NoC of 3*3 two dimensional grid. In this hierarchical NoC there are 12 interconnects in each cluster, 12 interconnects in second level NoC in the chip and thus total 12*9+12=120 interconnects in the entire SoC. As the number of IP blocks, which is 81 in this example, is much less than the total number of interconnects which is 120, communication concurrency is not suffered much in comparison to one level NoC with 144 interconnects for the entire chip. Within a cluster, communication concurrency is still maximum and is same as one level of NoC.

It is obvious that bus system in a neighborhood will use less communication resources than NoC system in a neighborhood. Communication resources can be saved by using bus system for a small group of IP blocks without sacrificing much on operating speed and throughput as analyzed here. In this analysis, the SoC is assumed to be implemented with 90 nm process technology. For 90 nm process technology the gate delay is 60 ps (pico seconds) and D Flip Flop (DFF) traversal time which includes setup and hold time is 0.3 ns (nano seconds) [14]. It is assumed 3 or fewer gate stages are needed for decision logic to latch the data to bus for up to 9 PEs connected to the same bus. Clock skew of .3 ns is assumed when the bus is up to 3 mm long. Propagation delay of data for a properly buffered wire of length 2 mm is assumed to be .44 ns and for wire of length 3 mm is assumed to be .66 ns.

Thus transport time across 2 mm wire including the clock-skew and DFF traversal time takes .44 + 3*.06 + .3 +.3 = 1.22 ns. So the maximum operating frequency of 2 mm wire is, $f = 1/1.22 \approx 800$ MHz. Similarly, transport time across 3 mm wire including the clock-skew and DFF traversal time takes .66 + 3*.06 + .3 + .3 = 1.44 ns. So the maximum operating frequency of 3 mm wire is, $f = 1/1.44 \approx 690$ MHz.

The two communication infrastructures, Option 1 and Option 2, as shown in Figure 2 and Figure 3 are compared in terms of operating frequency, throughput and resource usage by varying the number of neighborhoods (N) in the entire chip, number of blocks (B) per neighborhood and number of PEs (P) in a block for a given number of PEs in a chip. Depending on the operating frequency and throughput requirement of an application the best architecture which can save on communication resource usage can be chosen.

Option 1:

The communication infrastructure of Option 1 as shown in Figure 2 is as follows:

(a) PEs in a block are connected by point-to-point connections (not shown in Figure 2).
(b) Blocks in a neighborhood are connected by bus.
(c)Neighborhoods in a chip are connected by a 2 D mesh NoC.

Total interconnect length, L in the whole chip in this case can be formulated as Equation (1):

L=B*N[(P-1)*{1+(P-1)}/2]+[Bus_Length*N]+ [Level2_interconnect*(√N-1)*√N*2],

where Bus_Length = l, N is the number of neighborhoods in the entire chip, B is the number of blocks per neighborhood and P is the number of PEs in a block.

Bus length of 2 mm is assumed for connecting 4 PEs and bus length of 3 mm is assumed for connecting 9 PEs. Interconnect length of 2 mm is assumed for connecting the neighborhoods in a SoC. The higher interconnect length between the bus and the second level NoC will determine the frequency of operation in a system.

It has already been calculated that the operating frequency of 2 mm bus is 800 MHz and 3 mm bus is 690 MHz. The peak throughput, T, of the system will be determined by the number of buses and operating frequency f of the system and can be formulated as T = 4*N*f where number of buses is equal to the number of neighborhoods, N and 4 bytes wide data transfer is assumed in ACCESA architecture.

Option 2

The communication infrastructure of Option 2 as shown in Figure 3 is as follows:
(a) PEs in a block are connected by point-to-point connections (not shown in Figure 3).
(b) Blocks in a neighborhood are connected by first level 2 D mesh NoC.
(c)Neighborhoods in a chip are connected by second level 2 D mesh NoC.

Total interconnect length, L in the whole chip in this case can be formulated as Equation (2):

L=B*N[(P-1)*{1+(P-1)}/2]+ [Level1_interconnect_length*N*(√B-1)*√B*2]+ [Level2_interconnect_length*(√N-1)*√N*2],

where l is the higher of the Level1_interconnect_length and Level2_interconnect_length, N is the number of neighborhoods in the entire chip, B is the number of blocks per neighborhood and P is the number of PEs in a block as before.

The interconnect length in the first level NoC is assumed to be 1 mm whereas in second level NoC it is 2 mm. The higher interconnect length between the first level NoC and second level NoC will determine the frequency of operation in the system.

It has already been calculated that the operating frequency of 2 mm wire is 800 MHz. Thus the peak throughput, T, of the system will be determined by the size of the NoC and the operating frequency f of the system and can be formulated as T = 4*N*f where the NoC size is equal to the number of neighborhood, N and 4 bytes wide data transfer is assumed as before.

Table 1.Option 1 vs. Option 2 Communication Cost

# PEs	# NHs	# BL/NHB	# PE/BLP	Option 1				Option 2			
				l mm	f MHz	T GB/Sec	L mm	l mm	f MHz	T GB/Sec	L mm
64	8	4	2	2	800	25.6	72	2	800	25.6	98
64	4	4	4	2	800	12.8	112	2	800	12.8	120
128	16	4	2	2	800	51.2	144	2	800	51.2	240
128	8	4	4	2	800	25.6	232	2	800	25.6	280
128	4	8	4	3	690	11	208	2	800	12.8	296

Table 1 compares two different options of connecting the PEs in different size chips. Table 1 shows what frequency and throughput they can support and what will be the corresponding interconnect cost. As seen from Table 1, Option 1 of connecting the PEs in a chip uses much less resources than that of Option 2 for all five cases. First 4 cases do not sacrifice on the system support of operating frequency and throughput for a small size bus of 2mm. Only when the bus length is 3 mm in a neighborhood as in the last case of Table 1, the throughput of the application is little less in Option 1 than in Option 2. Communication resource savings also implies power savings of the chip as analyzed in Section 4. Thus bus can be used for a small number of PEs in a local area.

4. Power Savings in ACCESA

Total dynamic power [19] consumed by a chip is due to charging and discharging of its various components like logic elements (LE), RAM, DSP, phase locked loops (PLL), clocking and routing. Total dynamic power can be calculated by multiplying the supply voltage VCCINT by total sum of currents, ICCINT passing through all the above mentioned components of the chip.

Dynamic power = VCCINT × ΣICCINT (LE, RAM, DSP, PLL, Clocks, Routing). Thus dynamic power consumption of the chip is directly proportional to the length of the routing. Routing's share [20] of power consumption is maximum (60%) among all other components of dynamic power.

Communication-centric architecture of ACCESA, reduces the power consumption of the chip by reducing the routing length without sacrificing on processor

speed and throughput. Simulation results of SystemC model of ACCESA architecture shows the difference of power consumption between Option 1 and Option 2 communication network as shown in Figure 4. All other components like LE, RAM etc. remaining the same, power consumption of Option 1 communication network is much lower than the power consumption of Option 2 communication network for all 5 different configurations C1, C2, C3, C4, C5 (Table 1) of PEs in Blocks(B) and Neighborhoods(NH). As total routing length of Option 1 is considerably less than routing length of Option 2 in ACCESA's hybrid and hierarchical communication network this power savings is observed for Option 1.

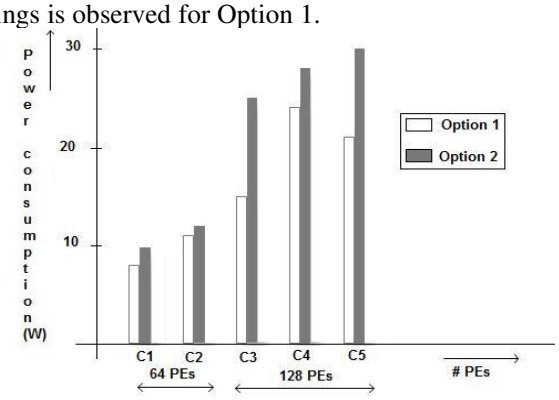

Figure 4. Power Comparison, Option 1 vs. Option 2

5. Conclusions

In this paper a modular and block based application configuration architecture named ACCESA has been proposed which will facilitate efficient resource and power management during the partial reconfiguration of the chip at run time. A pre-fabricated or pre-configured, hybrid and hierarchical communication infrastructure has been proposed for ACCESA architecture to support different levels of required concurrency differently and to reduce interconnect cost, power consumption and reconfiguration time of the chip. This basic communication architecture can be modified during run-time partial reconfiguration of the chip according to applications' requirements.

References

[1] ITRS, "International Technology Road map for Semiconductors", 2005 Edition.

[2] P. Heysters, G. Smit and E. Molenkamp. "A Flexible and Energy-Efficient Coarse-Grained Reconfigurable Architecture for Mobile Systems", *The Journal of Supercomputing, Vol. 26, Issue 3*, November 2003, PP. 283-308.

[3] J. Chen, J.Yeh, Y. Lan, L. Lin, F. Chen and S. Hung, "RAMP: Reconfigurable Architecture and Mobility Platform", In *Proceedings of IEEE Globecom, Vol. 6,* 2005, PP. 3564-3569.

[4] J. Oliver, R. Rao, P. Sultana, J. Crandall , E. Czernikowski, L. W. Jones IV, D. Franklin , V. Akella, and F. T. Chong, "Synchroscalar: A Multiple Clock Domain, Power-Aware, Tile-Based Embedded Processor", *2004 International Symposium on Computer Architecture,* June 2004, PP. 150-161.

[5] Xilinx, "Virtex-4 User Guide UG070 (v2.2)", April 10, 2007.

[6] Xilinx, "Virtex-5 User Guide, UG190 (v3.0)", February 2, 2007.

[7] Xilinx, "Virtex-5 Family Overview, LX, LXT and SXT Platforms, DS100 (V3.2)", September 4, 2007.

[8] ALTERA, "Stratix III TriMatrix Memory", *www.altera.com*, October 2007.

[9] A. Tiwari and K. A. Tomko, "Saving Power by Mapping Finite-State Machines into Embedded Memory Blocks in FPGAs", In *Proceedings Design, Automation and Test in Europe Conference and Exhibition, 2004, Vol. 2,* February 2004, PP. 916 –921.

[10] P. Guerrier and A. Greiner, "Generic Architecture for On-Chip Packet-Switched Interconnections", In *Proceedings of Design Automation and Test in Europe (DATE)*, March 2000, PP. 250-256.

[11] S. Kumar et al, "A Network on Chip Architecture and Design Methodology", In *Proceeding of International Symposium on VLSI (ISVLSI),* PP. 117-124, 2002.

[12] R. Marculescu, "Energy, Fault-Tolerance and Scalability Issues in Designing Networks-On-Chip", *www.gigascale.org/pubs/440.html,* February 2004.

[13] 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,* 2004, PP. 105–128.

[14] Arteris, "A comparison of Network-on-Chip and Busses", *www.arteris.com.* October 2007.

[15] D. Bertozzi, A. Jalabert, S. Murali, R. Tamhankar, S. Stergiou, L. Benini, G. Micheli, "NoC Synthesis Flow for Customized Domain Specific Multiprocessor Systems-on-Chip", *IEEE Transaction on Parallel and Distributed Systems, Vol. 16, No. 2,* February 2005.

[16] A. Vahdatpour1, A. Tavakoli and M. Falaki1, "Hierarchical Graph: A New Cost Effective Architecture for Network on Chip", *EUC 2005, LNCS 3824,* 2005, PP. 311–320.

[17] N. Bambha and S. Bhattacharyya, "Joint Application Mapping/Interconnect Synthesis Techniques for Embedded Chip-Scale Multiprocessors", *IEEE Transactions on Parallel and Distributed Systems, Vol. 16, Issue 2,* February 2005*,* PP. 99 – 112.

[18] S. Bourduas, Z. Zilic, "A Hybrid Ring/Mesh Interconnect for Network-on-Chip Using Hierarchical Rings for Global Routing", *NOCS 2007, Vol. 7, Issue 9*, May 2007, PP 195-204.

[19] ALTERA, "Power Overview", *www.altera.com*, February 2007.

[20] L. Shang, A. S. Kaviani, K. Bathala, "Dynamic Power Consumption in Virtex-II FPGA Family", *ACM*, February 2002.