Evaluation and Design Trade-Offs Between Circuit-Switched and Packet-Switched NOCs for Application-Specific SOCs

Kuei-Chung Chang, Jih-Sheng Shen, Tien-Fu Chen
Dept. of Computer Science and Information Engineering
National Chung Cheng University, Chia-Yi, Taiwan
{ckj,sjs92,chen}@cs.ccu.edu.tw

ABSTRACT

NOC architectures have to deliver good latency-throughput performance in the face of very tight power and area budgets. However, the latency and the power consumption for transferring information down the transmitter stack, through the channel, and up the receiver stack might be unacceptably high. In this paper, we evaluate the designs of packet-switched and the proposed circuit-switched NOC in detail, and we advocate using circuit-switched NOC as it is more attractive for application-specific SOC designs because of communication localization. We implement and synthesize the designs of packet-switched and circuit-switched NOCs, and we take multimedia applications as our case studies. The experimental results show that the area, latency and the energy consumption of the packet-switched NOC are much larger than that of the circuit-switched NOC for application-specific designs.

Categories and Subject Descriptors

B.4.3 [**INPUT/OUTPUT AND DATA COMMUNICATIONS**]: Interconnections—*Topology*

General Terms

Design, Experimentation

Keywords

Low-Power, SOC, Application-specific designs, NOC

1. INTRODUCTION

As technology scaling enables the integration of billions of transistors on a chip, economies of scale are prompting the move toward parallel chip architectures with application-specific system-on-chip (SOC) leveraging multiple processing cores on a single chip for better performance at manageable design costs [2, 10]. As these parallel chip architectures scale in size, on-chip networks have becoming the main communication architecture, replacing dedicated interconnections and shared buses. NOC architectures have to deliver good latency-throughput performance in the face of very tight power and area budgets. These trends make on-chip network design to be one of the most challenging and significant design problems.

The packet-switched NOCs resemble the interconnection architecture of high-performance parallel computing systems. However, it is still unclear if such a full blown protocol stack will be practical, since the latency for transferring information through the transmitter stack, the channel, and the receiver stack might be unacceptable high. In addition to the latency of the protocol stack, the packet-switched NOC structure consists of many additional resources, such as queuing buffer and packet format processing units, which consume too much energy for transmitting one packet. A power constrained environment completely changes the approach to processor design. Rather than looking for the fastest implementations, one has to look for the most energy efficient implementations, since the highest performance implementation dissipates too much power [8, 9].

Since different architectures are efficient for different classes of applications, machine of the future will contain a number of different execution cores, each being efficient for a class of applications [10]. Application specific is always a tradeoff among competing design goals. The search for flexibility in design without paying a significant area, time, power cost is the primary problem for application-specific SOC designs.

The main contribution of the paper is the detailed comparative evaluations of recently proposed interconnection architectures including segmented bus, packet switching NOC, and circuit switching NOC with realistic case studies. None of existing works on NOCs has compared these interconnection architectures relative to implementation cost, latency, and energy. In this paper, we claim that if the scale of the SOC is not too large (under few tens of processing units), the proposed circuit-switched crossroad interconnection architecture is suitable for application-specific SOC designs.

The rest of the paper is organized as follows. Section 2 summarizes the related work and gives detailed comparisons between different interconnection desings. We will describe the characteristics of application-specific SOC designs in Section 3. The features and the constraints of packet-switched NOC are discussed in Section 4. Section 5 presents the proposed circuit-switched NOC design. We give the detailed evaluation results between several interconnection implementation methods in Section 6. Finally, we summarize our findings in Section 7.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
DAC 2006, July 24–28, 2006, San Francisco, California, USA.
Copyright 2006 ACM 1-59593-381-6/06/0007 ...$5.00.

2. RELATED WORK

Table 1: The characteristics of four interconnection structures under few tens of cores

	Shared Bus	Segmented Bus	CS NOC	PS NOC
Impl. Complexity	L	M/H	M	H
Average Power	M/H	L/M	L/M	H
Agerage Latency	M/H	M/H	M/H	H
Throughput	L	L/M	H	H
Overhead	L	M	M	H

H:High,M:Medium,L:Low,CS:Circuit-switched,PS:Packet-switched

In general, four widely-used interconnection architectures have been developed including conventional shared bus, segmented bus, networks on chip and router based communication architectures [19]. The brief comparisons of implementation complexity, power, latency, throughput, and the overhead between these four architectures are shown in Table 1. We discuss the characteristics of these interconnection architectures as follows.

Most current SOCs use the simplest shared-medium buses as their interconnection structures. In this architecture, all communication devices share the transmission medium. Only one device can drive the network at a time. The advantages of shared-medium buses include simple topology, low area cost, and extensibility. The disadvantages of shared buses are large load, large latency, large power consumption, and low bandwidth when the number of cores connecting to the shared bus becomes large.

To improve the timing and energy consumption of the long bus, the bus can be partitioned into two bus segments. Bus splitting [5, 11, 18] involves splitting a monolithic bus into multiple segments, with dual port drivers connecting adjacent segments. Advantages of bus splitting [11] are smaller parasitic load, larger timing slack, smaller driver size, low power consumption, and lower noise problems. Experimental results in [11] showed that the energy saving of the segmented bus compared to the shared bus architecture varies from 16% to 50%, depending on the characteristics of the data transfer among the modules and the configuration of the segmented bus. However, it is not a point-to-point connection; when a module connects to another module in different segments, it still consuming unnecessary power. At the same time the bus segments along the communication path will be occupied by the transaction, and the performance will be slowed down. In order to control the communication path, it consists of a global controller to dynamically control CMOS switches to be turned on/off.

The early works in [2, 7] pointed out the need for more scalable architectures for on-chip communication and, therefore, to progressively replace shared buses and split buses with on-chip networks. One of the scalable architectures is to use a router based on-chip interconnection architecture [15]. In theory, the router could be a fully connected crossbar, though it may be optimized by exploiting the fact that not every component on the chip will need to talk to every other component. The router-based communication architecture can potentially lead to significant energy savings. However, the load on the input port is the total of the wire capacitance and the sum of all input capacitances of N switches. The bit energy will increase linearly with the number of input and output ports N. The power consumption and the design complexity will be very high for switch fabrics with large number of ports.

To overcome the above-mentioned problems, several research groups have advocated the use of a communication-centric approach to integrate IPs in complex SOCs [2, 7, 15]. In a network-centric approach, the communication between IPs can take place in the form of packets. The common characteristic of these kinds of architectures is that the functional IP blocks communicate with each other with the help of intelligent switches. The switch consists of packet processing units (ingress and egress), an arbitration unit, and a switch fabric, and it provides a robust data transfer medium for the functional IP modules. However, it is still unclear if such a full blown protocol stack will be practical, since the latency and the energy for transferring information through the transmitter stack, the channel, and the receiver stack might be unacceptable high.

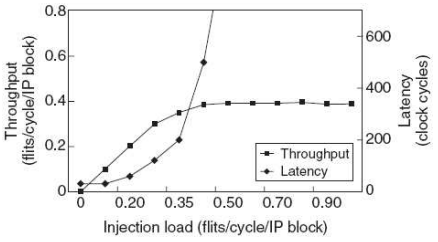
3. APPLICATION-SPECIFIC SOC DESIGN FEATURES

For many SOC applications, the system specifications are well defined and include real-time delay constraints. In addition, the application permits processors to specialize, handling a particular function. Custom-built application-specific interconnection architectures are another promising solution. Various trade-offs regarding latency, throughput, reliability, energy dissipation, and silicon area requirements characterize such communication-centric interconnection fabrics. The main characteristics of application-specific SOC designs are discussed as follows.

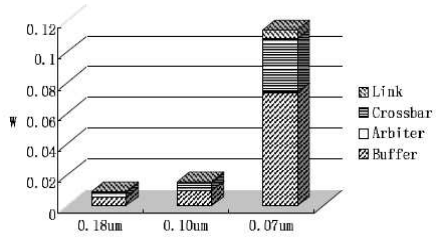
- **Customized communication topologies:**
  An application's nature will dictate the selection of a specific template for the communication medium. NOC-based interconnection performance strongly correlates with the topology selected for implementation. These topologies fall broadly into two categories: regular architectures [1, 16] and irregular [3, 12], application-specific NOC structures. In irregular architectures, the service requirements vary widely for the different processors and storage blocks. In the case of custom-built NOC architectures, switch blocks might not be identical; their design and placement depend on the specific communication requirements. Irregular network architectures might be necessary for realizing application-specific SOCs, such as those in mobile-phone systems, where different heterogeneous blocks with varying communication requirements must be linked.
- **Communication localization:**
  In a realistic SOC environment, different functions will be mapped to different parts of the SOC, and the traffic will exhibit highly localized patterns. Localization makes the shorter communication distance, and it takes less latency and energy for transmitting a message [3, 4]. For example, a localization factor of 0.3 signifies that 30% of the traffic generated by a core is local. As a result of this traffic localization, a system's throughput can be enhanced considerably, making it possible to transfer more data without saturating the network.
- **Parallel communications:**
  Traffic localization makes the whole network to be separated into several working regions. Increasing the amount of traffic localization causes more messages to

be injected without increasing the average energy dissipation. This happens because, on average, messages traverse fewer hops when there is greater localization. In this case, several masters can communicate to their slaves at the same time within different regions. This feature is suitable for multiprocessor or multithreading, where each processor or thread can work in its local regions and can communicate to other regions by coordinating NOC routers.

4. THE CHARACTERISTICS OF PACKET-SWITCHED NOCS



(a) NOC throughput and latency characteristics



(b) NOC power consumption of each part in the router

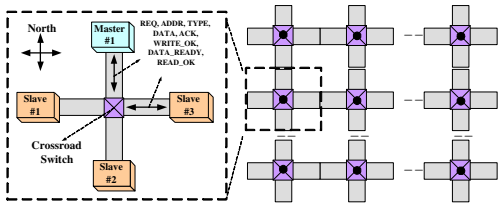
Figure 1: The characteristics for the packet-switched NOCs(ref. [6, 17]).

Several implementation issues related to packet-switched NOC chips are discussed in [13, 14], including circuit design, topology selection, packet format, and protocol. In general, a packet-switched router consists of four parts including the ingress packet process unit, the egress packet process unit, the arbitration unit, and the switch fabrics. The ingress packet process unit parallelizes the serial data flow on the transmission line into bus dataflow, and inspects the header and the content of the incoming packets. The egress process unit re-assembles the processed packets and delivers the packets to their destination ports. The arbitration unit determines when and where a packet should be routed from the ingress port to the egress port. It uses packet information from the ingress inspection and makes the decision based on arbitration algorithms. The switch fabric circuit is an interconnection network that connects the ingress ports to the egress ports.

When there are two or more packets in the ingress ports requesting the same destination port at the same time, destination contention will occur. In addition to the destination contention, inside switch fabric circuits, the same interconnect link may be shared by packets with different destinations. When contention occurs between packets, internal buffers are needed to temporarily store the packets with lower priorities. In switch fabric circuits, the buffers are normally implemented with shared SRAM or DRAM memories. The energy consumption in buffers comes from the data access energy, and the refreshing energy, consumed by the memory refreshing operation (in the case of DRAM).

The constraints of the packet-switched NOC are shown in Figure 1 including large energy consumption (queuing buffer, protocol stack processing units) and large delay for passing a message through a router. According to the Figure 1(a), throughput and latency characteristics for a typical NOC appear as a function of the injection load [6]. When the injection load approaches the throughput saturation limit, latency starts to increase exponentially. Figure 1(b) shows that the buffer consumes approximately 64% percent leakage power of the total node (router+link) for all process technologies, standing as the largest leakage power consumer [17]. It shows that as technology scales, leakage power becomes increasingly significant of the total power. The same results are also shown in [13], and it shows that the queuing buffer takes significant power of the total power in a mesh topology.

In addition to the significant energy consumption, in practical implementation, NOCs still suffer from long end-to-end latency [13]. Synchronization overhead in the mesochronous communication together with arbitration delay in all the switches produces this serious latency, which limits the packet-switched NOC's application to latency-insensitive systems such as multimedia processing systems.



(a) Crossroad communication block (CCB) (b) Crossroad interconnection architecture

Figure 2: The crossroad interconnection architecture and the CCB

5. THE PROPOSED CIRCUIT-SWITCHED CROSSROAD INTERCONNECTION ARCHITECTURE

In this paper, we propose a circuit-switched NOC design, named *Crossroad Interconnection Architecture*. The key idea of the crossroad interconnection architecture is to partition all cores on a chip as a well-organized 2-D irregular topology, and it uses crossroad switches to dynamically control active paths for those connections needed between a pair of cores. In addition to power optimization, the proposed crossroad interconnection architecture also gives better performance and parallel communications by providing several separated virtual bus segments at the same time.

The basic communication element of the architecture is the *crossroad communication block (CCB)* as shown in Figure 2(a). The *CCB* comprises a crossroad switch with a four-direction arbiter, circuit-switched fabrics and four communication links for data transmissions. More than one route can go through the switch at the same time if there is no contention. In addition, the path can connect cores by wrappers (implemented with OCP), which should take care of several design issues, such as synchronization, clocking and adop-

tion of data parameters. In this case, users can construct different NOC topologies based on the requirements such as power and performance. The crossroad interconnection architecture is shown as Figure 2(b). We can combine several *CCBs* to construct a large-scale interconnection network.

5.1 The crossroad switch design

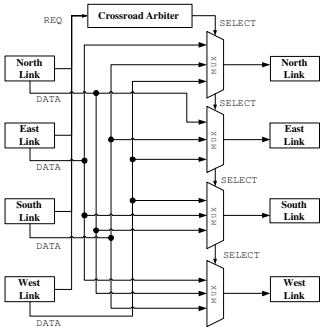


Figure 3: The simple design of the crossroad switch

The simple design block of the crossroad switch is shown in Figure 3, which consists of circuit-switched fabrics (fully-connected network) and an arbitration controller. We fixed the number of ports for inputs and outputs as four. The fabrics do the actual switching while the arbiter decides what direction to route incoming data through the fabrics. More than one path can be constructed by the crossroad switch as long as no contention between these paths. In order to keep low latency and high bandwidth in the network, the switch uses a circuit switching strategy that sets up an entire route and then sends the data as quickly as possible along the route. The route is then released immediately when the transaction has completed.

5.2 Bus Control and Arbitration

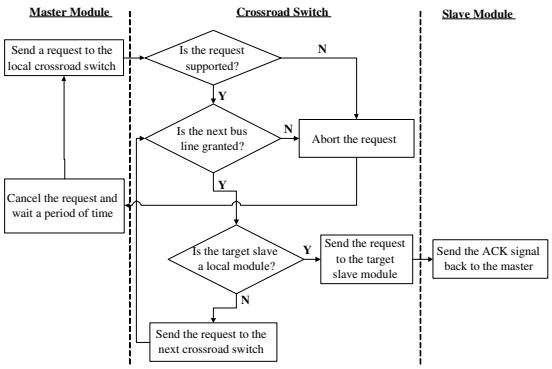


Figure 4: The flow of occupying the permission for a communication segment

Master cores are active parties in bus transactions, requesting services from slave cores - passive parties - in these transactions. The duty of the crossroad arbiter is to handle not only the communications between masters and slaves at the same CCB but also the communications between different CCBs. The flow of occupying the permission before any transaction is shown in Figure 4. If the target core is in the same CCB, the crossroad arbiter will immediately grant the master core to send requests to the target core when the target path is free. However, if the target core is in another CCB, the local crossroad arbiter will send requests to the next block, in which the crossroad switch will recognize these requests again until the master's requests arrive the target core. If one of the occupation is not granted to the master core, it will abort the transaction, and then try again later.

5.3 Features for application-specific designs

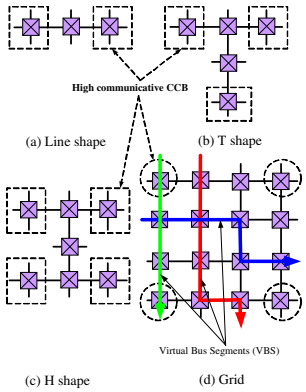
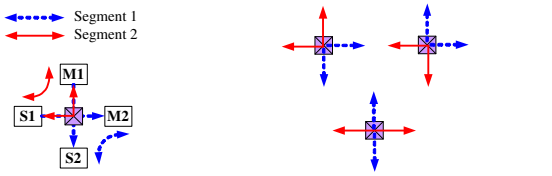


Figure 5: Different interconnection topologies

Our proposed crossroad interconnection architecture has several characteristics suitable for application-specific SOC designs, which will be described as follows.

- **Fully Configurable:**
  The proposed design can easily combine CCBs to make up our irregular network topologies according to the different application characteristics, as shown in Figure 5. In Figure 5, four example topologies are line shape, T-shape, H-shape and grid. A topology construction algorithm can be employed to explore the best solutions of the intercommunications for the purposes of high performance, low power consumption and low cost [4].
- **Power Optimization by Localization:**
  In our design, only one sender, one receiver and one or more than two switches are involved at each data communication, data exchanges among devices will result in minimum power consumption. Due to the full programmability, the topology mapping of cores will be optimized by profiling the communication traffic characteristics of applications [4]. We can cluster high communicative cores into several CCBs to provide separate high communicative CCBs. In this case, it can save more energy consumption due to the localization.
- **Better Communication Parallelism:**
  We design an "overpass" mechanism that can construct multiple segments to let different master cores communicate with their target slave cores at the same time, as shown in Figure 6(a). Figure 6(b) shows three possible combinations of parallel communications in a CCB. If those master-slave pairs are different, they may use different channels to communicate in the crossroad interconnection architecture at the same time.



(a) Two parallel communication segments (b) Three possible combinations of parallel segments

Figure 6: Parallel communications

6. EXPERIMENTAL RESULTS

In this section, we show the detailed experimental results of the circuit-switched and packet-switched NOCs. We use Modelsim to verify the correctness of the RTL designs of the segmented bus, packet-switched NOC, and the proposed circuit-switched NOC. Then, we utilize Synopsys design analyzer to synthesize the crossroad interconnection components and get the netlist schematic file. We use Nanosim to experiment with our low-level simulation and power consumption.

In design area, we simulate the crossroad architectures connecting more than one switch. The crossroad interconnection architecture adopts circuit switching mechanism, so the analysis of delay is very important. To demonstrate feasibility of the circuit-switched NOC architecture, the test chip is implemented using 0.18-μm technology, and then evaluate the delay constraints. Figure 7 shows the 5-switch NOC die photo and characteristics connecting with line-shape topology. The left-hand side of Figure 9 shows the cell area and latency characteristics of the designed crossroad interconnection topologies connecting with different number of crossroad switches. We can see that the increased area and latency will not be significant when the scale of the architecture becomes large.

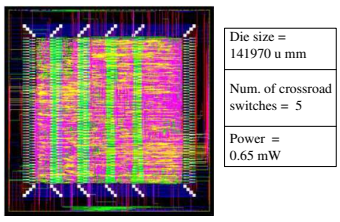
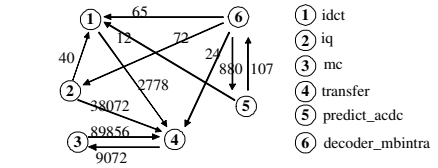


Figure 7: Die photo and characteristics of the design

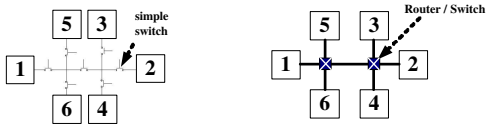
Table 2: Experimental results for different interconnection architectures

	Seg. Bus	CS NOC	PS NOC
Area (μm^2)	180.57 $\times 10^3$	56.26 $\times 10^3$	649.27 $\times 10^3$
Power (μW)	79.11	260.6	11793.69
Latency(ns)/switch	3.23	3.48	29.66
Performance(10^6 ns)	5.51	2.16	12.04

We take the MPEG-4 decoder as the case studies, and we collect 15 frames' cache access information as our simulated workloads. These frames include one I-frame and fourteen P-frames. Figure 8 shows the profiled core flow graph of MPEG-4 application and the experimental interconnection architectures with power-aware mapping [4], including the segmented bus, the packet-switched NOC, and the circuit-



(a) The core flow graph of the MPEG-4 decoder



(b) Segmented Bus (c) NoC architectures (Packet/Circuit swtiching)

Figure 8: Experimental interconnection topologies

switched NOC. These chips are implemented using 0.18-μm technology with 69-bit bus width, and the number of the queuing buffer in the packet-switched NOC is eight.

The experimental results of the segmented bus, packet-switched NOC and circuit-switched are shown in Table 2. The area means the cell area of implementing the interconnecting architecture in Figure 8. The latency means the average latency of passing one packet data through a router/switch. The latencies of passing one packet data through the circuit-switched switch and the packet-switched router are 3.48ns and 29.66ns, respectively. The area and the latency of the proposed crossroad interconnection architecture are much smaller than that of the packet-switched NOC because of the lake of the queuing buffer and packet processing units. The power and the performance mean the power consumption and the time of completing the workload of MPEG4 decoder, respectively. We can find that the proposed circuit-switched NOC is more power efficient and high performance than that of the packet-switched NOC for specific applications.

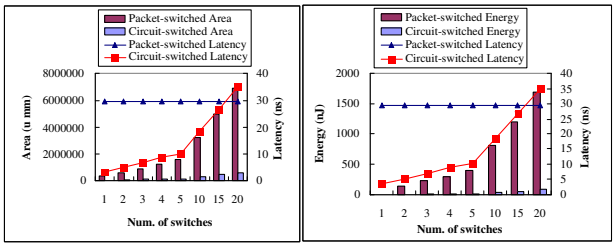


Figure 9: NOC area and latency characteristics

The left-hand side of Figure 9 shows the area and latency characteristics of the circuit-switched NOC and the packet-switched NOC. In the packet-switched NOC, the latency means the time of transmitting one message from one node (router or PU) to another node passing through a router. In circuit-switched NOC, the latency means the time of transmitting one message from one core to another core passing through switches. From the experimental results we can find that the delay of one packet-switched router is much larger than that of the switch because the former one consists of several complex components for processing packets. If the number of switches passing one message from one PU to another PU is less than few tens, the latency of transmission is

still less than that of the packet-switched NOC. Fortunately, in application-specific NOCs the length of critical path is always less than that. The right-hand side of Figure 9 shows the energy and latency characteristics of the circuit-switched NOC and the packet-switched NOC. The energy means the total energy consumption of completing MPEG-4 application workloads in line-shape topologies with different number of switches. From these two figures we can find that although the packet-switched NOC is high performance and high throughput, the area and the power consumption will significant increased when the scale of the network increases.

We also take applications MPEG-4, VOPD, and WMD as our cases studies, and the flow graph and the mapped topologies are discussed in [4]. In MPEG-4 and VOPD, the traffic will be highly localized within several pairs of cores, and the traffic is uniformly distributed among these cores in WMD. Figure 10 shows the results of cases studies for different applications with different communication localizations. The main advantage of the packet-switched NOC is that when the scale of the NOC becomes large, and PUs frequently and uniformly communicate to each other, it can provide high bandwidth for each pair for PUs because of the queuing buffer for keeping the conflicting messages. From this figure we can find that when the traffic is more localized, the energy and latency savings will be more. Increasing the amount of traffic localization causes less latency for a message in the circuit-switched NOC, and it also causes less energy dissipation. This happens because, on average, messages traverse fewer hops when there is greater localization. In this case, we can find that it is suitable for application-specific designs by circuit-switched NOCs under high-localization communications.

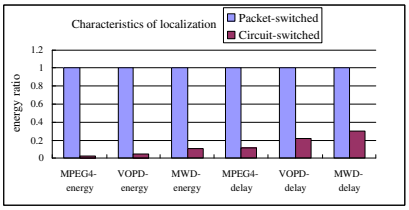


Figure 10: Effects of communication localization

The constraint of the proposed circuit-switched interconnection architecture is that when the scale of the NOC is too large, the latency of transmitting one message will be too high. In addition, it will consume more power when the traffic is uniformly distributed because of the long-distance transmissions and unexpected contentions. Fortunately, in a realistic application-specific SOC environment, different functions will be mapped to different parts of the SOC, and the traffic will exhibit highly localized patterns.

7. CONCLUSION

This paper presents the trade-offs between the packet-switched NOCs and the proposed circuit-switched *crossroad interconnection architecture*. The advantages of packet-switched NOC are flexible, scalability, and high throughput, however, they still suffer from long end-to-end latency, high implementation costs and unnecessary power consumption. Major advantages of the proposed circuit-switched interconnection architecture are lower power consumption, lower communication latency than that of the packet-switched NOC. If the scale of the SOC is just under few tens of cores, using the proposed circuit-switched NOC will be more attractive than the packet-switched NOC. Even in a large-scale application-specific SOC, cores are always mapped into local regions, and the proposed crossroad interconnection architecture is still a good choice.

8. REFERENCES

[1] Neal K. Bambha and Shuvra S. Bhattacharyya. Joint application mapping/interconnect synthesis techniques for embedded chip-scale multiprocessors. *IEEE Transaction on Parallel and Distributed Systems*, 16(2):99–112, February 2005.

[2] L. Benini and G. De Micheli. Networks on chips: A new soc paradigm. *IEEE Computer magazine*, pages 70–78, January 2002.

[3] Davide Bertozzi, Antoine Jalabert, Srinivasan Murali, Rutuparna Tamhankar, Stergios Stergio, Luca Benini, and Giovanni De Micheli. NoC synthesis flow for customized domain specific multiprocessor systems-on-chip. *IEEE Transaction on Parallel and Distributed Systems*, 16(2):113–129, February 2005.

[4] Kuei-Chung Chang, Jih-Sheng Shen, and Tien-Fu Chen. A low-power crossroad switch architecture and its core placement for network-on-chip. In *Proceedings of the International Symposium on Low Power Electronics and Design*, pages 375–380, August 2005.

[5] J. Y. Chen, W. B. Jone, J. S. Wang, H. I. Lu, and T. F. Chen. Segmented bus design for low-power systems. *IEEE Transactions on VLSI Systems*, 7(1):25–29, March 1999.

[6] X. Chen and L.-S. Peh. Leakage power modeling and. optimization in interconnection networks. In *Proceedings of the International Symposium on Low Power Electronics and Design*, pages 90–95, 2003.

[7] W.J. Dally and B. Towles. Route packets, not wires: On-chip interconnection networks. In *Proceedings of Design and Automation Conference*, pages 684–689, June 2001.

[8] Michael J Flynn. Area-time-power and design effort: the basic tradeoffs in application specific systems. In *Proceedings of the 16th International Conference on Application-Specific Systems, Architecture and Processors*, pages 3–5, 2005.

[9] Michael J Flynn and P. Hung. Microprocessor design issue: some thoughts on the road ahead. *IEEE Micro*, pages 16–31, 2005.

[10] M. Horowitz and W. Dally. How scaling will change processor architecture. In *Proceedings of the Solid-State Circuits Conference, Digest of Technical Papers*, pages 132–133, 2004.

[11] Cheng-Ta Hsieh and Massoud Pedram. Architectural energy optimization by bus splitting. *IEEE Transactions on Computer-Aided Design on Integrated Circuits and Systems*, 21(4):408–414, April 2002.

[12] A. Jalabert, S. Murali, L. Benini, and G.De Micheli. xpipescompiler: A tool for instantiating application specific networks on chips. In *Proceedings of DATE Conference*, 2004.

[13] Se-Joong Lee, Kangmin Lee, and Hoi-Jun Yoo. Analysis and implementation of proctical, cost-effective networks on chips. *IEEE Design and Test of Computers*, pages 422–433, 2005.

[14] Se-Joong Lee, Kangmin Lee, and Hoi-Jun Yoo. Packet-switched on-chip interconnection network for system-on-chip applications. *IEEE Transaction on Circuits and Systems*, 52(6):308–312, 2005.

[15] Se-Joong Lee, Seong-Jun Song, Kangmin Lee, Jeong-Ho Woo, Sun-Eun Kim, Byeong-Gyu Nam, and Hoi-Jun Yoo. A 800mhz star-connected on-chip network for application to systems on a chip. *ISSCC Dig. Tech. Papers*, pages 468–469, February 2003.

[16] S. Murali and G. De Micheli. Sunmap: A tool for automatic topology selection and generation for nocs. In *Proceedings of ACM/IEEE Design Automation Conference*, pages 914–919, 2004.

[17] P. P. Pande, G. D. Micheli, C. Grecu, A. Ivanov, and R. Saleh. Design, synthesis, and test of networks on chips. *IEEE Design and Test of Computers*, pages 404–413, 2005.

[18] Juha Plosila, Tiberiu Seceleanu, and Pasi Liljeberg. Implementation of a self-timed segmented bus. *IEEE Journals on Design and Test of Computers*, 20(6):44–50, 2003.

[19] Vijay Raghunathan, Mani B. Srivastava, and Rajesh K. Gupta. A survey of techniques for energy efficient on-chip communication. In *Proceedings of ACM/IEEE Design Automation Conference*, pages 900–905, June 2003.