A Power and Performance Model for Network-on-Chip Architectures

Nilanjan Banerjee, Praveen Vellanki and Karam S. Chatha,
Department of CSE, PO BOX 875406, Arizona State University, Tempe, AZ 85287-5406.
Email: {nilanjan.banerjee,praveen.vellanki,karam.chatha}@asu.edu

Abstract

Networks-on-Chip (NoC) has been proposed as a solution for addressing the design challenges of future high-performance nanoscale architectures. Innovative system-level performance models are required for designing NoC based architectures. This paper presents a VHDL based cycle accurate register transfer level model for evaluating the latency, throughput, dynamic, and leakage power consumption of NoC based interconnection architectures. We implemented a parameterized register transfer level design of the NoC architecture elements. The design is parameterized on (i) size of packets, (ii) length and width of physical links, (iii) number, and depth of virtual channels, and (iv) switching technique. The paper discusses in detail the architecture and characterization of the various NoC components. The paper presents results obtained by application of the model towards design space exploration, and power versus performance trade-off analysis of 4x4 mesh based NoC architecture.

1. Introduction

The International Technology Roadmap for Semiconductors (ITRS) predicts that by 2005, future generations of the high end custom VLSI architectures will be implemented in less than $50\ nm$ technology, and clocked in $10 - 20\ GHz$ range. A single VLSI chip will include tens of very complex highly integrated processing element blocks that will communicate with each other in $Gbits/s$. The physical characteristics of nanoscale technologies will pose several challenges to the designers. Global signal delays will span multiple clock cycles [1] [2]. Signal integrity will also be compromised due to increased RC effects, inductance, and cross-coupling capacitances [3] [4]. The large signal delays will make synchronous communication infeasible.

A paradigm shift in the design of global interconnection network is required for building nanoscale architectures. Nanoscale packet switched networks or Networks-on-Chip (NoC) are an attractive alternative to traditional interconnection network architectures [5] [6]. Packet switching supports asynchronous transfer of information. It provides extremely high bandwidth by distributing the propagation delay across multiple switches, thus pipelining the signal transmission.

Design of NoC based application specific architectures poses several challenges. It is not clear which network topologies are best suited for particular application specific architecture implementations [5]. NoC can exploit a lot more wiring resources than inter-chip networks. Hence, the topology design experience for shared memory multiprocessor systems does not necessarily apply to NoC. The trade-off in performance versus power consumption of interconnection network is also an open question. The trade-off analysis can be performed by varying the following design parameters: topology, length of physical links, width of physical links, buffer allocation, switching techniques, routing algorithms, and levels of service. Innovative performance evaluation models are required to address the design challenges of NoC based interconnection architectures.

Wassal et al. [7] proposed system-level performance and power models for a shared-memory internet protocol/asynchronous transfer mode switching fabric. Ye et al. [8] analyzed the power consumption in the switch fabrics of network routers and proposed system-level models for the same. Wang et al. [9] proposed a power-performance simulator for interconnection network called Orion. All these models are based on estimation of dynamic power due to switched capacitances. Hence, they are unable to estimate leakage power consumption. Further, since they model the architecture at the system-level, they do not consider power consumption due to control units and cross-coupling capacitances. Our model overcomes these limitations of the existing work. The results presented in Section 3 show that both power consumption due to controllers, and leakage power are significant components of the total power consumption of the NoC.

This paper presents a VHDL based cycle accurate model for evaluating the latency, throughput, dynamic, and leakage power of NoC based interconnection architectures. We implemented a parameterized register transfer level (RTL) design of the NoC architecture elements. The design is parameterized on (i) size of packets, (ii) length and width of physical links, (iii) number, and depth of virtual channels,

1530-1591/04 $20.00 (c) 2004 IEEE

Switching	Energy (in fJ)		
	100 μm	1000 μm	5000 μm
(000-000), (001-001),(010-010), (011-011), (100-100), (101-101),(110-110), (111-111)	0	0	0
(000-001), (000-100),(001-101), (010-011), (010-110), (011-111),(100-101), (110-111),	0.122	5.25	99
(000-010), (001-011),(100-110),(101-111)	0.189	12.48	213
(000-011), (000-110),(001-111), (100-111)	0.1914	5.94	121
(000-101), (010-111)	0.258	13.17	235
(000-111)	0.2075	20.46	66
(001-010), (010-100),(011-101), (101-110)	0.4314	29.54	504
(001-100), (011-110)	0.2309	7.83	165
(001-110), (011-100)	0.42	20.31	378
(010-101)	0.6864	48.9	830

Table 1: **3 wire-set characterization**

and (iv) switching technique. The RTL design was synthesized and the SPICE level netlist was extracted from the layout. The design was then characterized for delay, and dynamic and leakage power consumption at 180nm. The characterized values were integrated into the VHDL based RTL design to build the cycle accurate performance model.

The paper is organized as follows: Section 2 discusses the NoC based interconnection architecture and its performance characterization, Section 3 presents the experimental results, and Section 4 concludes the paper.

2. Architecture and Characterization

In the following paragraphs we describe the architecture of the various NoC elements (physical links, routers), and the techniques applied for their characterization.

2.1. Physical links

The physical links include the data and control wires for communication between two router elements of the interconnection network.

2.1.1. Characterization of physical links The power and performance of a physical link is determined by its width (number of bits of data and control signals), length, and capacitive load of the router. In nanoscale technologies, individual wires are modeled by distributed RLC expressions for accurate description of their physical characteristics [10]. The RLC and cross-coupling capacitances of the interconnection model were obtained from the Berkeley Predictive Technology Model website [11]. We characterized the links in sets of three, two and single wire, respectively for 0.18 μm technology. The three and two wire sets included the distributed RLC effects and cross-coupling capacitances, while the single wire model only included the distributed RLC effects. We considered three different types of links: local ($\leq 1000\mu m$), intermediate ($> 1000\mu m$ and $\leq 4000\mu m$), and global ($> 4000\mu m$) [1]. We obtained energy values for 64 (8 x 8), 16 (4x4) and 4 (2x2) different switching combinations for the three, two and single wire sets, respectively. The wire lengths were incremented in steps of $100\mu m$ up to $1000\mu m$, steps of $500\mu m$ up to $4000\mu m$ and steps of $1000\mu m$ up to $5000\mu m$. Table 1 summarizes the switching energy consumed in 0.18 μm technology for three wire-set switching for $100\mu m$, $1000\mu m$ and $5000\mu m$, respectively.

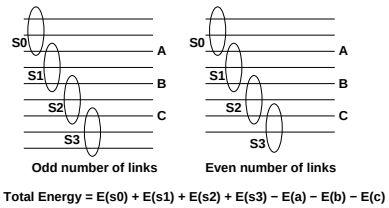

Figure 1: **Performance Evaluation of Links**

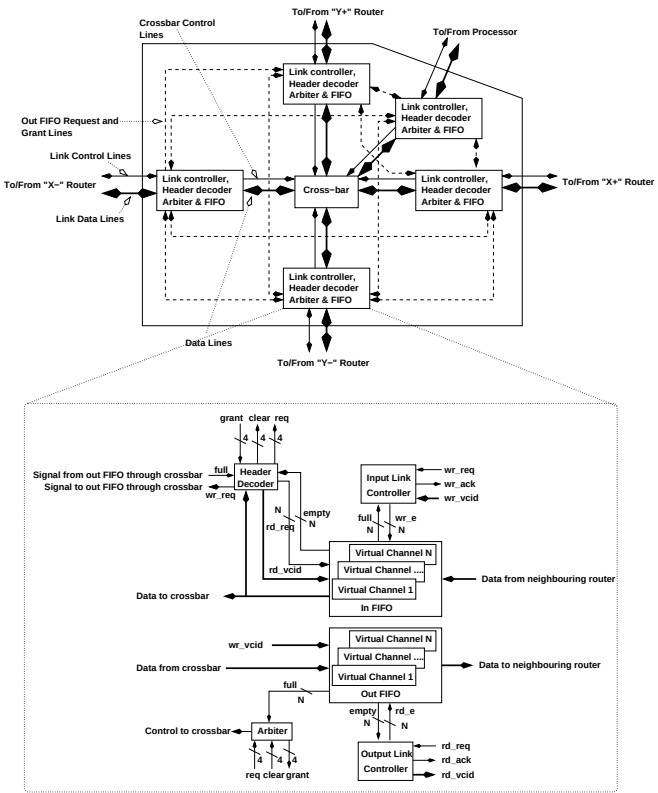

Figure 2: **Wormhole router architecture**

2.1.2. Performance evaluation of physical links We included the link characterization values as a table in our performance model. The energy consumed by a n-bit wide link can be calculated from the energy consumed by the three, two and single wire sets of similar length. For example, consider the 9-bit (odd) wide link shown in the left hand side of Figure 1. The total switching energy consumed by the links can be calculated by adding the switching energy consumed by the three wire sets S0, S1, S2 and S3, and subtracting the energy consumed by single wire links A, B, and C, respectively. In the case of a 8-bit (even) wide link shown in the right hand side of Figure 1, the energy consumed by two wire set S3 is included in the calculation. The length of the physical link which is a major factor in determining its power consumption and performance is specified by the designer.

2.2. The NoC Router

A wormhole router architecture that can be utilized in a 2D mesh topology is shown in Figure 2. The router consists of five unit routers to communicate in X-minus, X-plus, Y-minus, and Y-plus directions, and with the processor. Unit routers inside a single router are connected through a 5x5 crossbar. Data is transferred across routers or between the processor and the corresponding router by an asynchronous handshaking protocol. A single unit router is highlighted in lower half of Figure 2. It consists of input and output link controllers, virtual channels, a header decoder and an arbiter.

The message is partitioned into fixed length packets which are in turn broken down into flits for efficient data transfer. A packet consists of three kinds of flits - the header flit, the data flit and the tail flit, that are differentiated by two bits of control information. The header flit contains information of the destination router (X,Y) for each packet.

Data arrives at an input virtual channel of an unit router from either the previous router or the processor connected to the same router. The header decoder decodes the header flit of the packet after receiving data from the input virtual channel, decides the packet's destination direction (X-, X+, Y-, Y+, processor), and sends a request to the arbiter of the unit router in the corresponding direction. Once the grant is received the header decoder starts sending data from the input to the output virtual channel through the crossbar. There is a round robin priority based mechanism for each of the following tasks:

- Selection of an input virtual channel by the header decoder.
- Selection of an output virtual channel by the arbiter.
- Grant of the crossbar to the header decoder by arbiter.
- Selection of the output virtual channel by the link controller.

We designed register transfer level (RTL) models for each of the components separately. The larger components were characterized in terms of unit components like unit full adder, 2-bit comparator, 2:1 1-bit multiplexer, D flip-flop, and logic gates. SPICE net-lists for 0.18 μm technology were extracted for each component and characterized for energy and performance (shown in Table 2). In the following sections we describe in detail the functionality, architecture, and characterization of the various router elements (virtual channel, header decoder, arbiter, crossbar, link controller) shown in Figure 2. Performance evaluation of the entire router architecture is done by including the characterized energy values as table lookups in the RTL model.

2.2.1. Virtual Channel Every virtual channel (input and output) is a fifo. Each virtual channel consists of a memory bank, a status register to indicate full or empty conditions for every virtual channel location, and two registers that point to the next read and write locations respectively.

Unit Full Adder		2-bit Comparator	
1-bit flip at the output	0.096 pJ	Output Transition	0.15 pJ
2-bit flip at the output	0.1608 pJ	Input Change but no output change	0.0708 pJ
Input Change but no output change	0.0552 pJ	Leakage	0.077 fJ
Leakage	0.00438 fJ		
2-1 Multiplexer		**D Flip-Flop**	
Output Transition	0.061 pJ	Output Transition	0.1089 pJ
Input Change but no output change	1.527 fJ	Input Change but no output change	0.014 pJ
Leakage	0.013 fJ	Leakage	0.034 fJ

Table 2: **Unit components**

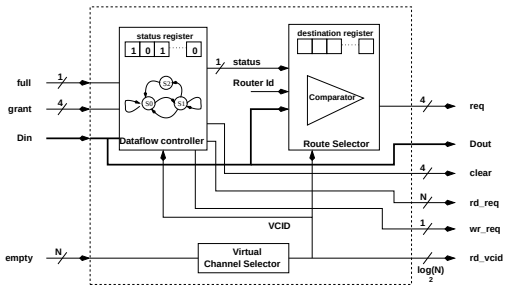

Figure 3: **Detailed view of header decoder**

The energy of the various sub-components was determined by utilizing the basic components shown in Table 2.

2.2.2. Header decoder The header decoder (Figure 3) implements the dimension order (X-Y) routing algorithm [12]. This is a deadlock free routing strategy where the packet is routed in the X direction first till the X-offset is zero, and then in the Y direction. It decodes the header flit of a packet from the nonempty input virtual channel and sends a request to the corresponding arbiter to obtain the crossbar connection. Once the connection is established, the header decoder starts sending data from the input virtual channel to the output virtual channel through the crossbar. If the communication gets blocked due to (a) the input virtual channel being empty or (b) the output virtual channel being full, the header decoder gives up the crossbar connection. It then selects the next non-empty input virtual channel for data transfer. In the following paragraphs we discuss the *virtual channel selector, dataflow controller,* and *route selector* in further detail.

Virtual channel selector: The virtual channel selector decides which input virtual channel will supply data to the header decoder by application of a round robin priority algorithm. It consists of a priority multiplexer unit, a modified onehot to binary converter, two adder units and a register as shown in Figure 4. The various components are divided into smaller units for characterization (see Figure 4).

Dataflow controller: This unit controls the dataflow from the input virtual channel to the output virtual channel through the crossbar. The controller shown in Figure 5 has three states - S0, S1 and S2. S0 represents the idle state. It shifts to state S1 when the input virtual channel is not empty. In state S1, the data is transferred from the input to the output virtual channel. In state S1, the controller waits for an arbiter grant. If it obtains a grant, it checks whether the selected input virtual

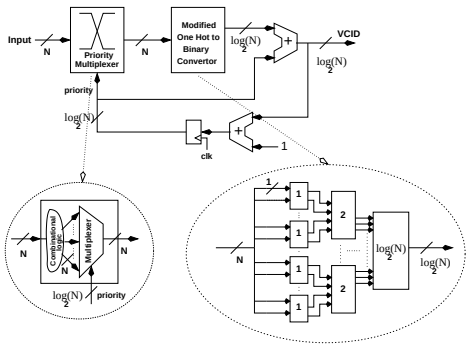

Figure 4: **Virtual channel selector**

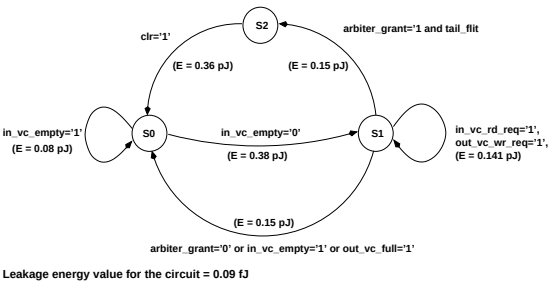

Figure 5: **Header decoder state machine**

channel is not empty, and the allocated output virtual channel is not full. If both the conditions are satisfied, it transfers the data from the input to the output virtual channel. On receiving the tail flit, it shifts to state S2. S2 represents the end of packet transmission. On the other hand, if it does not obtain a grant in S1, or the allocated output virtual channel becomes full or the selected input virtual channel becomes empty it goes to state S0. The state machine of the header decoder with the corresponding energy values of the various states and transitions is shown in Figure 5.

Route selector unit: The route select unit implements the dimension order routing. The power model for the route selector is implemented using unit multiplexers, unit comparators and flipflops that are shown in Table 2.

2.2.3. Arbiter An arbiter of a particular unit router (X+, X-, Y+, Y-, processor) receives requests from the header decoders of the other four unit routers and assigns an available output virtual channel to one of the requesting header decoders using a round robin arbitration mechanism. It has 3 sub-components: *virtual channel selector*, *status register* and *arbitration unit*.

Virtual channel selector: An output virtual channel can be allocated to a particular request only if it is not full and it has not been already allocated to a header decoder for data transfer. The output virtual channel selector used in the arbiter is similar to that of the header decoder discussed in Section 2.2.2.

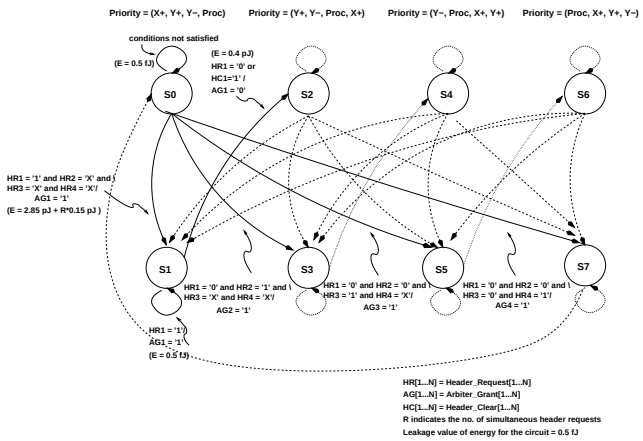

Figure 6: **Arbiter of "X-" unit router**

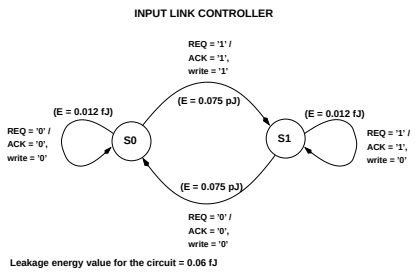

Figure 7: **Input link controller**

Status Register: It also maintains a status register to indicate whether an output virtual channel has been allocated for an incomplete data transfer. The status register is constructed using flipflops and combinational logic, that includes unit multiplexers and demultiplexers.

Arbitration unit: This unit grants control of the crossbar to requesting header decoders by the application of a round robin scheduling algorithm. It has eight states, S0 to S7. The states S0, S2, S4, S6 represent the idle states of the arbiter waiting for a request from the header decoders. The priority given to the requests from the header decoders of other unit routers for an X- arbiter is shown at the top of Figure 6. The states S1, S3, S5, S7 denote the condition when a grant has been given to a particular request. The arbiter remains in S1, S3, S5, and S7 states as long as the corresponding header request goes low or a clear signal is sent by the header decoder.

The state machine of the arbiter with the corresponding energy values of the various states and transitions has been shown in Figure 6. It should be noted that the energy consumed in the transitions from S0 to other states is empirically dependent on the number of simultaneous requests (R in Figure 6).

2.2.4. Crossbar We considered a multiplexer based completely connected architecture for the crossbar. It is made of five 4:1 K-bit multiplexers (where K is the fifowidth).

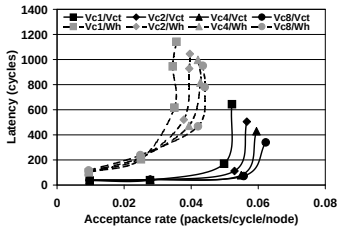

Figure 8: **Performance for VCT & WH**

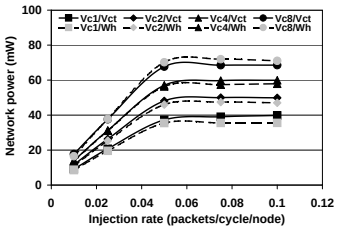

Figure 9: **Power for VCT & WH**

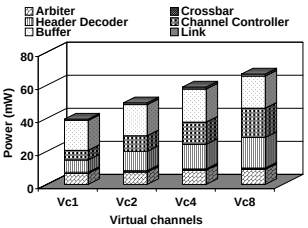

Figure 10: **Component power for VC**

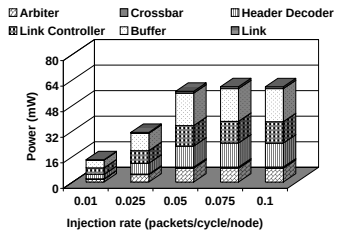

Figure 11: **Component power v/s injection**

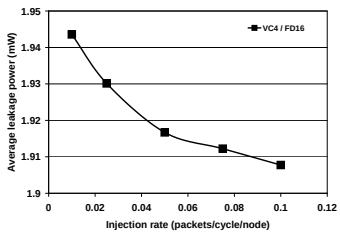

Figure 12: **Leakage power**

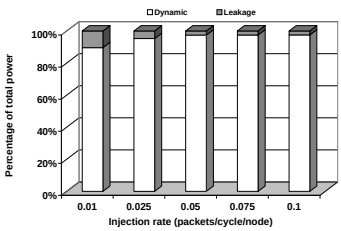

Figure 13: **Dynamic-Leakage v/s injection**

Each 4:1 K-bit multiplexer is composed of K 4:1 1-bit multiplexers. These 4:1 1-bit input multiplexers are further divided into 2 levels of multiplexers. Thus for a fifowidth of K, the total number of 2:1 1-bit multiplexers are 5*3*K. The characterized energy values for the 2:1 1-bit multiplexer are shown in Table 2.

2.2.5. Link controllers The link controller of the output virtual channels of a router communicates with the link controller of the input virtual channels of the neighboring router to write a flit. Since we consider GALS based SoC architecture, we assume that the two link controllers communicate with each other in a four phase asynchronous handshaking protocol based on *request* and *acknowledge* signals. The input and output virtual channels are connected on a one-to-one basis (output channel 1 connected to input channel 1, output channel 2 to input channel 2 and so on). The characterized input link controller is shown in Figure 7. The output link controller energy consumption is similar to the input link controller.

3. Results

We performed design space exploration and performance versus power trade-off analysis for a 4x4 mesh topology of a NoC based interconnection network. The physical links were 10 bits wide (8 bits of data, 2 bits of control) and supported uni-directional communication. The length of each link (distance between routers) was assumed to be 4.5 mm for an estimated chip area of 20 mm x 20 mm. In our experiments, the simulator generated uniformly distributed traffic to random destinations. The traffic was injected by the 16 processors by utilizing a uniform distribution over a designer specified time interval. The network was allowed to stabilize for the first 1000 cycles, after which it was run for 10,000 clock cycles. At the end of 10,000 clock cycles the total number of packets reaching the destination was calculated and their average latencies was computed. The average dynamic and leakage power consumption of the various components was also calculated over 10,000 clock cycles. The clock width was assumed to be 3 ns.

3.1. Virtual-cut through versus wormhole switch.

Virtual-cut through (VCT) is a switching technique where the header flit is processed and forwarded by the header decoder before the entire packet arrives at the input virtual channel. Wormhole (WH) switching is identical to virtual cut-though except that the fifo depth at the input virtual channel is less than the size of the packet [12]. For a fixed depth of the virtual channels, the size of the packet determines the switching mechanism of the network.

Figure 8 plots the variation of latency versus acceptance rate for packet sizes of 5 (VCT) and 7 (WH), with 1 to 8 virtual channels, and constant channel depths of 5. Acceptance rate is the number of packet accepted at destination per cycle per node, and is a function of the injection rate. It rises linearly with the injection rate before the network is congested, and is constant after congestion. Hence in Figure 8, a particular latency plot increases exponentially af-

ter the network is congested. As is clear from the figure, VCT switching gives lower latencies at higher acceptance rates than WH switching. Figure 9 plots the average power consumption of the various network architectures. In all the plots the power consumption increases with injection rate, and levels-off once the network is congested. For a particular number of virtual channels, both VCT and WH consume almost identical power. Therefore, VCT gives better performance at similar power consumption in comparison with WH. Hence, for the remainder of the paper we focus our attention on VCT switching.

3.2. Variation in number of virtual channels

Figures 8 and 9 also plot the performance and power consumption, respectively of the network for varying number of virtual channels. Increase in the number of virtual channels results in lower packet latencies with a concomitant increase in power consumption. Figure 10 plots the average power consumption per component for different virtual channels at the maximum injection rate. For all the architectures, the virtual channel buffers, header decoder and link controllers are the dominant consumers of power. Figure 11 plots the variation in component power consumption with increase in injection rate for a router with 4 virtual channels. Virtual channel buffers, header decoders, link controllers and arbiters show a larger increase in power consumption with increase in injection rate.

Figure 12 plots the variation in leakage power for increasing injection rate for a NoC consisting of routers with 4 virtual channels. As the injection rate is increased the leakage power drops from 1.943 mW to 1.907 mW. Figure 13 plots the percentage of dynamic and leakage power with increasing injection rate. At lower injection rates, the leakage power is over 15% of the total power consumption since many components are idle. At the higher injection rates, the dynamic power dominates due to increased switching activity. Figure 14 plots the dynamic and leakage power consumption per component at the highest injection rate (0.1). It can be seen from the plots, the virtual channel buffers are the major consumers of leakage power.

The following conclusions can be drawn from the plots. A NoC router architecture that is to be operated at higher injection rates should be designed with more number of virtual channels. Further, the virtual channels should be optimized for dynamic power consumption. A router that is to be operated at lower injection rates, should incorporate fewer virtual channels. In a low injection rate design, the virtual channels should be optimized for both dynamic and leakage power consumption.

4. Conclusion

We presented a cycle accurate performance and power evaluation model for NoC based interconnection networks. We presented results for extensive design space exploration and performance versus power trade-off analysis of a 4x4 mesh architecture. The current version of the model does not include error correction/detection, adaptive routing, and flow control schemes. Further, the model is currently applicable to only 0.18 μm technology. Future work will address these limitations.

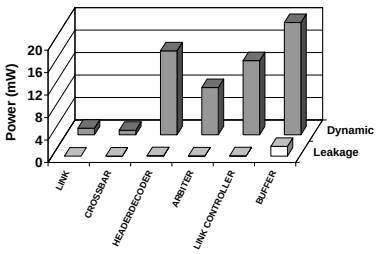

Figure 14: **Dynamic-Leakage/Component**

References

[1] D. Sylvester and K. Keutzer. "A Global Wiring Paradigm for Deep Submicron Design". *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, pages 242–252, February 2000.

[2] R.Ho, K. Mai and M. Horowitz. "The Future of Wires". *Proceedings of IEEE*, pages 490–504, April 2001.

[3] J. Davis and D. Meindl. "Compact Distributed RLC Interconnect Models - Part II: Coupled Line Transient Expressions and Peak Crosstalk in Multilevel Networks". *IEEE Transactions on Electron Devices*, 47(11):2078–2087, November 2000.

[4] D. Sylvester and K. Keutzer. "Impact of Small Process Geometries on Microarchitectures in Systems on a Chip". *Proceedings of the IEEE*, pages 467–484, April 2001.

[5] William J. Dally and Brian Towles. "Route Packet, Not Wires: On-Chip Interconnection Networks". In *Proceedings of DAC*, June 2002.

[6] Luca Benini and Giovanni De Micheli. "Networks on Chips: A New SoC Paradigm". *IEEE Computer*, pages 70–78, January 2002.

[7] A.G. Wassal and M.A. Hasan. "Low-power system-level design of VLSI packet switching fabrics". *IEEE Transactions on CAD*, 20:723–738, June 2001.

[8] Terry T. Ye, Luca Benini and Giovanni De Micheli. "Analysis of Power Consumption on Switch Fabrics in Network Routers". In *Proceedings of DAC*, 2002.

[9] H-S Wang, L-S Peh and S. Malik. "Orion: A Power-Performance Simulator for Interconnection Network". In *International Symposium on Microarchitecture*, Istanbul, Turkey, November 2002.

[10] P. Sotiriadis and A. Chandrakasan. "A Bus Energy Model for Deep Sub-micron Technology".

[11] Berkeley Predictive Technology Modeling. http://www-device.eecs.berkely.edu/ ptm. Technical report.

[12] J. Duato, S. Yalamanchili, L. Ni . *"Interconnection Networks, an Engineering Approach"*. IEEE Computer Society, 1997.