

A Distributed FIFO Scheme for on Chip Communication

Ray Robert Rydberg III, Jabulani Nyathi, José G. Delgado-Frias
School of EECS, Washington State University
Pullman, Washington 99164-2752
E-mail: {rrydberg,jabu,jdelgado}@eeecs.wsu.edu

Abstract—Interconnect delays are increasingly becoming the dominant source of performance degradation in the nano-meter regime, largely because of disturbances that result from parasitic effects. On chip communication now requires multiple clock cycles for signal propagation between communicating modules/components. Repeater insertion is widely used to improve global interconnect delays. We propose having distributed first in first out buffers to facilitate communication between components/modules of highly integrated systems, such as system on chip. This stateful scheme has very good tolerance for voltage and temperature variations. The buffer control circuitry is self-timed and allows for ease of interfacing in multiple domain clock designs. In this paper, we present the buffer and its associated control circuits that allow data transfers at a maximum frequency of 1.67 GHz in a 0.25 μm technology.

Keywords: On-chip-communication, buffer, wave-pipelining, self-timed, interconnect

I. INTRODUCTION

Global on chip interconnects are increasingly becoming a limiting performance factor in highly integrated systems such as system-on-chip (SoC). Some of the reasons leading to global interconnects being a limiting factor in system performance include: power supply drop variations, process variations, single clock synchronization [1], large wires with unpredictable delays [2], and interconnect power dissipation [3]. These limitations also imply that it would be difficult to achieve correct functional and reliable operations while maintaining low energy consumption within the interacting modules or components of SoC [3]. Proposed solutions to alleviate these limitations include: separating the computation problem from the communication problem and introducing networks on chips [3], having communicating components operate asynchronously, while the computational blocks operate synchronously based on locally generated clocks [4] and optimizing repeater insertion for global interconnect [5], [6].

Our work is based on the premise that global wires that span a significant fraction of the chip will impose signal delays that exceed the clock period [1]. Additionally, synchronizing operations on components running at different clock speeds is becoming more difficult due to clock skew and distribution [3]. With wire delays that exceed the clock period, it becomes apparent that the interacting components in a SoC design will perform computations much faster than the results could be transferred between components (a process that now requires multiple clock cycles). This assertion can further be substantiated by the fact that transistor switching speeds are much faster than wire delays. It is stated in [1] that as the transistor switching speeds improve, the wire delays increase. A copper low-k 35-nm process technology is reported to have device switching speeds of 2.5 ps, while interconnects of 1 mm in the same technology have delays of 250 ps (two orders of magnitude slower).

Much work has been done at the architectural, system and circuit levels to present some solutions to the various on chip communication issues. Some examples include: bus splitting, router based communication architectures and system level techniques such as commu-

nication based power management and adaptive supply voltage links [7]. In [3] the interacting components are viewed as a micronetwork with communication taking place among the components. This view allows for the replacement of the global wiring on a chip with a general purpose interconnection network. There are also circuit level techniques that have been explored to deal with the wire delay issue. The most widely used being the classical two inverter configuration with intermediate repeaters used for longer transmission distances. The list provided here is by no means exhaustive. We propose a distributed buffer approach that addresses the single clock synchronization issue, has very good tolerance for environmental variations, and has the potential to enable the communication to occur at the same rate as the computation.

II. MINIMIZING WIRE DELAYS

Existing literature shows that considerable work on repeater insertion at the algorithmic, as well as circuit, levels has been done [8], [9]. The published work presents approaches of optimizing current or voltage drivers and determining optimal distances at which buffers can be inserted. Power dissipation and optimization for various interconnect lengths have been reported in [10]. In the nano-meter regime, repeater insertion will not cope with increased disturbances caused by parasitic effects. To this end, Kaul and Sylvester have proposed a transition aware global signaling scheme and report clock rates of 1.5 GHz in 0.13 μm technology for 8 mm long wires [11]. Synchronizing multiple clock domain designs is yet another one of the emerging issues with the advent of system-on-chip designs.

The two inverter approach lacks memory and to avoid data overrun, the component producing data has to operate at the rate the signals can propagate through the inverter stages. The computing components, if far apart, have to perform operations at slower speeds to allow for slow global transfers through the global wires. The repeaters are inserted to break the wire delays. It must be noted that most communicating on chip modules might require data storage capabilities at the interface and buffer insertion does not provide this capability. Figure 1 shows a diagram of the two inverter approach with each data line staggered in order to reduce inter-wire capacitive coupling.

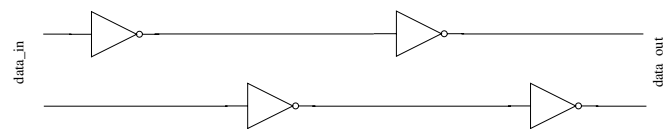


Fig. 1. Classical Buffer Insertion

Some disadvantages of this approach include: the need to provide additional circuitry to synchronize with components that run at different clock frequencies from the sending component, need for multiple clock cycles for successful transfer of data implying that the sending component must accommodate this constraint to avoid

data overrun, and need for multiple inverter stages to manage clock skew and distribution.

We suggest using a distributed first in first out (FIFO) buffering scheme to enable the communicating components to (i) operate at higher frequencies than could be attained with repeaters, (ii) provide a seamless interfacing capability for the interacting components operating at different speeds, (iii) permit the communicating components and the FIFO to remain idle and retain their current status in the absence of data, (iv) eliminate single clock synchronization, and (v) reduce the delays associated with interconnects. Figure 2 is a block diagram of the proposed scheme. Either one of the communicating components can signal the FIFO control circuit and initiate data transfer. We must add that the distributed buffer scheme lends itself well for use in any of the bus architectures such, as static priority based shared bus, hierarchical bus or ring based bus [12].

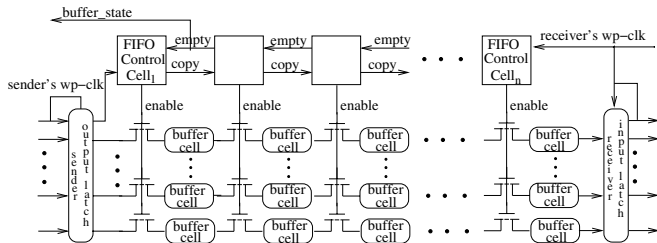


Fig. 2. FIFO buffer and control structure

The distributed FIFO scheme uses wave-pipelined clocks [13] in conjunction with some of the control circuit's internal signals to trigger the start of a transfer. The hybrid wave-pipelined clocks arrive at any of the component's latches, with their associated data, and are thus ideal for starting the inter-component data transfers.

We realize there is a cost in area due to additional circuitry. There is also a possible penalty on power dissipation, particularly under conditions that allow maximum speeds for data computation and transfers. The dynamic power component will be the most prominent under these conditions. We provide detailed analysis on these issues particularly power in Section IV. Despite these drawbacks, we show that the benefits of the distributed FIFO approach are not outweighed by these disadvantages. Multiple clock domains can easily be synchronized with the distributed FIFO communication scheme. Communicating components and the FIFO buffer can lie idle in the absence of data without a need for additional circuitry to provide this functionality or to perform clock gating. Allowing the circuits to lie idle implies that there is no switching activity leading to reduced power dissipation. We recognize that while keeping the circuit idle in the absence of data reduces dynamic power, static currents would contribute to power dissipation. Leakage currents in digital CMOS are minimized by techniques such as dynamic threshold CMOS, multiple threshold CMOS, substrate self-biasing scheme, etc [14]. We expect some of these techniques could be employed to lower the scheme's power given the fact that the design operates correctly at voltages as low as 0.6 V in a modest 0.25 μm technology.

Another attribute of the distributed FIFO scheme is that it enables for the elimination of the global clock, since each stage of the control circuitry becomes self-timed. The most important aspect of this approach is that the global interconnects can now allow data to propagate to the intended component at speeds limited only by the logic. Interconnects now have memory, permitting the communicating components to temporarily store the results on the data paths if necessary.

To realize the outlined advantages, we use a modified FIFO [15] control circuit and we add a feedback loop to the classical two inverter approach. The feedback loop gives the inverter pair the capability to store data at the cost of two additional transistors. Details of the FIFO circuit behavior can be found in references [15], [16]. Figure 3 shows control and buffer circuitry basic cells. We discuss the control circuit to show how the signals of interest are generated and how the approach promotes ease of interfacing in multiple clock domain designs.

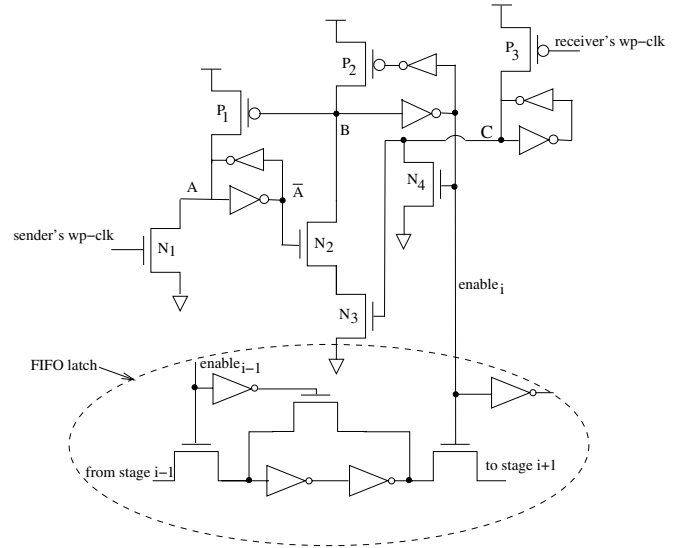


Fig. 3. FIFO control and latch basic cells

The nodes of interest within the FIFO control basic cell are labeled A , \bar{A} , B , C and $enable_i$. The components connected to the communication channel can either read from (receiving component) or write (sending component) to the communication channel. Permission to read from the channel is granted if there is data to be retrieved. This status is indicated by the logic level at node C . A logic 1 at node C indicates that the current cell is empty, while a logic 0 indicates that no new data can be accepted by the particular FIFO buffer cell being driven by the resulting enable signal. A logic 0 at node C ensures that the enable signal remains at logic 0 and thus no data movement from the previous FIFO latch to the current one takes place. In the event that node C is at logic 1 (a condition that constitutes the initial status of this node) and node A makes a transition from logic 1 to logic 0, node B is discharged. The enable signal after one inverter delay makes a transition from logic 0 to logic 1, allowing data on the channel to be shifted from the current latch to the next latch. The $wp-clk$ signals of the sender and receiver are responsible for initiating data transfers between communicating components.

The communication channel has been designed to provide a means to interface components that could be operating at different frequencies (multiple clock domain). This is achieved by providing the buffer status to both the sender and the receiver. At any given time the communicating modules can determine if they can access the channel or not. If the channel has capacity the intending component can transfer data. On the other hand if the channel is full the sending component cannot access it, only the receiver can retrieve data when ready. Nodes B and C of the first and last buffer control circuits have signals that indicate the status of the channel i.e. whether it has capacity or not. For any given stage if both nodes B and C are at logic 0, it implies that data transfer is in progress and the channel is

not available for new data. The condition $B = 0$ and $C = 1$ is not a valid state and can only occur due to delays in inverter switching speeds. Node C can be at logic 0 while node B is at logic 1. This state if recorded at the control stage closest to the sending component indicates that the buffer is full. When both nodes B and C are at logic 1 new data can be admitted into the channel. This is the normal state of the FIFO control; our design resets to this state.

The signals of nodes B and C for the first and last stages of the control circuitry are combined to provide a means to signal the communicating components to send data through the channel or to stop sending/receiving data. Performing a logic AND of these signals with the wave-wave-pipelined clock constitutes a very simple but effective interface. Figure 4 shows simulation results of two communicating components operating at different frequencies.

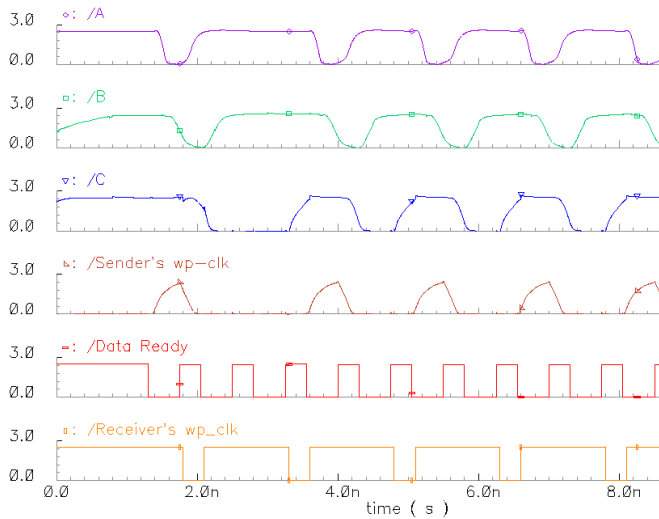


Fig. 4. Communicating components operating at different speeds

III. ENVIRONMENTAL PARAMETER VARIATION

A distributed FIFO buffer that uses shift registers to shift data along the channel and to retain the data in the event of stalls has been designed for comparison purposes. We refer to this design as the Distributed Shift Register (DSR). A channel that uses repeater insertion has also been simulated to compare delays under environmental variations. Power supply voltage and temperature fluctuations could lead to unreliable operations. The increased device densities imply that a large number of gates switch resulting in increased dynamic activity. This could result in power supply drop variations. These changes in supply voltage can result in data corruption or increased delays. Circuits need to be able to tolerate these changes. Therefore, it becomes necessary to ensure that a design is not sensitive to these changes. We have tested our circuits over a large range of power supply drops and temperature variations. Figure 5 shows the design's operation delays as the power supply voltage is reduced. The supply voltage was reduced from 2.5 V to 0.6 V. At 0.6 V, it was determined that the FIFO slowed to 34.6 MHz, a 99.8% degradation in operating speed. The latches of the channel held and propagated valid data at this low voltage. Temperature was varied from $-100\text{ }^{\circ}\text{C}$ to $225\text{ }^{\circ}\text{C}$ to evaluate the design's response against temperature changes. Performance degrades by 60% at $225\text{ }^{\circ}\text{C}$ and this value can be determined from Figure 6. These simulation results indicate that the distributed FIFO scheme though sensitive to environmental variations is able to maintain correct functional operations.

Simulation results show that the power supply voltage can only drop to 1.3 V before the operation is incorrect for the DSR and the classical inverter approach. The distributed FIFO at this 1.3 V point is 94 % faster. The distributed FIFO also shows very good tolerance to temperature changes as shown in Figure 6.

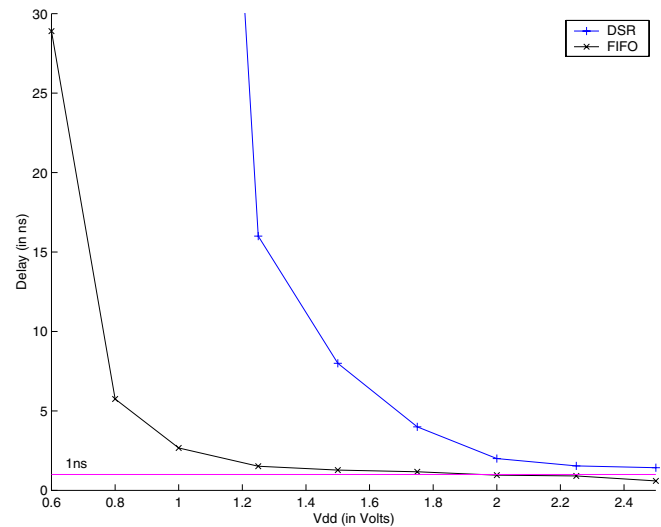


Fig. 5. Effect of supply voltage variation on FIFO performance

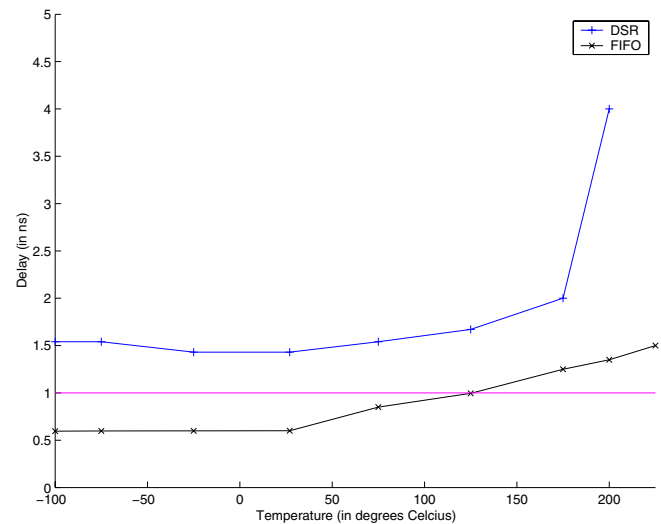


Fig. 6. Effect of temperature variation on FIFO performance

IV. POWER ANALYSIS AND TRANSISTOR COUNT COMPARISON

The departure from the classic two inverter insertion scheme for delay reduction leads to an increase in transistor count and additional switching activity. Comparing power dissipation and area to the buffer insertion scheme would result in an unfair comparison. Thus we use the DSR circuitry to make comparisons. We compute power for the two designs in idle, burst and normal modes. The normal mode refers to an operation that can occur at the highest frequency achievable for a continuous data stream. In idle mode, the clock driving the shift register is gated. The power analysis is for a 16-stage, 32-bit bus and the values recorded are post layout results.

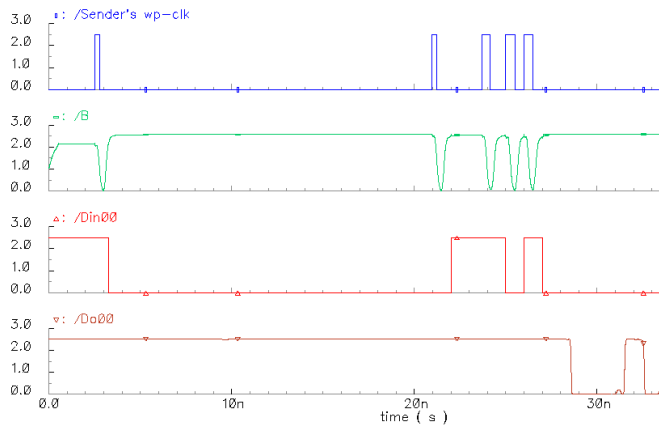


Fig. 7. Burst mode operation

TABLE I
POWER AND PDP FOR 16 STAGES OF FIFO AND DSR OVER 35 NS

Network - Speed	Power	Latency	PDP
FIFO - 1.67 GHz	111.25 mW	8 ns	890 pJ
FIFO - 750 MHz	87.25 mW	8 ns	698 pJ
FIFO - Burst	13.50 mW	8 ns	108 pJ
FIFO - Idle	192 μ W	N/A	N/A
DSR - 750 MHz	80.75 mW	24 ns	1938 pJ
DSR - Burst	28.75 mW	24 ns	690 pJ
DSR - Idle	100 μ W	N/A	N/A

The simulation results for burst mode operation appear in Figure 7 and depict a case where the computing component signals the FIFO controller to indicate that it is ready to accept data. There is only a single 32-bit word to be transferred when the FIFO is signaled thus the wave-pipelined clock stays at logic 0 after the initial transfer. The FIFO buffer propagates the data to the receiving module in 8 ns and returns to its normal state (all nodes corresponding to B and C set to logic 1). There is no activity for 20 ns and a burst of data arrives at time 21 ns and is transferred.

Table I has the related delays, power and energy figures relating to each mode of operation. The FIFO network consumes 40% more power than the DSR when in idle mode. During normal operation the FIFO consumes 7% more power than the DSR at the same operating frequency. The burst mode simulation show the most promise for the FIFO network in terms of power dissipation. For data sets that are intermittent, the FIFO network maintains the same latency while consuming 53% less power. The maximum speed for the FIFO is more than double that of the DSR (1.67 GHz compared to 750 MHz). A 16-bit stage FIFO design has 584 transistors compared to 510 for the DSR. This is a 12.6% increase in device count. For such a small penalty in area the FIFO buffer provides a significant 55% performance improvement.

V. CONCLUDING REMARKS

A distributed FIFO buffer that uses self-timed circuitry to propagate data between communicating components is presented. It can be used in any of the significant SoC bus architectures. We have shown that data transfers can be performed at 1.67 GHz. The scheme allows for ease of multiple clock domain interfacing and has very good tolerance for changes in temperature and power supply voltage. Supply voltage

drops down to 0.6 volts can be tolerated. The proposed approach is 55% faster than a clocked distributed FIFO buffer design under normal power supply voltage and is 94% faster at 1.3 volts where the other designs fail. The distributed FIFO scheme has the potential to reduce considerably global interconnect delays because it is stateful and pipelined. It provides a capability to tolerate stalls. Further-more, memory is readily available for temporary storage at the interface. Increased transistor count does not over-shadow the proposed scheme's performance gains. The classical two inverter scheme obviously has fewer devices and consumes less power compared to any of our designs but significantly delays signal transmission between distant communicating modules (requiring more than a single clock cycle to transmit). The distributed FIFO dissipates 7% more power at speeds comparable to those of the DSR design because it has more devices.

REFERENCES

- [1] L. P. Carloni and A. L. Sangiovanni-Vincentelli, "Coping with Latency in SoC Design," *IEEE Micro*, October 2002, pp. 24-35.
- [2] R. Seigmund and D. Muller, "Efficient Modeling and Synthesis of On-Chip Communication Protocols for Network-On-Chip Design," *Proceedings of the 2003 International Symposium on Circuits and Systems*, Vol 5, May 25-28, 2003, pp. 81-84.
- [3] L. Benini and G. De Micheli, "Networks on Chips: A New SoC Paradigm," *IEEE Computer*, Jan. 2002, pp. 70-78.
- [4] A. Iyer and D. Marculescu, "Power and Performance Evaluation of Globally Asynchronous Locally Synchronous Processors," *29th Annual International Symposium on Computer Architecture*, May 25-29, 2002, pp. 158-168.
- [5] V. V. Deodhar and J. A. Davis, "Voltage Scaling and Repeater Insertion for High-Throughput Low-Power Interconnects," *Proceedings of the 2003 International Symposium on Circuits and Systems*, Vol 5, May 25-28, 2003, pp. 349-352.
- [6] A. Nalamalpu and W. Burleson, "A Practical Approach to DSM Repeater Insertion: Satisfying Delay Constraints While Minimizing Area and Power," *Proceeding of the 14th Annual IEEE International Conference on ASIC/SoC*, September 12-15, 2001, pp. 152-156.
- [7] V. Raghunathan, M. B. Srivastava and R. K. Gupta, "A Survey of Techniques for Energy Efficient On-Chip Communication," *Proceedings of Design Automation Conference*, June 2-6, 2003, pp. 900-905.
- [8] M. L. Mui, K. Banerjee and A. Mehrotra, "Global Interconnect Optimization Scheme for Nanometer Scale Dissipation," *IEEE Transactions on Electron Devices*, Vol. 51, February 2004, pp. 195-203.
- [9] R. Bashirullah, W. Liu, and R. K. Cavin III, "Current-Mode Signaling in Deep Submicrometer Global Interconnects," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 11, Issue 3, June 2003, pp. 406-417.
- [10] K. Banerjee and A. Mehrotra, "A Power-Optimal Repeater Insertion Methodology for Global Interconnects in Nanometer Designs," *IEEE Transactions on Electron Devices*, Vol. 49, Issue 11, November 2002, pp. 2001-2007.
- [11] H. Kaul and D. Sylvester, "Low-Power On-Chip Communication Based on Transition-Aware Global Signaling (TAGS)," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 12, No. 5, May 2004, 464-476.
- [12] K. Lahiri, A. Raghunathan and S. Dey, "Evaluation of the Traffic-Performance Characteristics of System-on-Chip Communication Architectures," *Fourteenth International Conference on VLSI Design*, January 3-7, 2001, pp. 29-35.
- [13] J. Nyathi and J. G. Delgado-Frias, "A Hybrid Wave-Pipelined Network Router," *IEEE Transactions on Circuits and Systems, I: Fundamental Theory and Applications*, Vol. 49, No. 12, December 2002, pp. 1764-1772.
- [14] H. Soeleman, K. Roy and B. C. Paul, "Robust Subthreshold Logic for Ultra-Low Power Operation," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 9, No. 1, February 2001, pp. 90-99.
- [15] I. Sutherland and S. Fairbanks, "GasP: A Minimal FIFO Control," *Proc. of ASYNC*, 2001, pp. 46-53.
- [16] R. Rydberg, J. Nyathi and J. G. Delgado-Frias, "A Distributed FIFO Scheme for System on Chip Inter-Component Communication," *Proceedings of the International Conference on VLSI*, Las Vegas, Nevada, USA, June 2004, pp. 536-540.