Bus Serialization for Reducing Power Consumption

ABSTRACT

As device scales, on-chip interconnect is becoming a major consumer of VLSI power. Consequently bus power reduction becomes effective in total power reduction on chip multiprocessors and system-on-a-chip requiring long interconnects as buses.

In this paper, we advocate use of bus serialization to reduce bus power consumption. Bus serialization decreases the number of wires, and increases the pitch between the wires. The wider pitch decreases the coupling capacitances of the wires, and consequently reduces bus power consumption. Evaluation results indicate that our technique can reduce bus power consumption by 30% at 45nm technology process.

1. INTRODUCTION

Power reduction has emerged as one of the most important issues in recent VLSI design. As device scales, on-chip interconnects have increasing impact on total power consumption. In addition, this trend intensifies on chip multiprocessor (CMP) and system-on-a-chip (SoC) requiring a lot of long interconnects. For example, in a SoC with 4 ARM processors, 10 - 15 % of power is consumed by the interconnects [5].

Generally, there are two approaches for reducing on-chip bus power consumption: signal transition density reduction and effective capacitance reduction. A concept of signal transition density reduction is to minimize the signal transition on bus by data encoding schemes [9][2][4].

Effective capacitance reduction is minimizing the effective capacitance of wires by layout optimization. Coupling-driven bus ordering [8] and Non-uniform wire placement [6] have been proposed. The former reduces effective capacitance by reordering bus wires. The latter applies non-uniform spacing wire placement to address bus. The effectiveness of both two techniques depends on predictability of bit patterns.

In this paper, we propose a bus serialization technique for reducing on-chip bus power consumption without causing area and throughput penalties. The concept of our proposal is to reduce the coupling capacitances of adjacent wires. In the proposal, a conventional parallel bus is replaced with some serial buses. Adopting serial bus allows less number of wires and more spacing between wires in the same chip size. This results in reduced coupling capacitances and consequently bus power consumption. Though our proposal is categorized as effective capacitance reduction, bus serialization is effective in non-predictable bit pattern compared to the prior efforts.

This paper describes the details and quantitative effects of bus serialization. Advantages and disadvantages of it are also examined.

2. BUS SERIALIZATION

2.1 Concept

Bus power consumption P is generally calculated using the following formula.

$$P = afWCV^2.$$

In the formula, a is switching activity, f is bus frequency and W is number of wires. C is bus capacitance and V is voltage swing. It indicates that bus power consumption can reduce by reducing bus capacitance. In particular, in deep submicron technologies, coupling capacitance between wires is dominant in bus capacitance. Consequently, reducing coupling capacitance is effective in reducing bus power consumption.

We notice this point and propose a bus serialization technique. Bus serialization is a technique that a conventional parallel bus is replaced with some serial buses. Introducing serial bus decreases the number of wires, and permits wider spacing between wires in the same area. It makes coupling capacitances decrease. Therefore bus serialization can reduce bus power consumption.

In addition, bus serialization permits higher bus frequency. The wider wire pitch allows us room for improving bus capacitance as well as bus resistance. If wire spacing increases by the extra spacing, coupling capacitance decreases. If wire width increases otherwise, bus resistance decreases. Bus frequency is approximately in inverse proportion to the product of the bus capacitance and resistance. Therefore both low

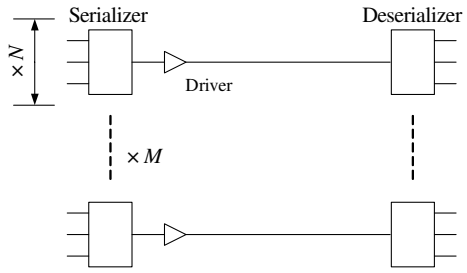

Figure 1: Circuit Structure of Serialized Bus.

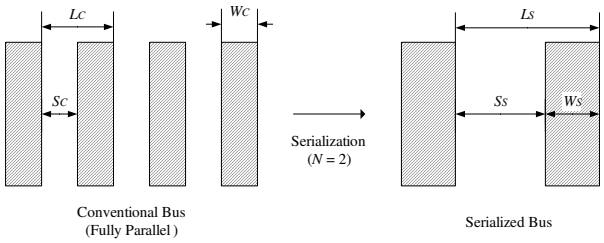

Figure 2: Bus Layout Design. Serialization decreases the number of wires, and increases wire width and spacing.

power consumption and high frequency can be achieved by optimizing wire width.

2.2 Basic Structure

Figure 1 shows the basic structure of a serialized bus. Total bus width is the product of number of wires M and serialization degree N. By serialization, the number of wires decreases from $M \cdot N$ to M. Each serial bus transfers N bits data per one transaction. Consequently the frequency of serialized bus must be $f \cdot N$ for the same throughput as conventional bus. Serializer and deserializer are used to convert from parallel data to serial data and vice versa.

Power consumption of conventional bus P_C and that of serialized bus P_S are shown in the followings.

$$P_C = af(M \cdot N)CV^2.$$
$$P_S = a(f \cdot N)M(C/\alpha)V^2.$$

It indicates that the serialized bus can reduce power consumption by capacitance ratio α without reducing throughput.

2.3 Layout Design Optimization

As has been mentioned, the extra spacing allowed by bus serialization can reduce bus capacitance or resistance. In this section, we propose a methodology for determining optimum wire width and spacing.

We define following parameters.

N : *Serialization degree.*
W_S : *Wire width* (*serialized bus*).
S_S : *Wire spacing* (*serialized bus*).

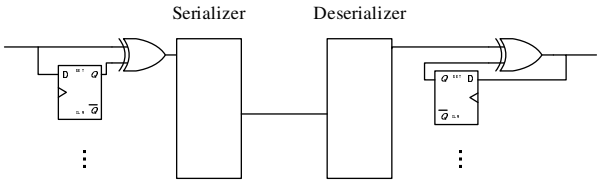

Figure 3: A Circuit Example of Differential Data Transfer.

We assume that the following parameters are defined by a bus specification and a metal configuration.

$M \cdot N$: *Total bus width.*
W_C : *Wire width* (*conventional bus*).
S_C : *Wire spacing* (*conventional bus*).
f_C : *Bus frequency* (*conventional bus*).

The following parameters can be calculated from previous parameters.

C : *Bus capacitance.*
R : *Bus resistance.*
f_S : *Bus frequency* (*serialized bus*).

Figure 2 shows the extra spacing gained by bus serialization. L_C is wire pitch in conventional (fully parallel) bus. Bus serialization increases wire pitch from L_C to L_S. For an identical wire's area, wider L_S can be used to increase wire spacing S_S or wire width W_S. These can be reduced to following.

$$W_S + S_S = (W_C + S_C) \cdot N. \quad (1)$$

This equation indicates the constraint for area.

To maintain the same throughput, bus frequency of a serialized bus must be N times as high as that of conventional bus. Therefore the following inequality is the constraint for bus frequency.

$$f_S > f_C \cdot N. \quad (2)$$

In this paper, we assume the formula developed by Kawaguchi and Sakurai [3] for calculating bus frequency, and the capacitance model developed by Chern et al. [1] for calculating bus capacitance.

When Equation (1) and Inequality (2) are fulfilled and C is minimized, the best W_S and S_S can be found.

2.4 Differential Data Transfer

Though bus serialization can reduce bus capacitance, bus serialization may also increase power consumption. Figure 4 shows an example of the case. When bits at a clock cycle are similar to bits at the previous clock, only a little power is consumed by the conventional bus. However, in this case, extra power is consumed by the serialized bus. In address bus, bit pattern like this frequently appears.

The problem is caused since the present bits are similar to the previous bits. Differential data transfer is a technique

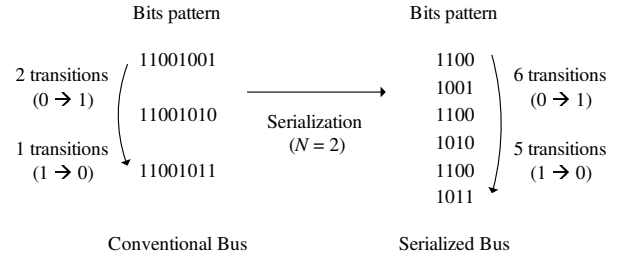

Figure 4: An Example where Bus Serialization Increases Power Consumption.

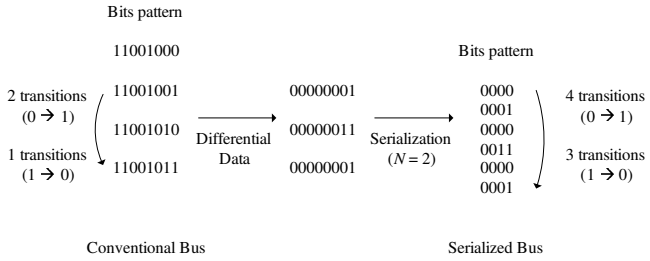

Figure 5: Differential Data Transfer Decreases The Power Consumption.

that transfers only the difference between the present bits and the previous bits as shown in Figure 5. By the technique, when bits pattern is sequential, many bits become 0, and the power consumption is reduced. As shown in Figure 4 and 5, signal transitions of serialized bus decreases from 11 to 7 by differential data transfer. Though the 7 transitions are more than 3 transitions of conventional bus, the difference between serialized bus and conventional bus becomes less with continuance of sequential bits pattern.

Figure 3 shows an example of circuit for differential data transfer.

2.5 Disadvantages of Bus Serialization

Possible disadvantages of our proposal are the additional power of peripheral circuits and clock skew. In this technique, we need serializers, deserializers and extra clock lines. If the power consumption of these circuits is larger than power reduction by our proposal, the technique is not effective. Therefore we must take care of the power for using the technique. Section 3.2.3 will examine the power consumption of peripheral circuits.

On the other hand, we must always consider the problem of clock skew. The margin of clock skew is in inverse proportion to serialization degree N. We currently do not investigate this issue in this paper.

3. EVALUATIONS

In this section, we evaluate the effects of our proposal. The bus specification that we assume is shown as follows.

$$\begin{aligned} \textit{Total bus width } M \cdot N &: 64\ \textit{bits} \\ \textit{Serialization degree } N &: 2 \\ \textit{Bus length} &: 5\ \textit{mm} \end{aligned}$$

Table 1: Processor Model.

issue width	4
data cache	16KB, 2-way, 64-byte block
instruction cache	16KB, 2-way, 64-byte block
L2 cache	ideal

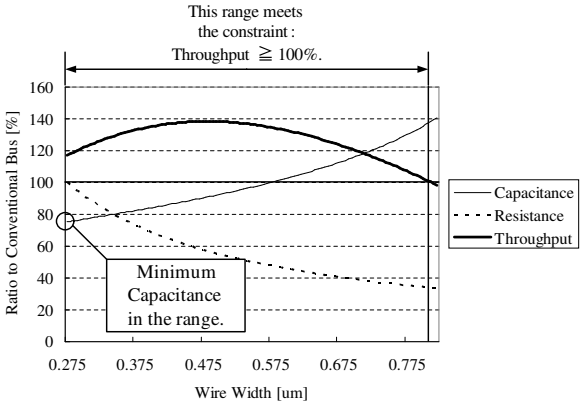

Figure 6: Layout Optimization (65nm process, serialization degree = 2).

We assume wire configurations derived from International Technology Roadmap for Semiconductors 2002 Update [7], and use eight applications from SPEC95int benchmark suite for estimating data dependency of bus power. We assume a processor with cache configuration shown in Table 1, and simulate 10 - 25 million bus transactions for each benchmark.

We assume bit patterns between L1 cache and L2 cache for estimation, and use load address, load data, store address, and store data in SPEC95int benchmark.

3.1 Capacitance Analysis

In this section, we estimate the effects of our proposal in reducing bus capacitance. We have proposed the methodology of layout design in Section 2.3. Figure 6 shows the relation among bus capacitance C, bus resistance R, and bus throughput T in 90nm technology. In Figure 6, throughput line in the area shown by the arrows meets Inequality (2). The circled point shows the wire width where bus capacitance is minimized. Therefore the wire width of this point is optimum from power viewpoint.

We find optimum width and capacitance in each technology by a similar approach. Figure 7 shows minimized bus capacitances by our proposal in each technology. It indicates that our proposal becomes more effective as gate length shrinks. This is because coupling capacitance becomes more dominant as wire spacing decreases.

3.2 Power Analysis

3.2.1 Power Reduction

Bus power consumption can be calculated from bus capacitance and bit patterns transferred by the bus.

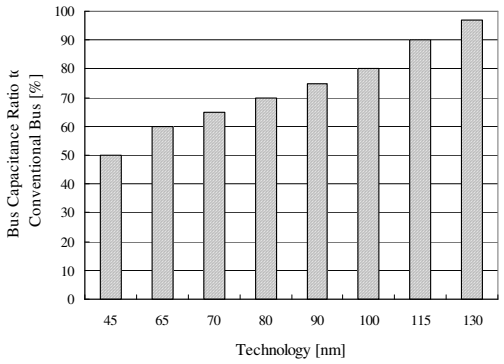

Figure 7: Capacitance Ratio of Serialized Bus to Conventional Bus.

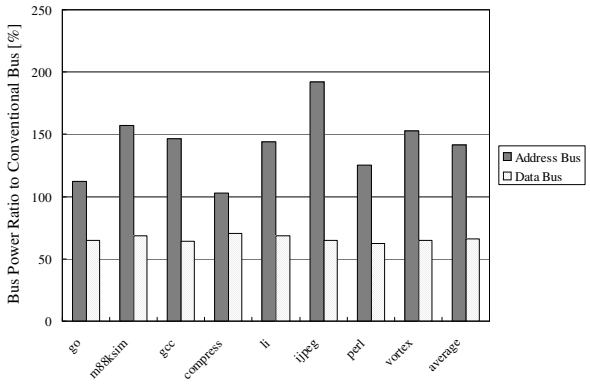

Figure 8: Power Consumption in Each Benchmark (45nm process).

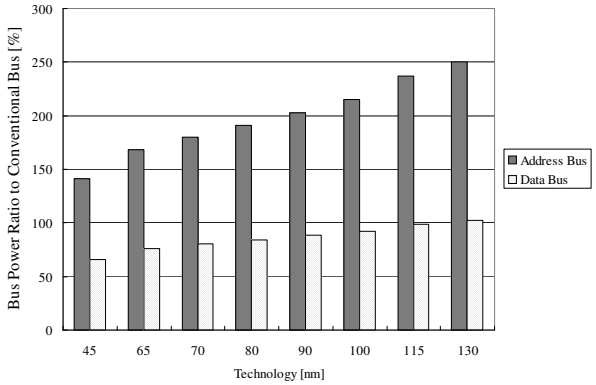

Figure 9: Average Power Consumption in Each Process.

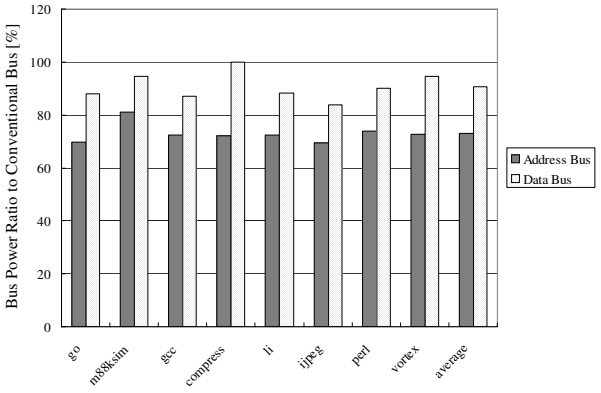

Figure 10: Differential Data Transfer: Power Consumption in Each Benchmark (45nm process).

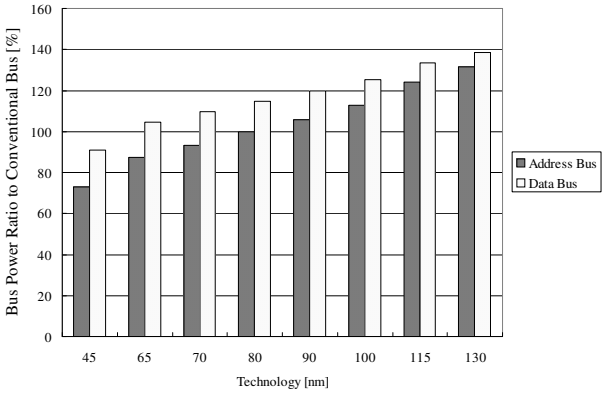

Figure 11: Differential Data Transfer: Average Power Consumption in Each Process.

Figure 8 shows power consumption ratio of the serialized bus to conventional bus in each benchmark. Figure 9 shows power consumption averages in each technology.

The results of Figure 8 indicate that there is a significant difference between address bus and data bus, and our proposal is effective when it is adopted to data bus. From Figure 9, we can find the same tendency and the effectiveness of our proposal becomes larger as gate length shrinks. However, these figures indicate that the power of address bus becomes worse, (the worst case: 250%) when we adopt the serialization.

3.2.2 Differential Data Transfer

As we have mentioned in Section 2.4, when bit pattern is sequential, bus power does not decrease by our proposal. The results shown in Figure 8 and 9 are got became an address is transferred sequentially.

Figure 10 and 11 show power consumption with differential data transfer shown in Section 2.4. It indicates that differential data transfer is effective in address bus.

Figure 12 shows comparison of unmodified serialized bus

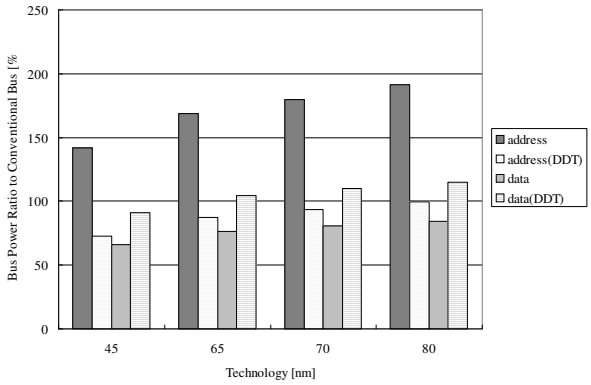

Figure 12: Compare Unmodified Serialized Bus to Serialized Bus with Differential Data Transfer.

and serialized bus with differential data transfer. According to the figure, differential data transfer is not effective in data bus. This is because bits pattern of data bus is not sequential and many bits of the bits pattern is 0. As shown in Figure 13 and 14, signal transitions increases from 6 to 10 by differential data transfer. Therefore, unmodified serialized bus is proper to data bus, and serialized bus with differential data transfer is proper to address bus.

3.2.3 Power of Peripheral Circuits

We have mentioned the circuit structure of serialized bus in Figure 1. In this section, we assume specific circuits shown in Figure 15 and 16 for estimating power of these circuits by SPICE simulation. Transistors in serializer, deserializer and D Flip-Flop (DFF) have the same gate width (basic width), and width of transistors in buffer is x 2, x 4 and x 8 of basic width. We assume that wire capacitance is $1pF$, which is calculated from device parameters and the bus length: 5mm, in both conventional bus and serialized bus.

Figure 17 shows the additional power of peripheral circuits in 180nm process. In the figure, *Peripherals* means serializer, deserializer and DFF in Figure 15 and 16. *Wire* means the power consumed in buffer. Indeed our proposal increases the power of peripheral circuits, but the additional power is only 2.4 % of conventional bus power consumption. As devise scales, power consumption of transistors relatively becomes less than that of wires. Therefore, in deep submicron technology: for example 45nm process, the additional power is not critical.

3.3 Delay and Area Analysis

Our proposal needs serializer and deserializer, and these additional circuits possibly cause additional delay. In this section, we estimate the additional delay by SPICE simulation. The circuits for SPICE simulation are shown in Figure 15 and 16. We assume that the delay by peripheral circuits is the interval from the rising of clock to the rising of buffer output.

Simulation results are shown in Table 2. This does not mean that serialization generally decrease delay because the delay depends on the circuit structure and gate width. However,

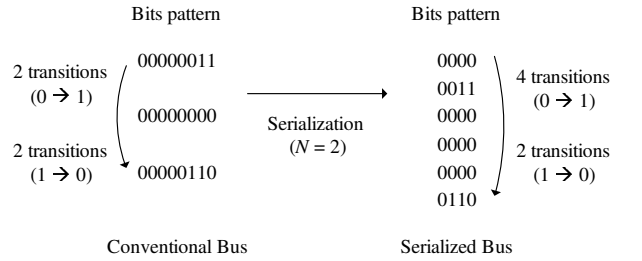

Figure 13: An Example where Differential Data Transfer Increases Power Consumption.

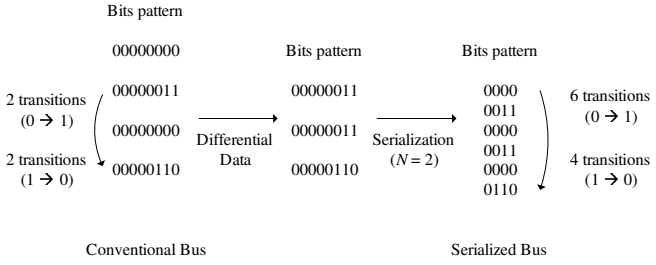

Figure 14: Differential Data Transfer Increases Power Consumption.

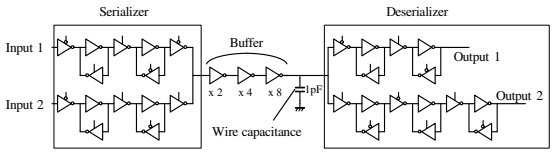

Figure 15: Circuits of Serialized Bus for Simulation.

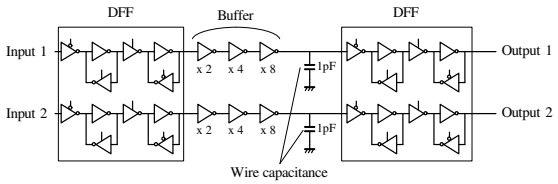

Figure 16: Circuits of Conventional Bus for Simulation.

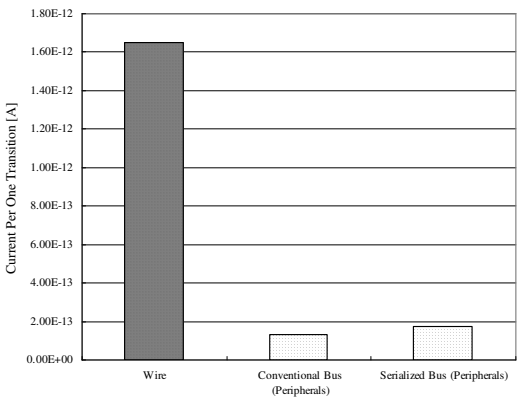

Figure 17: Current of Peripheral Circuits.

Table 2: Delay of Peripheral Circuits.

	Delay
Conventional Bus	0.17ns
Serialized Bus	0.15ns

the result indicates that an additional delay by bus serialization is not critical.

Additional area by bus serialization is not also critical. Indeed serialized bus requires serializers and deserializers that cause additional area, but serialized bus requires fewer buffers driving wires than conventional bus because of fewer wires required by serialized bus. For example, in Figure 15 and 16, serialized bus is almost equal to conventional bus in the number of transistors.

3.4 Variation of Serialization Degree

In this section, we consider increasing serialization degree from 2. Though more serialization degree cause less number of wires and less power consumption, serialized bus with large serialization degree requires higher bus frequency by the constraint shown in Inequality (2). According to our estimation, serialization degree 4 can not be achieved in 45nm process. This is because bus capacitance does not decrease to a quarter of it if the number of wires decreases to a quarter of it by bus serialization. However, more serialization degree is possible in more scaled process.

4. CONCLUSION

We first pointed out the importance for reducing bus power consumption. As gate length shrinks, power consumption of interconnects has more impact on total power consumption. In particular, buses are generally organized by long wires that have large capacitance, and coupling capacitance between wires is dominant in a deep sub-micron process.

We propose a bus serialization technique for reducing bus power consumption without decreasing throughput. Our proposal focuses on reducing coupling capacitance and introduces on-chip serial bus.

In this paper, we evaluated our proposal, assuming 64bit bus with serialization degree of 2 and wire length of 5mm. Evaluation results showed power reduction by our proposal depends on data that is transferred by bus. However, according to the results, bus power consumption decreases to 66% of conventional bus when serialized bus is adopted as data bus. Moreover, when serialized bus is adopted as address bus, bus power consumption decreases to 73% by differential data transfer.

We also evaluated additional costs by our proposal in 180nm process. Our proposal needs serializer, deserializer and extra clock line. However the additional delay and area by these circuits is negligible. The additional power consumption is 2.4% of conventional bus. This overhead is small enough compare to power reduction 27% - 34% by our proposal.

We did not evaluate clock skew and additional costs by differential data transfer. These are future works.

5. REFERENCES

[1] J.-H. Chern, J. Huang, L. Arledge, P.-C. Li, and P. Yang. Multilevel Metal Capacitance Models For CAD Design Synthesis Systems. *IEEE ELECTRON DEVICE LETTERS*, 13(1):32–34, 1992.

[2] M. Ikeda and K. Asada. Bus Data Coding with Zero Suppression for Low Power Chip Interface. In *Proc of. 1996 International Workshop on Logic and Architecture Synthesis*, pages 267–274, 1996.

[3] H. Kawaguchi and T. Sakurai. Delay and Noise Formulas for Capacitively Coupled Distributed RC Lines. In *Proc of. 1998 Asia South Pacific Design Automation Conference*, pages 35–43, 1998.

[4] S. Komatsu, M. Ikeda, and K. Asada. Bus Data Encoding with Coupling-driven Adaptive Code-book Method for Low Power Data Transmission. In *Proc of. 2001 European Solid-State Circuits Conference*, 2001.

[5] M. Loghi, M. Poncino, and L. Benini. Cycle-Accurate Power Analysis for Multiprocessor System-on-a-Chip. In *Proc of. 2004 ACM Great Lakes Symposium on VLSI*, pages 401–406, 2004.

[6] L. Macchiarulo, E. Macii, and M. Poncino. Wire Placement for Crosstalk Energy Minimization in Address Buses. In *Proc of. 2002 Design, Automation and Test in Europe*, pages 158–162, 2002.

[7] Semicondutor Industry Association. International Technology Roadmap for Semiconductors 2002 Update. http://public.itrs.net, 2002.

[8] Y. Shin and T. Sakurai. Coupling-Driven Bus Design for Low-Power Application-Specific Systems. In *Proc of. 2001 Design Automation Conference*, pages 750–753, 2001.

[9] M. R. Stan and W. P. Burleson. Bus-Invert Coding for Low-Power I/O. *IEEE Transactions on Very Large Scale Integration Systems*, 3(1):49–58, 1995.