Regular Paper

Bus Serialization for Reducing Power Consumption

Naoya Hatta,[†1] Niko Demus Barli,[†2] Chitaka Iwama,[†3]
Luong Dinh Hung,[†1] Daisuke Tashiro,[†4] Shuichi Sakai[†1]
and Hidehiko Tanaka[†5]

On-chip interconnects are becoming a major power consumer in scaled VLSI design. Consequently, bus power reduction has become effective for total power reduction on chip multiprocessors and system-on-a-chip requiring long interconnects as buses. In this paper, we advocate the 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% in the 45 nm technology process.

1. Introduction

Power reduction has emerged as one of the most important issues in recent VLSI design. With the shrinking scale of devices, on-chip interconnects are having an increasing impact on total power consumption. This trend is particularly intense for chip multiprocessors (CMPs) and system-on-a-chip (SoC) requiring many long interconnects. For example, in a SoC with 4 ARM processors, 10–15% of the power is consumed by the interconnects [5].

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

Effective capacitance reduction consists in minimizing the effective capacitance of wires by optimization of their layout. Two other proposed methods for achieving this are coupling-driven bus ordering [8] and non-uniform wire placement [6]. The former reduces effective capacitance by reordering bus wires, while the latter applies non-uniform-spacing wire placement to the address bus. The effectiveness of both two techniques depends on the 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 by several serial buses. Adopting the use of serial buses allows fewer wires and more spacing between them for the same chip size. This results in reduced coupling capacitances and consequently lower bus power consumption. Though our proposal is categorized as effective capacitance reduction, bus serialization is more effective than previously proposed techniques in reducing non-predictable bit patterns.

This paper describes the details and quantitative effects of bus serialization, and examines its advantages and disadvantages.

The remainder of the paper is organized as follows. Section 2 presents the details of our proposal. Section 3 reports the evaluation results. Finally, Section 4 concludes this paper.

2. Bus Serialization

2.1 Concept

Bus power consumption P is generally calculated by using the following formula:

$$P = afWCV^2.$$

In the formula, a is the switching activity, f is the bus frequency, W is the number of wires, C is the bus capacitance, and V is the voltage swing. This indicates that the bus power consumption can be reduced by reducing the bus capacitance. In particular, in deep submicron technologies, the coupling capacitance between wires is the principal determinant of bus capacitance. Consequently, reducing the coupling ca-

†1 Graduate School of Information Science and Technology, the University of Tokyo
†2 Texas Instruments Japan, Ltd.
†3 School of Law, the University of Tokyo
†4 Hitachi, Ltd.
†5 Institute of Information Security

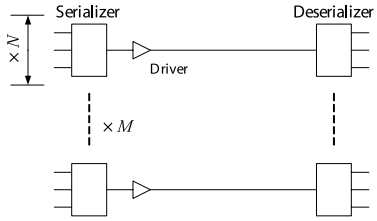

Fig. 1 Circuit structure of a serialized bus.

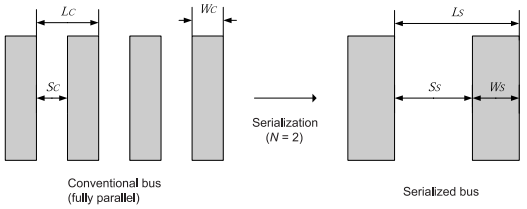

Fig. 2 Bus layout design. Serialization decreases the number of wires, and increases the wire width and spacing.

pacitance is effective for reducing the bus power consumption.

We propose a bus serialization technique that utilizes the above concept. Bus serialization is a technique whereby a conventional parallel bus is replaced by several serial buses. The introduction of serial buses decreases the number of wires, permits wider spacing between wires in the same area, and decreases coupling capacitances, thus reducing bus power consumption.

In addition, bus serialization permits higher bus frequency. The wider wire pitch allows room for improvement of bus capacitance as well as bus resistance. If the wire spacing is increased by the extra spacing, the coupling capacitance is decreased. Alternatively, if the wire width is increased, the bus resistance is decreased. Bus frequency is approximately in inverse proportion to the product of the bus capacitance and the resistance. Therefore, the both low power consumption and high frequency can be achieved by optimizing the wire width.

Generally, the advantages of bus serialization are high-speed data transfer achieved by resolving the signal integrity problem, and a relaxation of the pin limit. However, bus serialization requires high frequency, which makes implementation difficult. In our proposal, while the speed of data transfer does not increase, low-power data transfer can be achieved. One contribution of our research is the introduction of this trade-off into bus serialization.

2.2 Basic Structure

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

The power consumption of a conventional bus P_C and that of a serialized bus P_S are as follows:

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

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

2.3 Layout Design Optimization

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

We define the following parameters:

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

We assume that the following parameters are defined by a bus specification and a wire 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*).
T_C : *Bus throughput* (*conventional bus*).

The following parameters can be calculated from previous parameters:

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

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

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

This equation indicates the constraint for the area.

The bus throughputs T_C and T_S are calculated as follows:

$$T_C = f_C \cdot M \cdot N \quad (2)$$

$$T_S = f_S \cdot M \quad (3)$$

To maintain the same throughput, the following inequality must be fulfilled:

$$T_S \geq T_C \quad (4)$$

This means that the bus frequency of a serialized bus must be N times as high as that of a conventional bus. Therefore, the following inequality is the constraint for bus frequency:

$$f_S \geq f_C \cdot N. \quad (5)$$

In this paper, we assume the following formula developed by Kawaguchi and Sakurai [3] for calculating the bus frequency f:

$$\frac{1}{f} \approx R(C_C + C_L)\left(\frac{1.63C_C}{C_C + C_L} + 0.37\right). \quad (6)$$

We also assume the capacitance model developed by Chern, et al. [1] for calculating the bus capacitance. The details of the capacitance formulae are given in Appendix A.1.

When Eq. (1) and Inequality (5) 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, it may conversely increase power consumption. **Figure 3** shows an example of such a case. When the bits in a clock cycle are similar to bits in the previous clock cycle, only a small amount of power is consumed by a conventional bus. However, in this case, extra power is consumed by the serialized bus. In an address bus, such bit patterns frequently appear.

The problem arises because the present bits are similar to the previous bits. Differential data transfer is a technique whereby only the difference between the present bits and the previous bits is transferred, as shown in **Fig. 4**. In this technique, when a bit pattern is sequential, many bits become 0, and the power consumption is reduced. As shown in Figs. 3 and 4, the signal transitions of a serialized bus are decreased from 11 to 7 by differential data transfer. Though the 7 transitions are more than the 3 transitions of a conventional bus, the difference in the number of transitions between a serialized bus and a conventional bus becomes less with a continuance of sequential bit patterns.

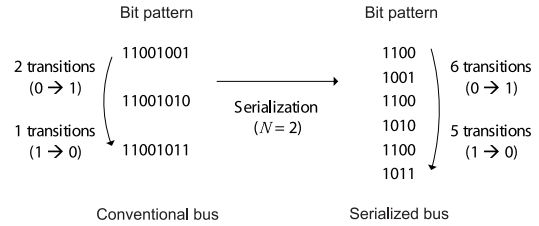

Fig. 3 Example where bus serialization increases the number of transitions.

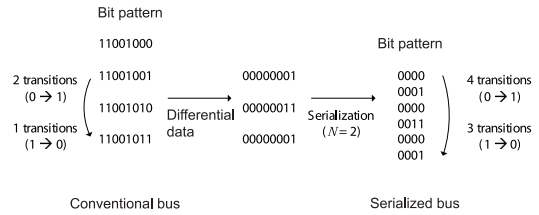

Fig. 4 Differential data transfer decreases the number of transitions.

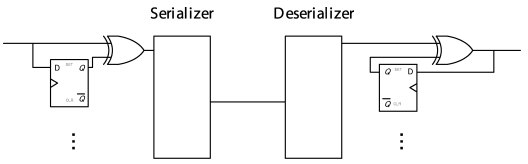

Fig. 5 Circuit example of differential data transfer.

Figure 5 shows an example of a circuit for differential data transfer.

2.5 Disadvantages of Bus Serialization

Possible disadvantages of our proposal are the need to provide additional circuits for bus serialization and problems related to the use of a high-frequency clock. In this technique, we need serializers and deserializers for serialization. If the power consumption of these circuits is larger than the power reduction achieved by our proposal, the technique is not effective. Section 3.2.3 will examine the power consumption of these circuits. In Section 3.3, we will show the performance penalty and the area penalty of these circuits.

Furthermore, we must always consider the problems related to the use of a high-frequency clock. A high-frequency clock may require more a complex circuit for clock generation. In addition, higher frequency causes a smaller margin of clock skew. The margin of clock skew is inversely proportional to the serialization degree N. If the margin of clock skew decreases, additional delay buffers may be required to de-

Table 1 Wire configuration.

Technology	Width W_C	Spacing S_C	Wire thickness T	Dielectric thickness H
130 nm	450 nm	450 nm	720 nm	630 nm
115 nm	380 nm	380 nm	608 nm	532 nm
100 nm	320 nm	320 nm	544 nm	480 nm
90 nm	275 nm	275 nm	468 nm	413 nm
80 nm	240 nm	240 nm	408 nm	360 nm
70 nm	215 nm	215 nm	366 nm	344 nm
65 nm	195 nm	195 nm	351 nm	312 nm
45 nm	135 nm	135 nm	243 nm	216 nm

Table 2 Processor configuration.

Issue width	4
Data cache	16 KB, 2-way, 64-byte block
Instruction cache	16 KB, 2-way, 64-byte block
L2 cache	ideal

crease the clock skew. The clock generation circuit and delay buffers may cause additional power and area. Though these problems should be considered before adopting bus serialization, they are not investigated in this paper.

3. Evaluation

In this section, we evaluate the effects of our proposal. We assume a chip multiprocessor with a shared L2 cache as the target processor, and apply bus serialization to the bus between the L1 cache and the L2 cache. The bus specification that we assume is as follows:

Total bus width $M \cdot N$: 64 bits
Serialization degree N : 2
Bus length : 5 mm

Table 1 shows the wire configurations derived from the International Technology Roadmap for Semiconductors [7].

To estimate the data dependency of bus power, we use a processor simulator of a conventional single processor. This simulation corresponds to a simulation of executing an application on a chip multiprocessor. Eight applications from the SPEC95int benchmark suite are used for the estimation. We assume the cache configuration shown in **Table 2**, and simulate 10–25 million bus transactions for each benchmark.

3.1 Capacitance Analysis

In this section, we estimate the effects of our proposal in reducing bus capacitance. We have proposed a methodology for layout optimization in Section 2.3. **Figure 6** shows the relations among the bus capacitance C, the bus resistance R, and the bus throughput T in 90 nm technology. In Fig. 6, the throughput line in the area shown by the arrows satisfies Inequality (5). The circled point shows the wire width at which the bus capacitance is minimized. The wire width of this point is optimal from a power viewpoint.

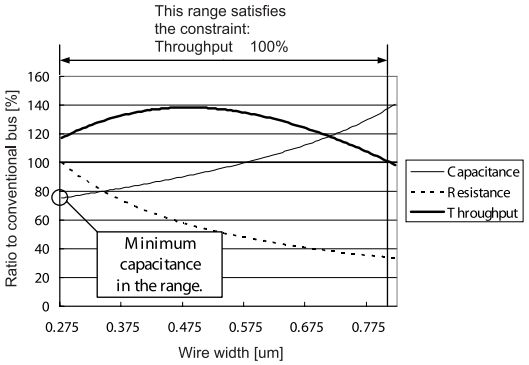

Fig. 6 Layout optimization (90 nm process, serialization degree = 2).

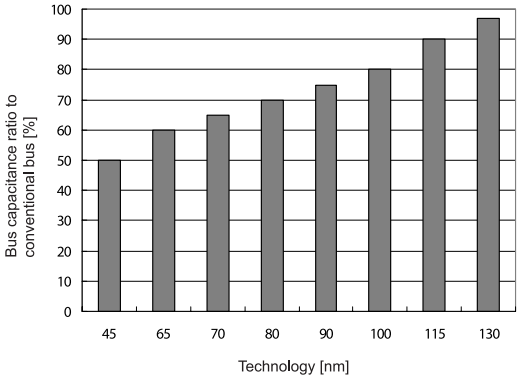

Fig. 7 Capacitance ratio of serialized bus to conventional bus.

We find the optimum width and capacitance in each technology by using a similar approach. **Figure 7** shows the minimized bus capacitances obtained by following our proposal in each technology. It indicates that our proposal will become more effective as process technology advances. This is because the coupling capacitance becomes more dominant as wire spacing decreases.

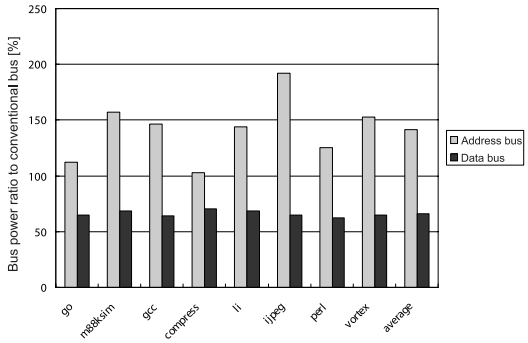

Fig. 8 Power consumption in each benchmark (45 nm process).

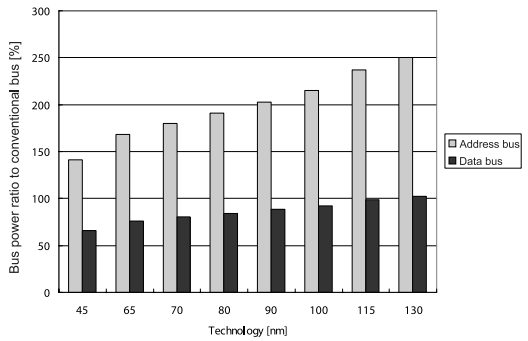

Fig. 9 Average power consumption in each technology.

3.2 Power Analysis

3.2.1 Power Reduction

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

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

The results in Fig. 8 indicate that there is a significant difference between the address bus and the data bus, and that our proposal is effective when it is adopted for the data bus. Figure 9 shows the same tendency, and the effectiveness of our proposal becomes larger as the gate length shrinks. However, these figures indicate that the power of the address bus decreases, (the worst case: 250%) when we adopt a serialization strategy. This is because bit patterns are sequential in an address bus and power consumption increases as a result of the effect described in Section 2.4.

3.2.2 Differential Data Transfer

As mentioned in Section 2.4, when the bit pattern is sequential, the bus power consumption is increased by our proposal. The results shown in Figs. 8 and 9 confirm our observation.

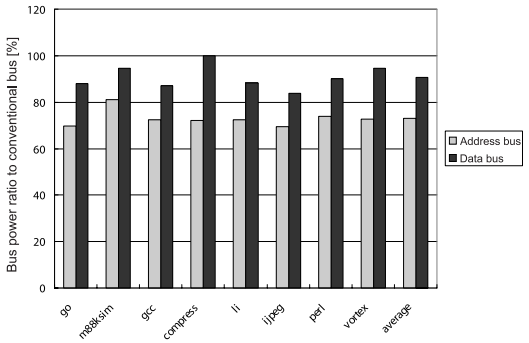

Fig. 10 Differential data transfer: power consumption in each benchmark (45 nm process).

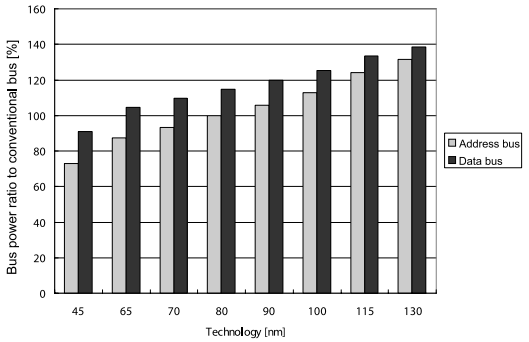

Fig. 11 Differential data transfer: average power consumption in each technology.

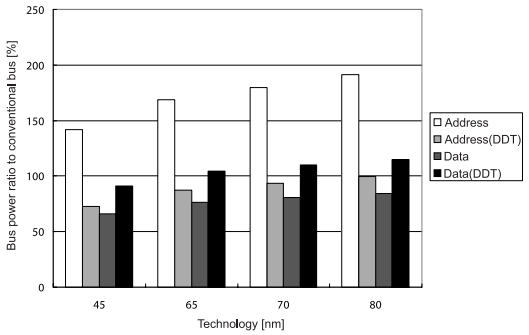

Fig. 12 Comparison of an unmodified serialized bus and a serialized bus with differential data transfer.

Figures 10 and **11** show power consumption with the differential data transfer described in Section 2.4. Differential data transfer is effective in an address bus.

Figure 12 shows a comparison of an unmodified serialized bus and a serialized bus with differential data transfer. According to the figure, differential data transfer is not effective in a data bus. This is because the bit pattern of

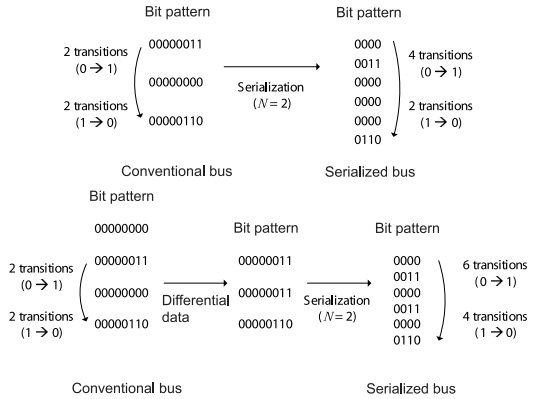

Fig. 13 Example where differential data transfer increases the number of transitions.

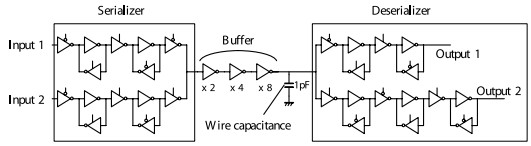

Fig. 14 Circuits of a serialized bus for simulation.

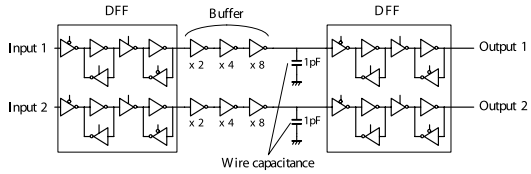

Fig. 15 Circuits of a conventional bus for simulation.

a data bus is not sequential, and many bits of the bit pattern are 0. As shown in **Fig. 13**, signal transitions increase from 6 to 10 as a result of differential data transfer. Therefore, an unmodified serialized bus is suitable for use as a data bus, and a serialized bus with differential data transfer is suitable for use as an address bus.

3.2.3 Power of Peripheral Circuits

We showed the circuit structure of a serialized bus in Fig. 1. In this section, we assume the specific circuits shown in **Figs. 14** and **15**, and estimate the power of these circuits by SPICE simulation. The transistors in a serializer, deserializer, and D Flip-Flop (DFF) have the same gate width (basic width), and the widths of transistors in buffers are two times, four times, and eight times the basic width. We assume that the wire capacitance is $1\,pF$, which is calculated from device parameters and the bus length (5 mm), in both conventional and serialized buses.

Figure 16 shows the additional power of

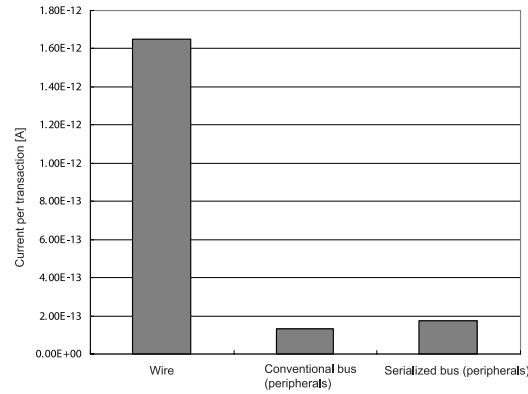

Fig. 16 Current of peripheral circuits.

Table 3 Delay of peripheral circuits.

	Delay
DFF + buffer (conventional bus)	0.17 ns
Serializer + buffer (serialized bus)	0.15 ns

peripheral circuits in the 180 nm process. In the figure, *Peripherals* includes the power consumption by serializer, deserializer, and DFF in Figs. 14 and 15. *Wire* includes the power consumed in the buffer. Indeed, our proposal increases the power of peripheral circuits, but the additional power is only 2.4% of the power consumed by a conventional bus. As the scale of devices shrinks, the power consumption of transistors becomes relatively less than that of wires. Therefore, in deep submicron technology, the additional power is not critical.

3.3 Delay and Area Analysis

Our proposal requires a serializer and deserializer, and these additional circuits may cause additional delay. In this section, we estimate the additional delay by SPICE simulation. The circuits for SPICE simulation are shown in Figs. 14 and 15. We assume that the delay caused by the peripheral circuits is the interval from the rising of the clock to the rising of the buffer output.

The simulation results are shown in **Table 3**. They do not mean that serialization generally decreases the delay, because the delay depends on the circuit structure and gate width. However, they indicate that an additional delay due to bus serialization is negligible. Since the additional delay caused by the peripheral circuits is negligible, degradation of bus performance is also negligible.

The additional area created by bus serialization is also not critical. Indeed, a serialized bus requires serializers and deserializers that cause

additional area, but a serialized bus requires fewer buffers driving wires than a conventional bus because it has fewer wires. For example, in Figs. 14 and 15, the serialized bus has almost the same number of transistors as the conventional bus.

3.4 Variation of Serialization Degree

In this section, we consider increasing the serialization degree from 2. Though a higher serialization degree causes fewer wires and lower power consumption, a serialized bus with a high serialization degree requires a higher bus frequency, because of the constraint shown in Inequality (5). According to our estimation, serialization degree 4 cannot be achieved in 45 nm process. This is because the bus capacitance does not decrease to a quarter of its original value if the number of wires is decreased to a quarter by bus serialization. However, a higher serialization degree is possible in a more scaled process.

4. Conclusion

We began by pointing out the importance of reducing bus power consumption. As gate length shrinks, the power consumption of interconnects has a greater impact on total power consumption. In particular, buses are generally designed as 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 the coupling capacitance by introducing an on-chip serial bus.

In this paper, we have evaluated our proposal, assuming a 64 bit bus with a serialization degree of 2 and a wire length of 5 mm. The evaluation results showed that the power reduction achieved by our proposal depends on the data that is transferred by the bus. According to the results, the bus power consumption decreases to 66% of that of a conventional bus when a serialized bus is adopted as data bus in the 45 nm process. Moreover, when a serialized bus is adopted as the address bus, the bus power consumption decreases to 73% as a result of differential data transfer in the 45 nm process.

We also evaluated the additional costs of our proposal in the 180 nm process. Our proposal requires a serializer, a deserializer and an extra clock line. However, the additional delay and area required by these circuits are negligible, and the degradation of the total bus performance is also negligible. The additional power consumption is 2.4% of that of a conventional bus. This overhead is small in comparison with the power reduction of 27–34% realized by our proposal.

We did not evaluate the additional costs caused by the use of a high-frequency clock and differential data transfer. These remain subjects for future work.

Acknowledgments This research is partially supported by Grant-in-Aid for Fundamental Scientific Research B(2) #13480077 and B(2) #1630013 from Ministry of Education, Culture, Sports, Science and Technology Japan, Semiconductor Technology Academic Research Center (STARC) Japan, CREST project of Japan Science and Technology Corporation, by 21st century COE project of Japan Society for the Promotion of Science, and VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Cadence Design Systems, Inc, Hitachi Ltd, Mentor Graphics, Inc, and Synopsys, Inc.

References

1) Chern, J.-H., Huang, J., Arledge, L., Li, P.-C. and Yang, P.: Multilevel Metal Capacitance Models for CAD Design Synthesis Systems, *IEEE Electron Device Letters*, Vol.13, No.1, pp.32–34 (1992).
2) Ikeda, M. and Asada, K.: Bus Data Coding with Zero Suppression for Low Power Chip Interface, *Proc. 1996 International Workshop on Logic and Architecture Synthesis*, pp.267–274 (1996).
3) Kawaguchi, H. and Sakurai, T.: Delay and Noise Formulas for Capacitively Coupled Distributed RC Lines, *Proc. 1998 Asia South Pacific Design Automation Conference*, pp.35–43 (1998).
4) Komatsu, S., Ikeda, M. and Asada, K.: Bus Data Encoding with Coupling-driven Adaptive Code-book Method for Low Power Data Transmission, *Proc. 2001 European Solid-State Circuits Conference* (2001).
5) Loghi, M., Poncino, M. and Benini, L.: Cycle-Accurate Power Analysis for Multiprocessor System-on-a-Chip, *Proc. 2004 ACM Great Lakes Symposium on VLSI*, pp.401–406 (2004).
6) Macchiarulo, L., Macii, E. and Poncino, M.: Wire Placement for Crosstalk Energy Minimization in Address Buses, *Proc. 2002 Design, Automation and Test in Europe*, pp.158–162

(2002).
7) Semicondutor Industry Association: International Technology Roadmap for Semiconductors 2002 Update, http://public.itrs.net (2002).
8) Shin, Y. and Sakurai, T.: Coupling-Driven Bus Design for Low-Power Application-Specific Systems, *Proc. 2001 Design Automation Conference*, pp.750–753 (2001).
9) Stan, M.R. and Burleson, W.P.: Bus-Invert Coding for Low-Power I/O, *IEEE Transactions on Very Large Scale Integration Systems*, Vol.3, No.1, pp.49–58 (1995).

Appendix

A.1 Capacitance Formulae

The model of Chern, Huang, et al. [1] is one of the empirical models for multilevel interconnect capacitance. In this model, the capacitance of the target wire is calculated from the wire configurations of the target metal layer 2, the upper layer 3, and the lower layer 1. According to the model, the formulae for the total capacitance C, coupling capacitance C_C, and load capacitance C_L are as follows:

$$C = 2C_C + C_L \tag{7}$$

$$\begin{aligned}\frac{C_C}{\epsilon} &= \frac{W_1W_2}{H} + 0.9413\left(1 - 0.326e^{\frac{-T_1}{0.133S_1}}\right.\\ &\left. - 0.959e^{\frac{-S_1}{1.966H}}\right)2W_2\left(\frac{S_1}{S_1+0.01H}\right)^{0.2}\\ &+ 0.9413\left(1 - 0.326e^{\frac{-T_2}{0.133S_2}}\right.\\ &\left. - 0.959e^{\frac{-S_2}{1.966H}}\right)2W_1\left(\frac{S_2}{S_2+0.01H}\right)^{0.2}\\ &+ 1.14\left(1 - 0.326e^{\frac{-T_1}{0.133S_1}}\right.\\ &\left. - 0.959e^{\frac{-S_1}{1.966H}}\right)(S_2S_1)^{0.5}\left(\frac{W_2}{H}\right)^{0.182}\\ &+ 1.14\left(1 - 0.326e^{\frac{-T_2}{0.133S_2}}\right.\\ &\left. - 0.959e^{\frac{-S_2}{1.966H}}\right)\\ &(S_1S_2)^{0.5}\left(\frac{W_1}{H}\right)^{0.182}\end{aligned} \tag{8}$$

$$\begin{aligned}\frac{C_L}{\epsilon} &= A\left\{\frac{T_2}{S_2}\left(1 - 1.897e^{\frac{-H_2}{0.31S_2}-\frac{-T_2}{2.474S_2}}\right.\right.\\ &\left. + 1.302e^{\frac{-H_2}{0.082S_2}} - 0.1292e^{\frac{-T_2}{1.326S_2}}\right)\\ &+ 1.722\left(1-0.6548e^{\frac{-W_2}{0.3477H_2}}\right)e^{\frac{-S_2}{0.651H_2}}\Big\}\end{aligned}$$

$$\begin{aligned}&+ B\left\{\frac{T_2}{S_2}\left(1-1.897e^{\frac{-(W_2+2H_2)}{0.31S_2}-\frac{-T_2}{2.474S_2}}\right.\right.\\ &\left. + 1.302e^{\frac{-(W_2+2H_2)}{0.082S_2}} - 0.1292e^{\frac{-T_2}{1.326S_2}}\right)\\ &+ 1.722\left(1 - 0.6548e^{\frac{-W_2}{0.3477(W_2+2H_2)}}\right)\\ &e^{\frac{-S_2}{0.651(W_2+2H_2)}}\Big\}.\end{aligned} \tag{9}$$

$$A = \frac{R}{W_1 + W_2 + S_1 + S_2}. \tag{10}$$

$$B = \frac{O}{W_1 + W_2 + S_1 + S_2}. \tag{11}$$

$$R = W_1 + 2T_1 + W_3 + 2T_3 \quad for\ S_1 \geq 2T_1\ and\ S_3 \geq 2T_3. \tag{12}$$

$$R = W_1 + T_1 + W_3 + 2T_3 \quad for\ S_1 < 2T_1\ and\ S_3 \geq 2T_3. \tag{13}$$

$$R = W_1 + 2T_1 + W_3 + T_3 \quad for\ S_1 \geq 2T_1\ and\ S_3 < 2T_3. \tag{14}$$

$$R = W_1 + T_1 + W_3 + T_3 \quad for\ S_1 < 2T_1\ and\ S_3 < 2T_3. \tag{15}$$

$$O = S_1 - 2T_1 + S_3 - 2T_3 \quad for\ S_1 \geq 2T_1\ and\ S_3 \geq 2T_3. \tag{16}$$

$$O = S_3 - 2T_3 \quad for\ S_1 < 2T_1\ and\ S_3 \geq 2T_3. \tag{17}$$

$$O = S_1 - 2T_1 \quad for\ S_1 \geq 2T_1\ and\ S_3 < 2T_3. \tag{18}$$

$$O = 0 \quad for\ S_1 < 2T_1\ and\ S_3 < 2T_3. \tag{19}$$

ϵ : *Dielectric permittivity.*
W_n : *Wire width on layer n.*
S_n : *Wire spacing on layer n.*
T_n : *Wire thickness on layer n.*
H : *Dielectric thickness.*

(Received June 29, 2005)
(Accepted November 3, 2005)

Naoya Hatta is currently a M.E. student in Information and Communication Engineering in The University of Tokyo. His research interests are in vulnerability detection techniques.

Niko Demus Barli received the M.E. degree in Information Engineering from The University of Tokyo in 2001. He graduated with the Ph.D. degree in Information and Communication Engineering from The University of Tokyo in 2004. His graduate research mainly focused on speculative multithreading techniques on Chip Multiprocessors. He currently works for Texas Instruments Japan.

Chitaka Iwama received the M.E. degree in Information and Communication Engineering from The University of Tokyo in 2003. She is currently a student in School of Law in The University of Tokyo.

Luong Dinh Hung is currently a Ph.D. student in Information and Communication Engineering in The University of Tokyo. He received the M.E. degree in Information and Communication Engineering from The University of Tokyo in 2004. He actively pursues new ideas in the field of architecture and circuit techniques for VLSI power reduction.

Daisuke Tashiro received the M.E. degree in Information Engineering from The University of Tokyo in 2002. He graduated with the Ph.D. degree in Information and Communication Engineering from The University of Tokyo in 2005. His graduate research mainly focused on speculation and compiler techniques for speculative multithreading architectures. He currently works for Hitachi, Ltd.

Shuichi Sakai got B.S., M.S. and D.E. from The University of Tokyo in 1981, 1983 and 1986, respectively. He joined Electrotechnical Laboratory (ETL) in 1986. From 1991 to 1992, he was a visiting scientist in MIT. After working for ETL, RWC and University of Tsukuba, he became Associate Professor in University of Tokyo in 1999 and Professor in Graduate School of Information Science and Technology, The University of Tokyo in 2001. He got Japan IBM Award and IPSJ Best Paper Award in 1991, Ichimura-Award and IEEE Outstanding Paper Award in 1995. He is a member of IPSJ, IEICE, IEEE, ACM.

Hidehiko Tanaka graduated the Department of Electronic Engineering, University of Tokyo in 1965, completed the graduate course and received the degree of Ph.D. of Engineering in 1970. He was assigned as lecturer, associate professor and professor of Faculty of Engineering, The University of Tokyo, in 1970, 1971 and 1987 respectively. He was the Dean of Faculty of Information Science and Engineering, The University of Tokyo from 2001 to 2004. He is now the Dean of graduate school of the Institute of Information Security. He is interested in computer architecture, artificial intelligence, distributed processing and dependable information systems. He is the author of books such as "Non-Neuman Computer", "Computer Architecture", "VLSI Computer", "Parallel Inference Engine", etc.