A Swing Restored Pass-Transistor Logic-Based Multiply and Accumulate Circuit for Multimedia Applications

Akilesh Parameswar, *Member, IEEE,* Hiroyuki Hara, *Member, IEEE,* and Takayasu Sakurai, *Member, IEEE*

***Abstract*—Swing restored pass-transistor logic (SRPL), a high-speed, low-power logic circuit technique for VLSI applications, is described. By the use of a pass-transistor network to perform logic evaluation and a latch-type swing restoring circuit to drive gate outputs, this technique renders highly competitive circuit performance. An SRPL based multiply and accumulate circuit for multimedia applications is implemented in double metal 0.4 μm CMOS technology.**

I. Introduction

TO DATE, the most widely used VLSI circuit design technique has been full CMOS. It has been attractive because it makes it easy to implement reliable circuits that have excellent noise margins. However, the continuing push for higher performance systems has, in recent years, brought the disadvantages of full CMOS to the fore, and a number of researchers have proposed alternative logic techniques [1]–[3]. The majority of these have been static techniques because dynamic logic styles trade performance for charge sharing and noise margin issues and difficulties in design and design for testability.

Complimentary pass-transistor logic (CPL) [1] uses a complimentary output pass-transistor logic network to perform logic evaluation and CMOS inverters for driving of the outputs. This arrangement, however, can have leakage current through the inverter if the soft pull-up latch is not used. Double pass-transistor logic (DPL) [2] uses both pMOS and nMOS devices in the pass-transistor network to avoid nonfull swing problems, but it has high-area and high-power drawbacks. As the name suggests, differential cascode voltage switch with pass gate (DCVSPG) [3] is similar to the cascode voltage switch logic proposed in [4]. However, it shortens the stack height by the use of a pass-transistor network for logic evaluation and introduces a symmetrical logic topology in the true and complement logic evaluation trees. Unfortunately, this logic style can have degraded pull-down performance when used in a long chain without intermediate buffering.

Manuscript received January 19, 1995; revised March 5, 1996.

A. Parameswar is with Toshiba America Electronic Components, San Jose, CA 95131 USA.

H. Hara and T. Sakurai are with Toshiba Corporation, Saiwai-ku, Kawasaki, Japan 210.

Publisher Item Identifier S 0018-9200(96)04358-2.

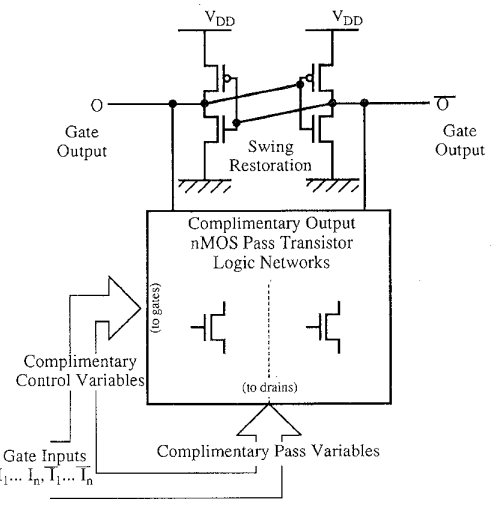

Fig. 1. Generic SRPL gate.

In this paper, we propose a high-speed, low-power logic circuit technique that attempts to overcome these problems.

II. Swing Restored Pass-Transistor Logic

A. Basic Circuit

The generic SRPL gate consists of two main parts as shown in Fig. 1—a complimentary output pass-transistor logic network that is constructed of *n*-channel devices and a latch-type swing restoring circuit consisting of two cross-coupled CMOS inverters. The gate inputs are of two types: pass variables that are connected to the drains of the logic network transistors and control variables that are connected to the gates of the transistors. The logic network has the ability to implement any random Boolean logic function. Fig. 2, for instance, shows the implementation of an SRPL full adder. The complimentary outputs of the pass-transistor logic network are restored to full swing by the swing restoration circuit.

B. Gate Optimization

We have found that in the interest of speed, the nMOS transistors of the logic network farther away from the output should have larger drivability (i.e., size) than those closer to the output. This is because the transistors closer

0018-9200/96$05.00 © 1996 IEEE

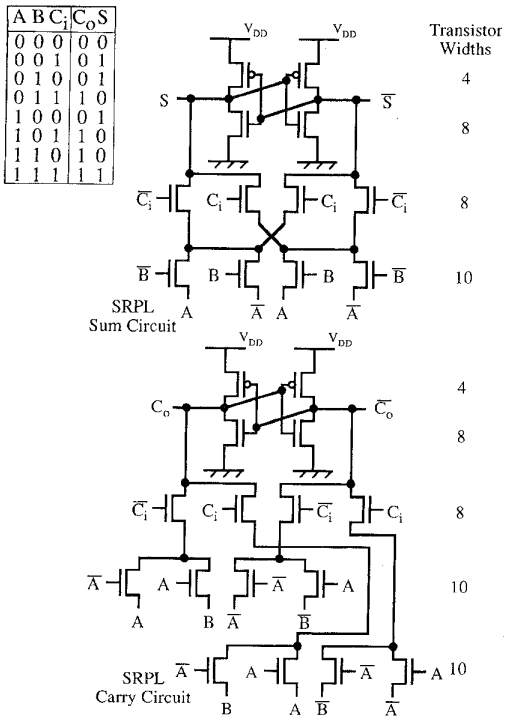

Fig. 2. Full adder circuit in SRPL.

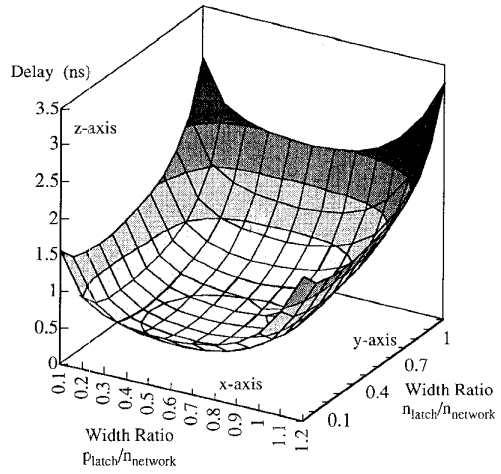

Fig. 3. Dependence of delay on transistor widths.

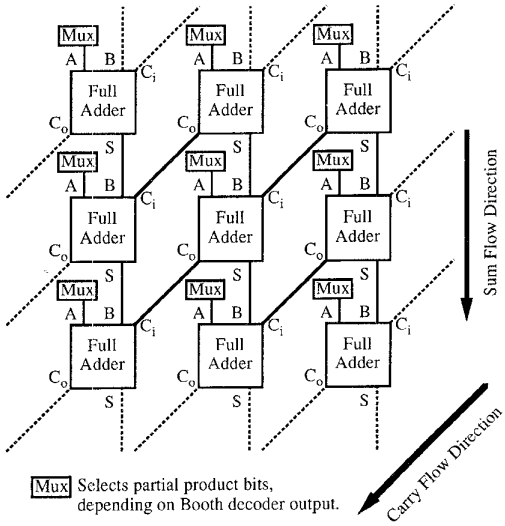

Fig. 4. Carry save addition of partial products.

to the output pass smaller swing high signals due to the voltage drop across the transistors farther away from the output. The precise values for a given circuit depend on layout and other circuit considerations, so they must be determined by case-by-case simulation. Possible values might be as indicated in Fig. 2.

The optimization of the swing restoring latch is an important determinant of overall gate speed. If high-speed latch inversion is required, the pMOS transistors should not be made too large. However, a large pMOS transistor size means that faster driving of a heavier load is possible. Hence, a trade-off exists, which is qualitatively demonstrated by the graph in Fig. 3. Simulations were performed on identical cascaded SRPL gates, each with a fanout of two, similar to the situation shown in Fig. 4. Gate outputs were assumed to connect to both pass and control inputs, and pass network transistor sizes were assumed to be similar to those of Fig. 2. Simulations were done with SPICE, using the parameters of a 0.4 μm CMOS process. The x-axis of Fig. 3 plots the ratio of the size of the pMOS transistor of the latch to the size of the topmost pass network nMOS transistor, while the y-axis plots the ratio of the size of the nMOS transistor of the latch to the size of the topmost pass network nMOS transistor. The z-axis plots the delay from the 0.5 V_{DD} mark of a pass input of the gate to the 0.5 V_{DD} mark of the output of the *subsequent* gate in the cascade.

For very small values of the $p_{latch}/n_{network}$ ratio, the gate output load becomes too large for the pMOS to be able to drive efficiently, and for very large values, the latch requires an inordinate amount of time to flip, reaching infinity (i.e., does not invert) over a certain limit. As Fig. 3 shows, there exists a further dimension to the optimization in that there is a trade-off in the values of the $n_{latch}/n_{network}$ ratio. If the latch nMOS device is too small in relation to the pass network nMOS device, discharging is bottlenecked, and the latch of the subsequent gate is not provided a firm pull-down path. However, if it is too large, the device introduces undue capacitive loading to the pull-up operation.

Notwithstanding these trade-offs, the bottommost (white) level of the graph shows that there exists a wide range of $p_{latch}/n_{network}$ and $n_{latch}/n_{network}$ for which the delay remains fairly stable. In other words, there exists substantial design margin, making it easy to design and lay out circuits in swing restored pass-transistor logic (SRPL). This design margin also means that SRPL circuits are quite robust against process variations, which might cause the threshold voltages of the transistors to fluctuate.

C. Power Dissipation, Capacitance, and Area Reduction

In a system of cascaded SRPL gates operating at a frequency of $f = 1/t$, the dynamic power dissipated by a gate, P_d, is given by the following equation:

$$P_d = \frac{C_{int,n}}{t}\int_{V_{DD}-V_{th,n}}^{V_{DD}} V_{\text{up}}\,dV_{\text{up}} + \frac{C_{\text{wire+gate,up}}}{t}\int_{0}^{V_{DD}} V_{\text{up}}\,dV_{\text{up}} + \frac{C_{int,n+1}}{t}\int_{0}^{V_{DD}-V_{th,n}} V_{\text{up}}\,dV_{\text{up}} + \frac{C_{int,n+1}}{t}\int_{V_{DD}}^{0} (V_{DD}-V_{\text{down}})\,d(V_{DD}-V_{\text{down}}) + \frac{C_{\text{wire+gate,down}}}{t}\int_{V_{DD}}^{0} (V_{DD}-V_{\text{down}}) \cdot d(V_{DD}-V_{\text{down}}). \quad (1)$$

The first three terms in the equation above represent the power dissipated by the charging inverter of the SRPL gate, whereas the last two terms represent the power dissipated by the discharging inverter. $C_{int,n}$ and $C_{int,n+1}$ represent the sum of the node capacitances internal to the pass-transistor logic blocks of the driving gate n and the receiving gate $n+1$, respectively, whereas the $C_{\text{wire+gate}}$ terms represent the sum of the wiring capacitance (to the next gate) and gate capacitance (of the next gate) seen by the output of the inverter. V_{up} and V_{down} are the output voltages of the driving gate, whereas V_{DD} and V_{th}, are the supply and n-transistor threshold voltages, respectively.

With respect to the charging side of the SRPL gate, the dynamic power dissipated is the sum of the power dissipated in raising to full V_{DD} the voltage at the internal nodes of the driving SRPL gate, the power dissipated in driving the output wiring and gate capacitance, and finally, the power dissipated in charging to $V_{DD} - V_{th}$, the internal nodes of the driven SRPL gate. With respect to the discharging side of the SRPL gate, the dynamic power dissipated is the sum of the power dissipated in discharging from V_{DD} the output wiring and gate capacitance and similarly discharging from V_{DD} the internal nodes of the driven SRPL gate.

If $C_{int,n}$ and $C_{int,n+1}$ are assumed to be approximately equal, and the $C_{\text{wire+gate}}$ terms are also assumed to be close, the equation above reduces to

$$P_d = f(C_{int} + C_{\text{wire+gate}})V_{DD}^2. \quad (2)$$

Hence, as with full CMOS, the power consumed is governed by a $P \propto fCV^2$ relationship. A smaller wire, gate, or internal capacitance will lead to a proportional decrease in power consumed. The structure of SRPL—whereby Boolean evaluation is done using a complimentary nMOS pass transistor network—lends itself naturally to low-wire, gate, and internal capacitance values. Wiring capacitance is lower than the CMOS because no connections to a pMOS network are required. The gate capacitances of the larger pMOS network are also absent, as are the larger parasitic capacitances of the driving full CMOS gate.

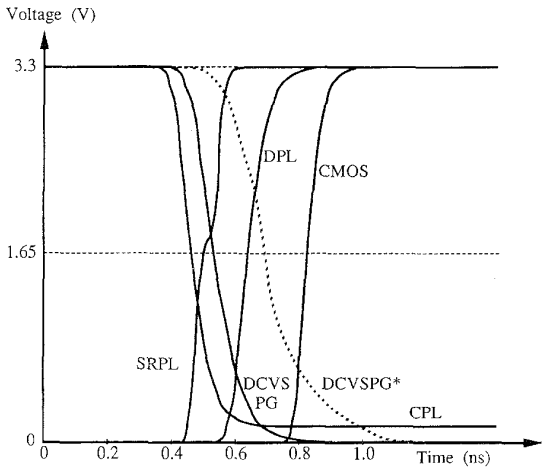

Fig. 5. Full adder worst case waveforms.

Area reduction is also facilitated by the structure of SRPL. As the numbers in the next section will show, lower transistor count is achieved because it is possible to share common Boolean terms within the complimentary nMOS network, something that cannot be done between the nMOS and pMOS networks of full CMOS. Moreover, area in full CMOS is wasted in separating the p/n well boundaries for each gate. This separation has to be done only for the two relatively small sized pMOS devices of each SRPL gate, subsequent to the nMOS pass transistor network.

Hence, because Boolean logic can be more efficiently represented in SRPL, the power consumed by a circuit of given functionality is much less than the full CMOS implementation as will be demonstrated in the next section.

D. Performance Comparison with Competing Techniques

Full adders in CMOS, CPL, DPL, DCVSPG, and SRPL were constructed and simulated in the cascaded conditions shown in Fig. 4. Again, 0.4 μm CMOS process parameters were used to perform SPICE simulations. The worst case waveforms for each of the full adders are shown in Fig. 5. Other performance values are recorded in Table I.

As Fig. 5 shows, CMOS has the slowest speed. Moreover, power consumption is quite high. The main reason for these poor performance figures is that the inefficient pMOS network of CMOS leads to a higher transistor count, larger gate area, and larger input capacitances due to the poor drivability of the pMOS transistor. DPL proves to be about 30% faster than CMOS, but this is at the expense of a higher transistor count and more power consumed. DCVSPG is much faster than CMOS, but may not

TABLE I
COMPARISON OF FULL ADDER CIRCUITS

	CMOS	CPL	DPL	DCVS PG	SRPL
Speed (ns)	0.82	0.44	0.63	0.53	0.48
Power at 100Mz (mW)	0.52	0.42	0.58	0.3	0.19
Power-Delay Product (normalized)	1.0	0.43	0.86	0.37	0.21
Transistor Count	40	28	48	24	28

be easy to use in regular array structures. The reason for this is that there is no pull-down mechanism other than that through the pass-transistor networks. Unless periodic CMOS buffering is provided in between long chains of cascaded gates, the pull down becomes degraded as shown by the dotted line of Fig. 5.

CPL, as Fig. 5 clearly shows, is the fastest of the five techniques. However, this is achieved at the expense of high-power consumption. Furthermore, CPL suffers from the drawback that it is a nonfull swing technique. The nonfull swing signals at the inputs of the inverters mean poor noise margins, particularly as the inverter threshold is susceptible to process variations. These process variations affect the inverter (pMOS) threshold independent of the affects on the pass-transistor network (nMOS) threshold, meaning that there could be significant margin degradation in the worst case.

Moreover, CPL circuits consume static power because of the leakage current that is always flowing through one of the inverters of a gate. The inverter output never quite reaches V_{SS} as the curve of Fig. 5 shows. Because a V_{DD} of 3.3 V is high relative to a channel width of 0.4 μm, the speed degrading effects of the leakage are not prominent. However, when V_{th} is a significant fraction of V_{DD}, as it will certainly be in the future, the fall time of the output lengthens and CPL becomes slower than SRPL.

SRPL has good speed performance. In the simulated conditions of Fig. 4, each SRPL circuit within the full adder fans out to only two other similarly sized circuits (carry and sum). This implies relatively light loading conditions, much less than the usual CMOS stage ratio of 3.5 or 4. It is important to note that this condition is not restricted to the simulated case. Low fanout is a very common occurrence in the design of VLSI circuits, particularly in data paths. In such conditions, it makes sense to connect the pass-transistor network output to the gate output and to restore the swing with the cross-coupled pair of inverters. The initial rise in voltage caused by the pass network output takes the gate output voltage a good margin above the $V_{th,n}$ of the transistors of the following gate, speedily setting up the correct logical path. Also, because of the relatively light loading conditions, the inversion of the latch is faster, and so the $p_{\text{latch}}/n_{\text{network}}$ ratio can be made slightly larger. Thus, a good pull-up time through the *a priori* set-up logical path of the following gate is achieved.

As Table I shows, SRPL has the lowest power consumption and the lowest power-delay product of the different techniques. The main reasons for the low power are the low transistor count and the low-input capacitance. Also, the fast inversion action of the latch quickly cuts off any dc path through the pass network.

E. Testability

Though it is believed in the field of custom IC design that the testability of pass-transistor networks is a problem, prior research [5], [6] on this issue has shown this not to be so. Reference [5], for instance, defines a function-preserving, failure-preserving transformation of a switching network into a logic network. The transformation maintains the failure structure of the original circuit and ordinary test pattern generation procedures on the transformed circuit yield tests that are automatically tests for failures in the original pass-transistor network.

In summary, SRPL shows itself to be a testable, low area, low-power, high-speed circuit technology. This promising logic technique was used to construct a multiply and accumulate circuit (MAC) for multimedia applications.

III. MULTIPLY AND ACCUMULATE CIRCUIT

The multiply and accumulate operation is crucial to a wide range of signal processing applications. With the increasing level of integration on processors, de/coders and special purpose IC's dedicated to multimedia, it has become essential that high-speed MAC macrocells be provided on chip. However, high speed is not the sole imperative. System portability is also a key issue, and hence, low power is also very important. The MAC presented in this paper was designed with these requirements foremost in mind.

A. MAC Architecture

The overall circuit is shown in Fig. 6. The multiplier and multiplicand are 16 b wide, whereas the accumulated result has a bit width of 32. A pipelined scheme was not implemented because the frequency of operation was expected to be more than sufficient to cover even the most advanced multimedia applications. Furthermore, pipelining introduces problems of complicated control and timing, extra area, and power required by the pipeline registers.

A Booth decoding scheme was used to obtain eight partial products, which are added in a carry-save manner as shown in Fig. 4. Each full adder row receives a running sum and carry from the row above. The very top adder of each column of the summation receives one of its inputs from the accumulated total of the previous cycle, which is fed back as shown in Fig. 6. A Wallace tree architecture for partial product addition was not used because such

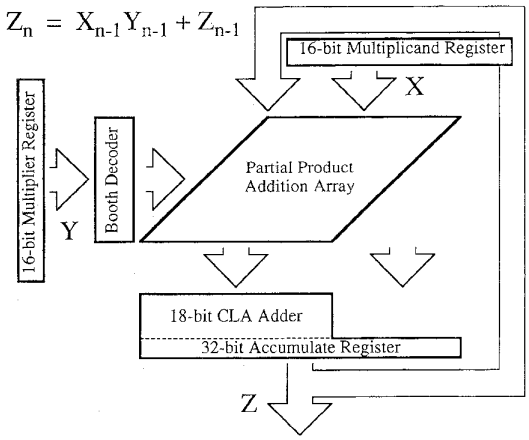

Fig. 6. Multiply and accumulate circuit.

an architecture would lead to larger power consumption due to the larger area and wiring requirements. Each of the full adders in the partial product summation array is constructed using the SRPL techique described above.

The final CLA adder plus register to which the partial product summation array outputs its carry-saved result uses the same design as that of [7], where a dynamic sense amplifying scheme is used to perform both carry propagation and latching of the final result. This design is ideally suited to the MAC design because of the high-speed addition followed by the instantaneous latching. The complimentary outputs of the SRPL-based summation array perfectly match the complimentary input requirements of the sense amplifying technique used by the final adder. It should be noted though that the dynamic sense amplifying techique used in [7] is quite different from the static swing restoring technique proposed in this paper.

B. Performance

The MAC was fabricated using a double metal 0.4 μm process as summarized in Table II. The chip photomicrograph is shown in Fig. 7. As Table II shows, the MAC operates at a maximum frequency of 150 MHz, which is more than sufficient for multimedia applications. Moreover, the power consumed is only 34 mW at this frequency, satisfying the other important multimedia requirement. The 150 MHz operating frequency translates to a one cycle delay time of 6.7 ns. For comparison, the MAC was simulated with a CPL partial product addition array. The simulated delay time was 6.3 ns.

Though the SRPL MAC is 0.4 ns slower than the CPL version, it should be remembered that CPL is the fastest technique ever reported, being nearly twice as fast as CMOS. Moreover, the power consumed by the CPL version was estimated to be more than twice that consumed by the SRPL MAC. In addition, as has been mentioned, CPL suffers from noise margin problems that will be exacerbated by the future reduction of the supply voltage, and this reduction in V_{DD} will also lead to speed degradation.

TABLE II
SUMMARY OF MAC CHARACTERISTICS

Technology	CMOS process
n Channel Length	0.4μm (Eff. 0.39μm)
p Channel Length	0.5μm (Eff. 0.47μm)
Gate Oxide Thickness	90 Å
No. of Metal Layers	2
Power Supply Voltage	3.3 Volts
Operating Frequency	150MHz
Latency	0 cycles
Power Consumed at 150MHz	34 mW
Transistor Count	12K
Active Area	0.98 mm^2

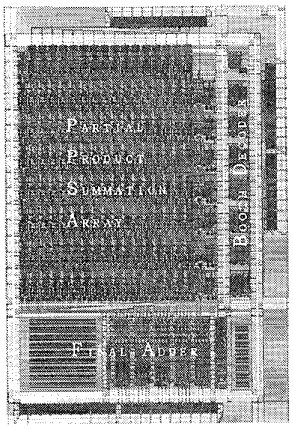

Fig. 7. MAC photomicrograph.

IV. CONCLUSION

A new high-speed, low-power logic circuit technology was proposed and used to implement a multiply and accumulate circuit in double metal 0.4 μm CMOS. The MAC achieves a frequency of 150 MHz while consuming 34 mW and shows much promise for multimedia applications.

ACKNOWLEDGMENTS

The authors would like to gratefully thank members of Toshiba Microelectronics Corp.: F. Sano for prodigious layout assistance and technical support, Y. Watanabe for design environment management, and K. Matsuda for help with chip testing. The authors are also deeply indebted to several individuals at Toshiba's Semiconductor Device Engineering Lab.: H. Koike, F. Matsuoka, and M. Kakumu, the process staff, for fabrication of the chip; M. Matsui for useful discussions and helpful suggestions; and K. Maeguchi, without whose blessings and encouragement, this work would not have been possible.

REFERENCES

[1] K. Yano *et al.*, "A 3.8 ns CMOS 16 × 16 multiplier using complimentary pass-transistor logic," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 388–395, Apr. 1990.

[2] M. Suzuki *et al.*, "A 1.5 ns 32-bit CMOS ALU in double pass-transistor logic," in *Proc. 1993 IEEE Int. Solid-State Circuits Conf.*, Feb. 1993, vol. 36, pp. 90–91.

[3] F. S. Lai and W. Hwang, "Differential cascode voltage switch with pass gate logic tree for high performance CMOS digital systems," in

Proc. 1993 Int. Symp. on VLSI Technology, Systems, and Applications, May 1993, pp. 358–362.

[4] L. G. Heller, W. R. Griffin, J. W. Davis, and N. G. Thoma, "Cascode voltage switch logic: A differential CMOS logic family," in *Proc. 1984 IEEE Int. Solid-State Circuits Conf.*, vol. 27, Feb. 1984, pp. 16–17.

[5] J. P. Roth, V. G. Oklobdzija, and J. F. Beetem, "Test generation for FET switching circuits," in *Proc. 1984 Int. Test Conf.*, Oct. 1984, pp. 59–62.

[6] V. G. Oklobdzija and P. G. Kovijanic, "On testability of CMOS domino logic," in *Proc. 14th Int. Conf. on Fault-Tolerant Computing*, June 1984, pp. 50–55.

[7] M. Matsui *et al.*, "200 MHz video compression macrocells using low-swing differential logic," in *Proc. 1994 IEEE Int. Solid-State Circuits Conf.*, Feb. 1994, vol. 37, pp. 76–77.

Akilesh Parameswar (S'89–M'91) received the B.Sc. (Honors) degree from the University of Zimbabwe, Harare, Zimbabwe, in 1988, and the M.Eng. degree from McMaster University, Hamilton, Ontario, Canada, in 1990.

In 1991, he joined the Semiconductor Device Engineering Laboratory of Toshiba Corp., Kawasaki, Japan, where he was involved in the circuit design of datapath macrocells of the R8000 processor. He was also engaged in research into low-power CMOS circuit techniques. In 1994, he moved to the Embedded Processors group of Toshiba America Electronic Components in San Jose, CA. His first project there was the development of a system ASIC for multimedia applications based on the R3900 processor. He is currently involved in the design of the memory management unit of Toshiba's next generation embedded processor.

Mr. Parameswar recently received TAEC's Core Values Award.

Hiroyuki Hara (M'87) was born on November 19, 1960, in Tokyo, Japan. He received the B.S. degree in electronic engineering from Shibaura Institute of Technology, Tokyo, Japan, in 1983.

In 1983, he joined Toshiba Corporation, Kawasaki, Japan, where he was engaged in the development and design of bipolar and BiCMOS LSI's. He is now in Toshiba's Semiconductor Device Engineering Laboratory, where he has been engaged in the research and development of BiCMOS macrocells for high-performance ASIC's. He has been working on the development of DCT macrocells and video compression/decompression LSI's. His present interests include low-power design.

Takayasu Sakurai (M'84) was born in Tokyo, Japan, in 1954. He received the B.S., M.S., and Ph.D. degrees in electronic engineering from the University of Tokyo, Tokyo, Japan, in 1976, 1978, and 1981, respectively. His Ph.D work was on electronic structures of an Si-SiO_2 interface.

In 1981, he joined the Semiconductor Device Engineering Laboratory, Toshiba Corporation, Kawasaki, Japan, where he was engaged in the research and development of CMOS dynamic RAM and 64 Kbit, 256 Kbit SRAM, 1 Mbit virtual SRAM, cache memories, and BiCMOS ASIC's. During the development, he also worked on the modeling of interconnect capacitance and delay, new memory architectures, hot-carrier resistant circuits, arbiter optimization, gate-level delay modeling, alpha/*n*th power MOS model, and transistor network synthesis. From 1988 through 1990, he was a visiting scholar at the University of California, Berkeley, doing research in the field of VLSI CAD. He is currently back at Toshiba managing multimedia LSI development. His present activities include low-power designs, media processors, and video compression/decompression LSI's. He is a visiting lecturer at Tokyo University and serves as a program committee member for the Custom Integrated Circuit Conference, the Design Automation Conference, the International Conference on Computer Aided Design, the International Conference on VLSI and CAD, the International Symposium on Low-Power Electronics and Design, and the FPGA Workshop. He is a technical committee chairperson for the '97 VLSI Circuits Symposium.

Dr. Sakurai is a member of the IEICEJ and the Japan Society of Applied Physics.