Multiple-Valued Radix-2 Signed-Digit Arithmetic Circuits for High-Performance VLSI Systems

doi: 10.1109/4.50294

Multiple-Valued Radix-2 Signed-Digit Arithmetic Circuits for High-Performance VLSI Systems

SHOJI KAWAHITO, MEMBER, IEEE, MICHITAKA KAMEYAMA, MEMBER, IEEE, AND TATSUO HIGUCHI, SENIOR MEMBER, IEEE

Abstract—**VLSI-oriented multiple-valued current-mode MOS arithmetic circuits using radix-2 signed-digit (SD) number representations are proposed. A prototype adder chip is implemented with 10-μm CMOS technology to confirm the principle operation. Furthermore, a new multiplication scheme using four-input current-mode wired summations is presented to realize a high-speed small-size multiplier. The designed 32×32-bit multiplier is composed of 18 800 transistors and required fewer interconnections. The multiply time is estimated to be 45 ns by SPICE simulation in 2-μm CMOS technology. It is shown that the developed technology is also potentially effective for the reduction of the data-bus area in VLSI.**

I. INTRODUCTION

IN VARIOUS VLSI systems for real-time applications, high-speed compact arithmetic circuits are required as macrocells. Special number representations, which are called signed-digit (SD) number representations, are useful for such high-speed arithmetic circuits [1]. It is well known that the addition can be performed in a constant time independent of the word length. Recently, several chips using the SD number representation have been reported [2]–[5]. Most of the developed chips use radix-2 SD (also called redundant binary) number representation and are implemented with ordinary binary logic gates. However, if the SD number representation is used at the system level, a problem arise along with the large data-bus area required for the redundant radix-2 SD number representation.

Multiple-valued logic circuit technology is another interesting approach in implementing SD arithmetic circuits because the SD number representation uses more than three values in each digit [6]. Moreover, the multiple-valued signals greatly reduce the number of interconnections. Except for memory chips, however, no multiple-valued VLSI chips have been fabricated. In this paper, we propose radix-2 SD arithmetic circuits based on the multiple-valued bidirectional current-mode MOS circuit technology [7]. The bidirectional current-mode circuit technology is attractive for implementing SD arithmetic circuits because the linear summation including polarity can be performed by wiring. We have already applied the technology to a 32×32-bit radix-4 SD multiplier [8]. The advantages in speed and compactness have been confirmed. The choice of radix 2 in this paper is because of its suitability for VLSI implementations and its speed performance [9]. In the VLSI implementation of the multiple-valued current-mode circuits, a decreased noise margin results from the channel-length modulation effect of the MOS devices. However, a sufficient noise margin can be obtained in multiple-valued radix-2 SD arithmetic circuits compared to higher radix SD arithmetic circuits.

For the first step in VLSI implementation of the multiple-valued arithmetic circuits, a prototype radix-2 SD parallel adder chip is designed and implemented. The adder can be realized with very few transistors, and it becomes quite simple using bidirectional current-mode wired summations. A multiplier is also designed using bidirectional current-mode circuits. A new modified binary-tree structure based on four-input addition of partial products is proposed. High-speed multiplication can be realized because the computation time is proportional to the logarithm of the word length of the operands. The current-mode wired summation can be fully used for the structure, so that the number of active devices and interconnections can be drastically reduced. Finally, we discuss VLSI implementation of the multiple-valued arithmetic circuits. The multiple-valued bidirectional current-mode circuit technology is particularly effective for the reduction of the data-bus area required for the SD number representation in VLSI.

Manuscript received July 19, 1989; revised September 18, 1989. A part of this research was supported by the Research Aid of Inoue Foundation for Science.

S. Kawahito is with the Department of Electrical and Electronic Engineering, Toyohashi University of Technology, Tempaku-cho, Toyohashi 440, Japan.

M. Kameyama and T. Higuchi are with the Department of Electronic Engineering, Tohoku University, Aoba, Aramaki, Aoba-ku, Sendai 980, Japan.

IEEE Log Number 8932959.

0018-9200/90/0200-0125$01.00 ©1990 IEEE

II. Radix-2 Signed-Digit Arithmetic Algorithm

A. Radix-2 SD Addition Algorithm

The radix-2 SD number representation utilized here is a redundant representation using a symmetrical digit set $\{-1,0,1\}$. Any integer X can be coded as a sequence of radix-2 signed digits x_i as follows:

$$X=(x_{n-1}\cdots x_i\cdots x_0)=\sum_{i=0}^{n-1}x_i2^i. \tag{1}$$

The number X is not uniquely coded in the SD number representation except for $X=0$. This redundancy allows totally parallel operation.

The addition of two numbers, $X=(x_{n-1}\cdots x_i\cdots x_0)$ and $Y=(y_{n-1}\cdots y_i\cdots y_0)$, where $x_i, y_i\in\{-1,0,1\}$, is performed by the following three successive steps in each digit:

$$\text{step 1:}\quad z_i=x_i+y_i \tag{2}$$

$$\text{step 2:}\quad 2c_i+w_i=z_i \tag{3}$$

$$\text{step 3:}\quad s_i=w_i+c_{i-1} \tag{4}$$

where $z_i\in\{-2,-1,0,1,2\}$, $w_i\in\{-1,0,1\}$, and $c_i\in\{-1,0,1\}$ are the linear sum, the intermediate sum, and the carry, respectively.

In order to retain the final s_i within the set $\{-1,0,1\}$, c_i and w_i are determined by z_{i-1} together with z_i subject to (3) as follows [10]:

$$\left.\begin{array}{lll} c_i=1, & w_i=0 & (\text{if } z_i=2)\\ c_i=1, & w_i=-1 & (\text{if } z_i=1 \text{ and } z_{i-1}\geq 1)\\ c_i=0, & w_i=1 & (\text{if } z_i=1 \text{ and } z_{i-1}<1)\\ c_i=0, & w_i=0 & (\text{if } z_i=0)\\ c_i=0, & w_i=-1 & (\text{if } z_i=-1 \text{ and } z_{i-1}\geq 1)\\ c_i=-1, & w_i=1 & (\text{if } z_i=-1 \text{ and } z_{i-1}<1)\\ c_i=-1, & w_i=0 & (\text{if } z_i=-2) \end{array}\right\}. \tag{5}$$

From (4) and (5), the final sum s_i is determined by z_i, z_{i-1}, and z_{i-2} independent of the other linear sum inputs. Therefore, the final sum can be obtained independent of the word length, because the carry-propagation chains are eliminated.

B. Multiplication Algorithm

In the multiplier considered here, two's complement binary number representations are used at the inputs and the output, and the radix-2 SD number representation is used for the internal expression to realize high-speed multiplication.

One of the outstanding features of the multiplication algorithm is the four-input addition of partial products; this speeds up the multiplication, and reduces the number of full-adder modules and interconnections. Since the inputs are two's complement binary number representations, the multiplicand X and the multiplier Y are expressed as

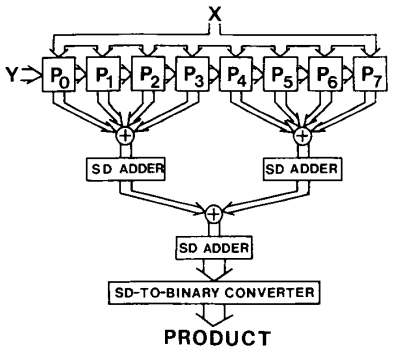


Fig. 1. Block diagram of the 8×8-bit SD multiplier.

$$X=-x_{m-1}2^{m-1}+\sum_{i=0}^{m-2}x_i2^i \tag{6}$$

$$Y=-y_{n-1}2^{n-1}+\sum_{j=0}^{n-2}y_j2^j \tag{7}$$

where n and m are even integers. The following expression of X is obtained by using the digit $\bar{x}_i=1-x_i$ and substituting $x_i=1-\bar{x}_i$ into (6):

$$X=\overline{x_{m-1}}2^{m-1}-\sum_{i=0}^{m-2}\bar{x}_i2^i-1. \tag{8}$$

The product P_j of X and an arbitrary digit y_j of Y is given by

$$P_j=y_jX=\sum_{i=0}^{n-1}p_{i,j}2^i \tag{9}$$

where $p_{i,j}$ is a partial product. When j is even, (6) is used for the generation of product P_j, and when j is odd, (8) is used. Thus, the partial products $p_{i,j}$ ($j=$ even) and $p_{i,j}$ ($j=$ odd), respectively, belong to the sets $\{0,1\}$ and $\{-1,0\}$ except for the most significant digits. Therefore, the linear sum $z_{i,j}$ of four successive partial products

$$z_{i,j}=p_{i,4j}+p_{i-1,4j+1}+p_{i-2,4j+2}+p_{i-3,4j+3} \tag{10}$$

obviously belongs to the set $\{-2,-1,0,1,2\}$. This means that four product operands, P_{4j}, P_{4j+1}, P_{4j+2}, and P_{4j+3}, can be added in parallel by using the radix-2 SD adder, because the linear sum of the SD adder allows values from -2 to 2 for each digit.

Another feature of the multiplier is the high-speed modified binary-tree structure. Fig. 1 shows a block diagram of the 8×8-bit multiplier. Since every four partial-product operands are added in parallel by using SD adders, the number of operands is reduced by one fourth at the outputs of the first-level SD adders. A product operand represented as an SD number is obtained by adding the two operands from the first level. At the final stage, the

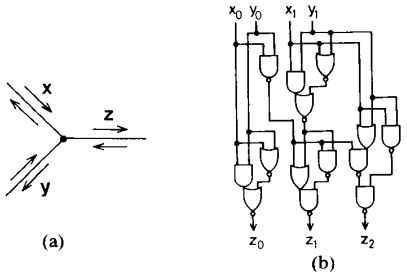


Fig. 2. Bidirectional current wired summation. (a) Wired summation. (b) Corresponding implementation with binary gates.

SD number is converted into a two's complement binary number as a final result. The conversion can be performed by high-speed two-input binary adder such as carry look-ahead adder [2]. Thus the multiplication can be achieved at high speed by the modified binary-tree scheme. The multiply time is proportional to the logarithm of the word length of operands. In this case, the final product is produced with two levels of SD adders and the SD-to-binary converter.

In order to reduce the number of partial products, binary multipliers often use the modified Booth's algorithm [11]. The algorithm can also be used in our multiplier. In the $n \times n$-bit multiplier, the number of full-adder levels necessary for reduction to a single product operand is given by

$$L = \lceil \log_2 (n/4) \rceil \tag{11}$$

where $\lceil a \rceil$ denotes the smallest integer such that $\lceil a \rceil \geq a$.

III. Circuit Design

A. Bidirectional Current-Mode Basic Circuits

The most attractive feature of the multiple-valued bidirectional current-mode MOS circuits is that linear summation including polarity can be performed by wiring. Fig. 2(a) illustrates the principle of the bidirectional wired summation. From Kirchhoff's current law, the current z is equal to the sum of the two currents x and y. We use the principle for the arithmetic linear summation, so that the resulting arithmetic circuits become quite simple. The corresponding implementation using ordinary binary logic gates is shown in Fig. 2(b) where $x, y \in \{-1,0,1\}$ and $z \in \{-2,-1,0,1,2\}$ are coded as 2-bit or 3-bit two's complement binary numbers. In Fig. 2(b), complicated circuits are necessary for the linear summation including polarity.

In order to realize the specific arithmetic circuits, several basic circuits are required together with wiring. Fig. 3 shows the available basic bidirectional current-mode circuits. We have already reported basic circuits suitable for radix-4 SD arithmetic chip [2]. In Fig. 3, several basic circuits are modified from the original version. Fig. 3(a) shows a current source using a p-channel depletion-mode MOSFET [2]. With the current source, the unit current I_a can be set at the specified value by threshold-voltage

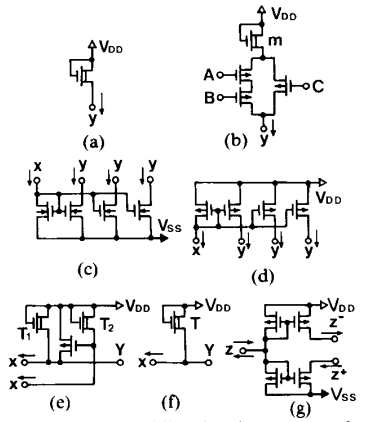


Fig. 3. Summary of the basic bidirectional current-mode circuits. (a) Depletion-mode PMOS current source. (b) Switched current source. (c) NMOS current mirror. (d) PMOS current mirror. (e) Threshold detector $TD(T_1, T_2; x)$. (f) Threshold detector $TD(T; x)$. (g) Bidirectional current input circuit.

control. This type of current source is quite insensitive to the fluctuation of the supply voltage V_{DD} and requires no bias source other than V_{DD}.

The switched current source can be realized by connecting pass transistors with the current source. Fig. 3(b) shows an example. The function is defined as

$$y = f(m; A \cdot B \upsilon C) = \begin{cases} 0, & (\text{if } A \cdot B \upsilon C = \text{“0”}) \\ mI_a, & (\text{if } A \cdot B \upsilon C = \text{“1”}) \end{cases} \tag{12}$$

where m, $\cdot$, and υ are a positive integer, logical AND operator, and logical OR operator, respectively, and A, B and C are two-valued voltage-mode signals in negative logic. The partial use of binary logical operations allows efficient design of the radix-2 SD arithmetic circuits.

The MOS current mirrors constitute the main basic circuits. Fig. 3(c) and (d) shows the NMOS and PMOS current mirrors producing three replicas of an input. The current mirrors are also used for inverting the current direction.

The threshold detection function can be realized with the principle of a current differencing amplifier. Fig. 3(e) shows the threshold detector whose function is given by

$$Y = TD(T_1, T_2; x) = \begin{cases} \text{“1,”} & (\text{if } T_1 < x < T_2) \\ \text{“0,”} & (\text{otherwise}). \end{cases} \tag{13}$$

The output is defined as negative logic. If $T_2 = \infty$, the schematic is simplified as shown in Fig. 3(f) and the function is denoted by $Y = TD(T; x)$. Fig. 3(e) is a newly introduced basic circuit. The resulting arithmetic circuits are simplified using the threshold detector of Fig. 3(e), as compared to the conventional method using only the threshold detector of Fig. 3(f).

The above basic circuits allow a single directional current at the inputs and outputs. An interfacing circuit between bidirectional and single-directional currents is required. The function is provided by the bidirectional current input circuit shown in Fig. 3(g). The schematic is

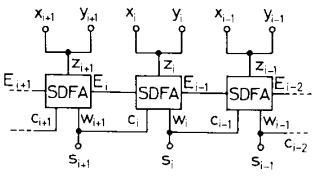


Fig. 4. Radix-2 parallel SD adder.

simpler than the original one which requires four additional transistors. Fig. 3(g) consists of NMOS and PMOS current mirrors. The threshold voltages of the transistors used for the current mirrors are set at

$$V_{Tn} = V_{DD}/2 \tag{14}$$

$$V_{Tp} = -V_{DD}/2 \tag{15}$$

where V_{Tn} and V_{Tp} are the threshold voltages of the NMOS and PMOS transistors, respectively. If the input current equals zero, both output currents of the NMOS and PMOS current mirrors are zero. Let the current flowing into the circuit be positive, while the current flowing out from the circuit be negative. If the input current is positive, the replica of the input current flows from the output of the NMOS current mirror and the output current of the PMOS current mirror remains zero. If the input current is negative, the replica of the input current flow from the output of the PMOS current mirror and the output current of the NMOS current mirror becomes zero. Thus the input bidirectional current is decomposed into positive and negative components.

B. Parallel SD Adder

By using the bidirectional current-mode circuit technology, the radix-2 parallel SD adder can be designed as shown in Fig. 4. Two linear summations of (2) and (4) can be performed by wiring without active devices. The operation of (3) is performed with a SD full adder (SDFA). The SDFA is designed based on (5), which is the detailed description of the SDFA. First, with the bidirectional current input circuit, the input signal z_i is decomposed into the positive component z_i^+ and negative component z_i^-, where

$$\left.\begin{array}{lll} z_i^+ = z_i, & z_i^- = 0 & (\text{if } z_i \geqslant 0) \\ z_i^+ = 0, & z_i^- = -z_i & (\text{if } z_i < 0) \end{array}\right\}. \tag{16}$$

Second, with the function of the threshold detectors the two-valued signals A_i^+, A_i^-, B_i^+ and B_i^- are defined as

$$A_i^+ = TD(0.5, 1.5; z_i^+) \tag{17a}$$

$$B_i^+ = TD(1.5; z_i^+) \tag{17b}$$

$$A_i^- = TD(0.5, 1.5; z_i^-) \tag{17c}$$

$$B_i^- = TD(1.5; z_i^-). \tag{17d}$$

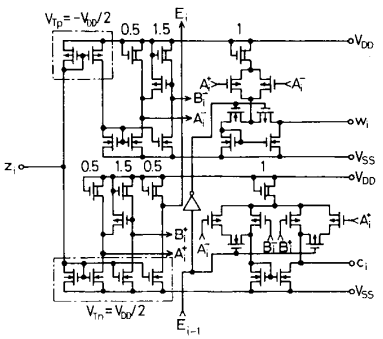


Fig. 5. Current-mode SD full adder (SDFA).

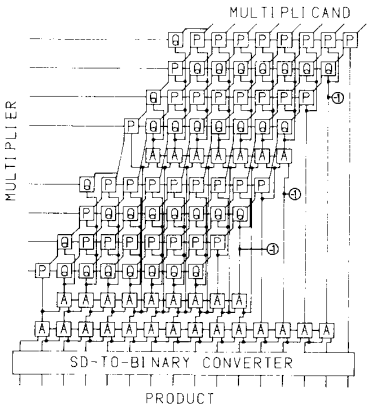


Fig. 6. Structure of the 8×8-bit SD multiplier.

The state signal E_i is also defined as

$$E_i = TD(0.5; z_i^+). \tag{17e}$$

From (5), (12), and (17), the carry c_i and w_i are written as

$$c_i = f(1; B_i^+ \cdot A_i^+ \vee E_{i-1}) - f(1; B_i^- \cdot A_i^- \vee \overline{E_{i-1}}) \tag{18}$$

$$w_i = f(1; (A_i^+ \vee A_i^-) \cdot \overline{E_{i-1}}) - f(1; (A_i^+ \vee A_i^-) \cdot E_{i-1}). \tag{19}$$

From (16)–(19), the SDFA is designed with 34 transistors as shown in Fig. 5. The structure of the adder is quite simple, and the number of interconnections and transistors is much fewer than in ordinary binary adders such as the carry lookahead adder.

C. Multiplier

Fig. 6 shows the structure of the 8×8-bit multiplier. The multiplier has a very regular structure, so that it is suitable for VLSI implementation. The inputs and output are of two's complement binary number representation. The P cells, Q cells, and A cells are the partial product generator (PPG) whose output is $\in \{0,1\}$, the PPG whose output is $\in \{-1,0\}$, and the SDFA, respectively. The schematics of the PPG's are quite simple, as shown in Fig. 7. The P cell of Fig. 7(a) is composed of six transistors, while the Q cell of Fig. 7(b) is composed of eight transistors, because an NMOS current mirror is necessary for inverting the current direction. Since the linear summation of every four

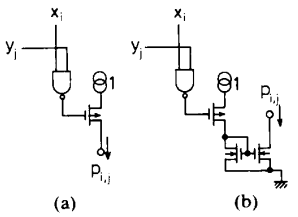


Fig. 7. Partial product generator (PPG). (a) P cell. (b) Q cell.

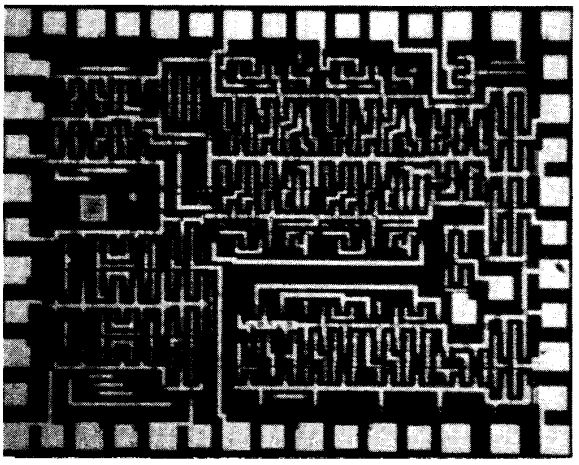


Fig. 8. Photomicrograph of the fabricated chip.

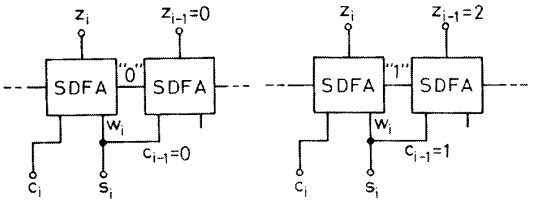
partial products is performed by wiring, interconnections between modules are simple and few. It is obvious that the multiplication is completed with two levels of SD adders and the SD-to-binary converter.

If the modified Booth's algorithm is used, the structure is further simplified because the number of partial products is reduced by half. In general, the schematic of the PPG is more complicated than the straightforward approach using the AND gate. However, since the reduction of the number of full adders also results from the use of the algorithm, the total number of transistors in the multiplier is greatly reduced.

IV. Results and Discussion

A. Implementation of Prototype Chip

To confirm the principal operation, a prototype 2-bit parallel SD adder chip is implemented using 10-μm CMOS technology. Fig. 8 shows the photomicrograph. The effective size of the SDFA module is $70\times160\lambda^2$, where $\lambda = 5\ \mu$m. A CMOS binary full adder is also designed based on the same design rule. The effective size is $66\times140\lambda^2$. The typical transfer characteristics for the case of $z_i \geq 0$ are shown in Fig. 9. Fig. 9(a) shows the characteristics and schematic when $z_{i-1}=0$. In this case, the final sum s_i is equal to w_i. The characteristics correspond to Table I, which shows the relationship between inputs and outputs of the SDFA when $z_{i-1}>1$. Fig. 9(b) shows the characteristics and schematic when $z_{i-1}=2$. In this case, the carry c_{i-1} equals 1, and the final sum s_i equals w_i+1. These characteristics correspond to Table II. Consequently, it is clear that the fabricated chip is operating well according to the additional algorithm mentioned above.



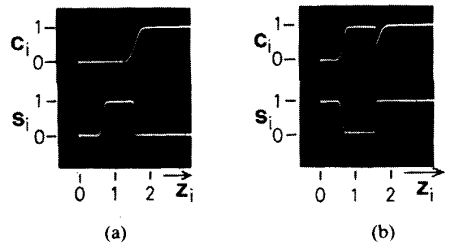


Fig. 9. Current transfer curves of the SD adder (24.4 μA/div). (a) $z_{i-1}=0$. (b) $z_{i-1}=2$.

TABLE I
MAPPING FOR c_i, w_i, AND s_i FROM z_i IN THE CASE OF $z_{i-1}\geq 1$

z_i	c_i	w_i	s_i
2	1	0	0
1	0	1	1
0	0	0	0
-1	-1	1	1
-2	-1	0	0

TABLE II
MAPPING FOR c_i, w_i, AND s_i FROM z_i IN THE CASE OF $z_{i-1}<1$

z_i	c_i	w_i	s_i
2	1	0	1
1	1	-1	0
0	0	0	1
-1	0	-1	0
-2	-1	0	1

B. Speed Evaluation Using SPICE2

In multiple-valued current-mode integrated circuits it is difficult to measure the intrinsic propagation delay time because the parasitic capacitance of the measurement tool affects the current waveform or response of the chip. In order to measure the accurate response, an on-chip encoder, decoder, and output buffer are necessary [12]. The fabricated chip does not include such measurement cir-

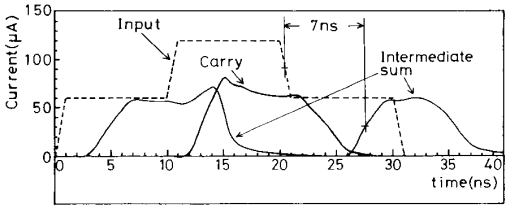


Fig. 10. Current response of the designed SDFA.

cuits. Therefore speed performance of the SDFA is estimated by SPICE simulation. Fig. 10 shows the simulated current response of the SDFA. In this simulation 2-μm LSI implementation is assumed. The broken line shows the input current signal, and the solid lines show the carry and intermediate sum output. The maximum delay time is estimated to be approximately 7 ns, which is faster than the 11 ns of the radix-4 SD adder that has been reported [7]. The improvement of the speed when compared to the radix-4 adder is due mainly to its simple structure. In the radix-4 adder, a quantizer circuit is required for the regeneration of the current level. The quantizer connected with the output of the full adder causes additional delay. On the other hand, the signal levels of the radix-2 SD adder are regenerated at the output without the quantizer.

The multiply time of the proposed multiplier is also estimated by simulation. In the 32×32-bit multiplier using the modified Booth's algorithm, the multiply time is given by

$$t_m = t_e + t_p + 3t_a + t_c \tag{20}$$

where t_e, t_p, t_a, and t_c are the maximum delay time of the encoder, the PPG, the SD full adder, and the SD-to-binary converter. By substituting these delay times, $t_e = 5$ ns, $t_p = 2$ ns, $t_a = 7$ ns, and $t_c = 17$ ns, obtained by simulation, and the multiply time is estimated to be 45 ns.

C. Comparison

The comparison of the proposed multiplier and the binary multiplier using the Wallace tree [13] is shown in Table III for a 32-bit precision. Both multipliers use the modified Booth's algorithm for the partial product generation. According to simulation, the proposed multiplier is faster than the binary multiplier. With regard to area, our multiplier is obviously superior to the binary multiplier because the number of transistors and the number of interconnections between modules are greatly reduced.

D. VLSI-Implementation Considerations

The high-speed regular tree structure of the SD multiplier is useful for the complicated arithmetic VLSI systems. For example, an arithmetic circuit for vector inner product $C = \boldsymbol{A} \cdot \boldsymbol{B}$, where $\boldsymbol{A} = (A_1, A_2, A_3, A_4)$ and $\boldsymbol{B} = (B_1, B_2, B_3, B_4)$, can be constructed as shown in Fig. 11. In the multipliers, the SD-to-binary converter is removed so that the addition of their outputs can be performed by the SD adder in a tree structure. In general, the advantage of

TABLE III
COMPARISON OF TWO MULTIPLIERS

Multiplier	Multiple-Valued Multiplier	Binary Wallace Tree Multiplier
Multiply Time	45*	56
Number of Transistors	18,800	45,000
Number of Full Adders	240	470
Number of Interconnections Between Modules	310	1,490

* SPICE Simulation

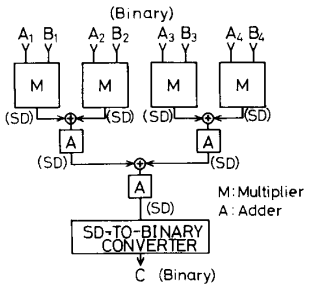


Fig. 11. Structure of the arithmetic circuits for vector inner product.

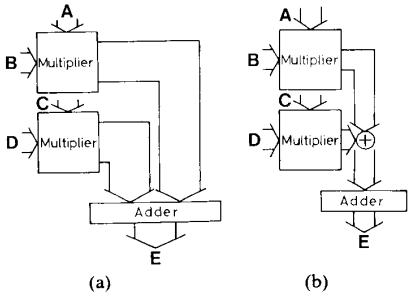


Fig. 12. Comparison of the data-bus area required for SD number representation. (a) With binary logic gates. (b) With multiple-valued current-mode logic circuits.

fast SD arithmetic may be lost by the conversion between the SD and binary number representations. If the SD-to-binary conversion is performed at the final stage of the system, the disadvantage of the conversion can be greatly reduced. The multiple-valued current-mode circuit approach is also effective for such complicated VLSI systems using SD number representation. Fig. 12 illustrates the data-bus area for connecting the outputs of the two SD multipliers with the input of the SD adder. If the system is implemented with binary gates, a large data-bus area is required. For instance, in 32-bit precision, 64 data lines are necessary because of the redundancy of the SD number representation. On the other hand, with the multiple-valued current-mode approach, the data-bus area can be reduced by at least one-fourth. This is because not only is the information delivered by multiple levels of signals (five levels at maximum), but also the information is compressed by the linear summation.

The VLSI implementation of the proposed multiple-valued arithmetic circuits using submicrometer technology still involves a few problems. For instance, a decreased noise immunity results from the channel-length modulation effect of the device used for the current source and current mirrors. One solution for this problem is to use a special device with the reduced channel-length modulation effect. For instance, a diffusion self-aligned MOS device which has a double-diffused structure at the source region [14] will be useful. If the problems are successfully solved, the multiple-valued current-mode circuit technique is expected to be a key technology in overcoming the problem of the complicated wiring in VLSI.

V. Conclusion

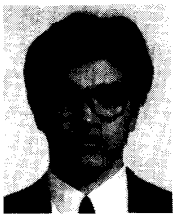
High-speed, compact, radix-2, signed-digit arithmetic circuits suitable for macrocells in VLSI are proposed. The multiple-valued bidirectional current-mode circuit technology is essentially suitable for the compact implementation. The successful operation of the implemented prototype adder chip is confirmed. The current-mode wired summation is effective not only for the realization of the high-speed compact multiplier, but also for the reduction of the data-bus area in the VLSI system.

Acknowledgment

The authors wish to thank Associate Professor M. Esashi and Dr. S. Shoji of Tohoku University for helpful comments on the chip fabrication. The authors would also like to thank Professor T. Nakamura of Toyohashi University of Technology for helpful discussions.

References

[1] A. Avizienis, "Signed-digit number representations for fast parallel arithmetic," *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 389–400, Sept. 1961.
[2] S. Kawahito *et al.*, "A 32×32 bit multiplier using multiple-valued MOS current-mode circuits," *IEEE J. Solid-State Circuits*, vol. 23, no. 1, pp. 124–132, Feb. 1988.
[3] H. Yoshimura *et al.*, "A 50 MHz CMOS geometrical mapping processor," in *ISSCC Dig. Tech. Papers*, Feb. 1988, pp. 162–163.
[4] H. Edamatsu *et al.*, "A 33 MFLOPS floating-point processor using redundant binary representation," in *ISSCC Dig. Tech. Papers*, Feb. 1988, pp. 152–153.
[5] M. Yamashina *et al.*, "200-MHz 16-bit BiCMOS signal processor," in *ISSCC Dig. Tech. Papers*, THPM12.8, Feb. 1989.
[6] M. Kameyama and T. Higuchi, "Design of radix-4 signed-digit arithmetic circuits for digital filtering," in *Proc. 1980 Int. Symp. Multiple-Valued Logic*, June 1980, pp. 272–277.
[7] S. Kawahito, M. Kameyama, and T. Higuchi, "VLSI-oriented bi-directional current-mode arithmetic circuits based on the radix-4 signed-digit number system," in *Proc. Int. Symp. Multiple-Valued Logic*, May 1986, pp. 70–77.
[8] M. Kameyama, S. Kawahito, and T. Higuchi, "A multiplier chip with multiple-valued bidirectional current-mode logic circuits," *IEEE Computer*, vol. 21, no. 4, pp. 43–56, Apr. 1988.
[9] S. Kawahito *et al.*, "High-performance multiple-valued radix-2 signed-digit multiplier and its application," in *Proc. Symp. VLSI Circuits*, May 1989, pp. 125–126.
[10] N. Takagi *et al.*, "High-speed VLSI multiplication algorithm with a redundant binary addition tree," *IEEE Trans. Comput.*, vol. C-34, 9, pp. 789–796, Sept. 1985.
[11] L. P. Rubinfield, "A proof of the modified Booth's algorithm for multiplication," *IEEE Trans. Comput.*, vol. C-24, no. 10, pp. 1014–1015, Oct. 1975.
[12] M. Kameyama *et al.*, "Bidirectional current-mode basic circuits for the multiple-valued signed-digit arithmetic and their evaluation," *Trans. IEICE Japan* (in Japanese), J71-D, 7. pp. 1189–1198, July 1988.
[13] A. E. Gammel *et al.*, "A CMOS 32b Wallace tree multiplier-accumulator," in *ISSCC Dig. Tech. Papers*, THPM15.5, 1986.
[14] Y. Tarui *et al.*, "Diffusion self-aligned MOST—A new approach for high speed devices," in *Proc. 1st Conf. Solid-State Devices*, 1970, pp. 105–110.



Shoji Kawahito (S'86–M'88) was born in Tokushima, Japan, on March 21, 1961. He received the B.E. and M.E. degrees in electronic engineering from Toyohashi University of Technology, Toyohashi, Japan, in 1983 and 1985, respectively, and the D.E. degree from Tohoku University, Sendai, Japan, in 1988.

He is currently a Research Associate with the Department of Electrical and Electronic Engineering, Toyohashi University of Technology. His research interests include multiple-valued arithmetic VLSI, integrated smart sensors, and three-dimensional integration.

Dr. Kawahito is a member of the Institute of Electronics, Information and Communication Engineers of Japan. He received the Outstanding Paper Award at the 1988 IEEE International Symposium on Multiple-Valued Logic (with T. Higuchi *et al.*).

Michitaka Kameyama (M'79) was born in Utsunomiya, Japan, on May 12, 1950. He received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1973, 1975, and 1978, respectively.

He is currently an Associate Professor in the Department of Electronic Engineering, Tohoku University. His general research interests include multiple-valued digital processing systems, VLSI systems, highly reliable digital systems, and robot electronics.

Dr. Kameyama is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Society of Instrument and Control Engineers of Japan, the Information Processing Society of Japan, and the Robotics Society of Japan. He received the Outstanding Paper Awards at the 1984, 1986, and 1988 IEEE International Symposiums on Multiple-Valued Logic (with T. Higuchi *et al.*), and the Technically Excellent Award from the Society of Instrument and Control Engineers of Japan in 1986 (with T. Higuchi *et al.*). He was the Program Co-chairman of the 1986 IEEE International Symposium on Multiple-Valued Logic.

Tatsuo Higuchi (M'70–SM'83) was born in Sendai, Japan, on March 30, 1940. He received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1962, 1964, and 1969, respectively.

He is currently a Professor with the Department of Electronic Engineering, Tohoku University. His research interests include the design of one- and two-dimensional finite word-length digital filters, multiple-valued digital processing systems, fault-tolerant computing, and VLSI computing structures for signal processing and image processing.

Dr. Higuchi is a member of the Institute of Electrical Engineers of Japan, the Institute of Electronics, Information and Communication Engineers of Japan, and the Society of Instrument and Control Engineers of Japan. He received the Outstanding Paper Awards at the 1984, 1986, and 1988 IEEE International Symposiums on Multiple-Valued Logic (with M. Kameyama *et al.*), the Outstanding Transactions Paper Award from the Society of Instrument and Control Engineers of Japan in 1984 (with M. Kawamata), and the Technically Excellent Award from the Society of Instrument and Control Engineers of Japan in 1986 (with M. Kameyama *et al.*). He was the Program Chairman of the 1983 IEEE International Symposium on Multiple-Valued Logic, and the Chairman of the Japan Research Group on Multiple-Valued Logic from 1983 to 1985.