Highly Parallel Residue Arithmetic Chip Based on Multiple-Valued Bidirectional Current-Mode Logic

Highly Parallel Residue Arithmetic Chip Based on Multiple-Valued Bidirectional Current-Mode Logic

MICHITAKA KAMEYAMA, MEMBER, IEEE, TSUTOMU SEKIBE, AND TATSUO HIGUCHI, SENIOR MEMBER, IEEE

Abstract—**This paper discusses the implementation of a residue arithmetic circuit using multiple-valued bidirectional current-mode MOS technology. Each residue digit is represented by new multiple-valued coding suitable for highly parallel computation. By the coding, mod m_i multiplication can be simply performed by a shift operation. In mod m_i addition, radix-5 signed-digit (SD) arithmetic is employed for a high degree of parallelism and multiple-operand addition, so that high-speed arithmetic operations can be achieved. Finally, the mod 7 three-operand multiply adder is designed and fabricated as an integrated circuit based on 10-μm CMOS technology.**

I. Introduction

THE DEMANDS for high-speed computations are obvious in many real-time applications such as ultra-high-speed signal processing and digital control systems. Many VLSI architectures have been proposed to perform real-time computation with large amounts of data. However, arithmetic operating speed is restricted by carry propagation in conventional binary systems, so that a new computer arithmetic system which provides faster computing speed than a conventional system is expected.

The residue number system (RNS) is of particular interest because of the inherent property that addition and multiplication are executed very fast without the need for carry propagation [1]. These residue arithmetic operations have usually been implemented by storing mod m_i arithmetic tables in read-only memories (ROM's), and sustain operations limited only by the access time of ROM [2]. However, this implementation is not always appropriate with respect to compactness and speed because a large number of ROM's is required to store all arithmetic tables, and the access time of ROM is not enough for some high-speed applications.

This paper presents a new residue arithmetic circuit based on multiple-valued coding and multiple-valued bidirectional current-mode MOS technology [3]. The RNS is essentially a number system which corresponds directly with multiple-valued logic system since each digit takes on m_i values. It has been well known that the use of multiple-valued logic has potential advantages [4], [5], so that multiple-valued coded RNS has been investigated for effective implementation [6]–[8]. However, the implementation of m_i-valued logic circuits with m_i signal levels is very difficult when m_i is large. Here, a new multiple-valued coded residue digit representation is introduced based on the pseudo-primitive root. With this coding, mod m_i multiplication and mod m_i addition can be executed using only shift and radix-5 arithmetic operations, respectively. Furthermore, mod m_i multiplication by a constant coefficient can be performed simply by exchanging wire connections.

Multiple-valued bidirectional current-mode MOS technology is employed to implement the required multiple-valued coded residue arithmetic circuits [9], [10]. Bidirectional current-mode circuits are suitable for arithmetic operations in symmetric number systems because linear addition including polarity can be performed by wiring. Through the use of these circuits, a radix-5 signed-digit full adder (SDFA) can be constructed easily. In a radix-5 SDFA, carry propagation is always limited to one position independently of the word length, so that mod m_i addition can be performed for each residue digit in parallel. Multiple-operand addition is also effective for high-speed computation [11]. In the residue arithmetic circuit described, three-operand addition can be performed simultaneously using the redundancy of the multiple-valued coding.

In order to confirm the principle operations, the mod 7 three-operand multiply adder composed of 190 transistors has been designed and fabricated in 10-μm CMOS technology. This arithmetic circuit has a regular array structure which offers the potential for compact VLSI implementation.

Manuscript received January 3, 1989; revised April 7, 1989.

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

T. Sekibe was with the Department of Electronic Engineering, Tohoku University, Sendai 980, Japan. He is now with the Matsushita Electric Industrial Company, Ltd., Osaka, Japan.

IEEE Log Number 8929555.

II. Symmetric Residue Number System

The symmetric RNS is constructed from a set of relatively prime odd-numbered moduli, $\beta = \{m_0, \cdots, m_i, \cdots, m_{N-1}\}$, where the residue digit with respect to a modulus

0018-9200/89/1000-1404$01.00

m_i is represented by the symmetric set [1]:

$$L_{m_i} = \{-(m_i-1)/2, \cdots, 0, \cdots, (m_i-1)/2\}. \quad (1)$$

Any integer $x \in [-(M-1)/2, (M-1)/2]$, where

$$M = \prod_{i=0}^{N-1} m_i$$

can be uniquely coded as a sequence of residue digit x_i according to

$$x = (x_0, \cdots, x_i, \cdots, x_{N-1})$$
$$x_i = |x|_{m_i} = x - [x/m_i] \cdot m_i \quad (2)$$

where $[x/m_i]$ is x/m_i rounded to the closest integer in the symmetric RNS, and each residue digit is defined to be the remainder of least magnitude when x is divided by m_i.

In the symmetric RNS, the addition and multiplication of two numbers $x = (x_0, \cdots, x_i, \cdots, x_{N-1})$ and $y = (y_0, \cdots, y_i, \cdots, y_{N-1})$ are performed by the following steps in the same way as in the ordinary RNS:

$$x + y = (x_0 \oplus y_0, \cdots, x_i \oplus y_i, \cdots, x_{N-1} \oplus y_{N-1})$$
$$x \cdot y = (x_0 \odot y_0, \cdots, x_i \odot y_i, \cdots, x_{N-1} \odot y_{N-1}) \quad (3)$$

where $\oplus$ and $\odot$ denote mod m_i addition and mod m_i multiplication, respectively, in the symmetric digit set of (1). Each residue digit can be computed independent of all others (i.e., carry-free arithmetic). This separability of binary operations allows for very fast parallel implementation.

In this number system, symmetry is retained as indicated in (4) for a sign conversion:

$$-x = (-x_0, \cdots, -x_i, \cdots, -x_{N-1}). \quad (4)$$

III. Multiple-Valued Coded Residue Number System

In the residue arithmetic, we must manipulate the numbers whose range is not a power of 2, and whose digits are not ordered. This speciality causes difficulty in hardware implementation. ROM implementation is not always appropriate with respect to compactness and speed. Direct implementation, in which each residue digit needs m_i levels, is very difficult in present LSI process technologies.

Here, we discuss a new approach based on multiple-valued coded residue digit representation. The use of multiple-valued logic enables simple and highly parallel implementation of residue arithmetic operations. In order to achieve effective multiple-valued coding, a concept of the pseudo-primitive root is introduced as follows.

Definition 1: Let the powers of an integer p in mod m_i representation be $|p^0|_{m_i}, |p^1|_{m_i}, \cdots$, and $|p^{(m_i-3)/2}|_{m_i}$. If their magnitudes include every value of $1, 2, \cdots$, and $(m_i-1)/2$, then p is called the pseudo-primitive root in mod m_i.

Example 1: Let $m_i = 7$ and $p = 5$. Then

$$|5^0|_7 = 1, \quad |5^1|_7 = -2, \quad \text{and} \quad |5^2|_7 = -3. \quad (5)$$

It is clear that 5 is the pseudo-primitive root in mod 7.

The multiple-valued coded residue representation based on the concept of the pseudo-primitive root is defined as follows.

Definition 2: If either $|5^n|_{m_i} = 1$ or $|5^n|_{m_i} = -1$ for $n = (m_i-1)/2$ and if $|5^k|_{m_i} \neq \pm 1$ for all integers k such that $0 < k < n$ (i.e., 5 is the pseudo-primitive root in mod m_i), then the residue digit x_i in mod m_i is coded as

$$x_i = \sum_{j=0}^{n-1} x_{ij} 5^j \quad (\text{mod } m_i) \quad (6)$$

$$x_{ij} \in \{-2, -1, 0, 1, 2\}. \quad (7)$$

In practical arithmetic circuits, the pseudo-primitive root $p = 5$ is appropriate for the radix of the multiple-valued coded residue digit representation, because:

1) radix-5 arithmetic circuit can be implemented easily with multiple-valued bidirectional current-mode circuits;
2) redundancy present in the multiple-valued coding with $p = 5$ enables simultaneous three-operand addition as shown later; and
3) $p = 5$ is applicable to many moduli m_i, such as

m_i=7,11,17,19,23,37,43,47,53,59,73,79,83,97.

For example, the residue digit in mod 7 is written as (8) using (5). All of the coefficients in (8) have magnitudes of 1, 2, and 3:

$$x_i = |5^2 x_{i2} + 5^1 x_{i1} + 5^0 x_{i0}|_7$$
$$= |-3x_{i2} - 2x_{i1} + x_{i0}|_7. \quad (8)$$

IV. Hardware Algorithm of Residue Arithmetic Operations

A. Mod m_i Multiplication

In the above residue digit coding, mod m_i multiplication can be performed simply by a shift operation, because all multipliers in the residue representation correspond to a power of 5. However, sign conversion is often required since both negative and positive values are included in a single digit. For example, mod 7 multiplication is shown in Table I. Since $5^0 = 1$, $5^1 = -2$, and $5^2 = -3$ in mod 7, the multiplication by a multiplier c_i is performed by the following operations:

$$c_i = \begin{cases} 3 \cdots \text{two-digit shift left and sign conversion} \\ 2 \cdots \text{one-digit shift left and sign conversion} \\ 1 \cdots \text{no operation} \\ 0 \cdots 0 \text{ for all digits} \\ -1 \cdots \text{sign conversion} \\ -2 \cdots \text{one-digit shift left} \\ -3 \cdots \text{two-digit shift left.} \end{cases}$$

TABLE I
MOD 7 MULTIPLICATION

$\odot$	-3	-2	-1	1	2	3
$5^0=1$	5^2	5^1	-5^0	5^0	-5^1	-5^2
$5^1=-2$	-5^0	5^2	-5^1	5^1	-5^2	5^0
$5^2=-3$	-5^1	-5^0	-5^2	5^2	5^0	5^1

Moreover, multiplication by a constant coefficient can be realized simply by exchanging the wire connection.

B. Mod m_i Addition

Radix-5 SD arithmetic is employed in mod m_i addition [9]. In radix-5 SD arithmetic, the addition of two numbers

$$x_i=(x_{i\,n-1},\cdots,x_{ij},\cdots,x_{i0})=\sum_{j=0}^{n-1}x_{ij}5^j$$

and

$$y_i=(y_{i\,n-1},\cdots,y_{ij},\cdots,y_{i0})=\sum_{j=0}^{n-1}y_{ij}5^j$$

is performed by following three successive steps in each digit:

$$z_{ij}=x_{ij}+y_{ij} \tag{9}$$

$$5c_{ij}+w_{ij}=z_{ij} \tag{10}$$

$$\left.\begin{array}{llll} w_{ij}=z_{ij}-5 & \text{and} & c_{ij}=1 & \text{if } z_{ij}>2 \\ w_{ij}=z_{ij} & & c_{ij}=0 & \text{if } -2\leqslant z_{ij}\leqslant 2 \\ w_{ij}=z_{ij}+5 & \text{and} & c_{ij}=-1 & \text{if } z_{ij}<-2 \end{array}\right\} \tag{10a}$$

$$s'_{ij}=w_{ij}+c_{i\,j-1} \tag{11}$$

where z_{ij}, w_{ij}, and c_{ij} are, respectively, the linear sum of x_{ij} and y_{ij}, a partial sum, and a carry, and where

$$z_{ij}\in\{-6,\cdots,0,\cdots,6\} \tag{12}$$

$$w_{ij}\in\{-2,-1,0,1,2\} \tag{13}$$

$$c_{ij}\in\{-1,0,1\} \tag{14}$$

$$s'_{ij}\in\{-3,\cdots,0,\cdots,3\}. \tag{15}$$

The sum s'_{ij} can be obtained almost in parallel and independently of the word length.

Because of the special property of the pseudo-primitive root, $|5^n|_{m_i}=1$ or $|5^n|_{m_i}=-1$ in the multiple-valued coded residue representation. Then, the carry c_{in-1} from the most-significant digit can be connected to the least-significant digit as shown in (16):

$$s'_{i0}=\begin{cases} w_{i0}+c_{i\,n-1}, & \text{if } |5^n|_{m_i}=1 \\ w_{i0}-c_{i\,n-1}, & \text{if } |5^n|_{m_i}=-1. \end{cases} \tag{16}$$

The linear sum of the inputs z_{ij} and the final sum s'_{ij} are determined by (12) and (15) for two-operand addition.

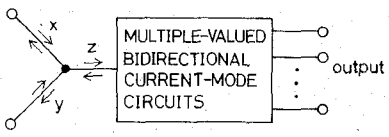


Fig. 1. Principle of bidirectional current-mode circuits.

In order to realize three-operand addition, s'_{ij} is converted into s_{ij} having the range stated in (7) according to (17) and (18):

$$s'_{ij}=\begin{cases} q_{ij}+2q'_{ik}, & \text{if } |2\cdot 5^j|_{m_i}=|5^k|_{m_i} \\ q_{ij}-2q'_{ik}, & \text{if } |2\cdot 5^j|_{m_i}=-|5^k|_{m_i} \end{cases} \tag{17}$$

$$s_{ij}=q_{ij}+q'_{ij} \tag{18}$$

where q_{ij}, $q'_{ik}\in\{-1,0,1\}$. Therefore, $s_{ij}\in\{-2,-1,0,1,2\}$.

Example 2: Consider the multiple-valued coded residue digit representation in mod 7 as shown in (8).

Let $(s'_{i2},s'_{i1},s'_{i0})=(3,-2,3)$. Since $|2\cdot 5^0|_7=-|5^1|_7$, $|2\cdot 5^1|_7=-|5^2|_7$, and $|2\cdot 5^2|_7=|5^0|_7$, each s'_{ij} is converted into s_{ij} as follows. According to (17), we have, for each s'_{ij}:

$$s'_{i0}=3=q_{i0}-2\cdot q'_{i1}$$

$$s'_{i1}=-2=q_{i1}-2\cdot q'_{i2}$$

$$s'_{i2}=3=q_{i2}+2\cdot q'_{i0}$$

where q_{ij} and $q'_{ik}\in\{-1,0,1\}$ are given by $q_{i0}=1$, $q'_{i1}=-1$, $q_{i1}=0$, $q'_{i2}=1$, $q_{i2}=1$, and $q'_{i0}=1$. From (18)

$$s_{i0}=q_{i0}+q'_{i0}=(1)+(1)=2$$

$$s_{i1}=q_{i1}+q'_{i1}=(0)+(-1)=-1$$

$$s_{i2}=q_{i2}+q'_{i2}=(1)+(1)=2$$

hence

$$(s_{i2},s_{i1},s_{i0})=(2,-1,2).$$

Since every s_{ij} satisfies the conditions given in (7), it is possible to realize three-operand addition.

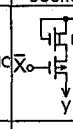
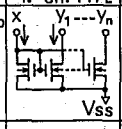
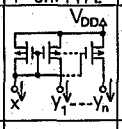
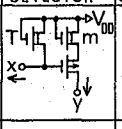
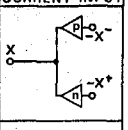
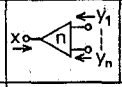
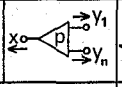
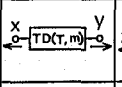
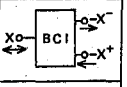
V. Residue Arithmetic Circuits Using Multiple-Valued Bidirectional Current-Mode MOS Technology

A. Multiple-Valued Bidirectional Current-Mode MOS Technology

Fig. 1 illustrates the principle of bidirectional current-mode circuits. From Kirchhoff's current law, the current z is equal to the sum of the two currents x and y. The current z is applied to successive bidirectional current-mode circuits, where the polarity and the current level are detected and arithmetic operations are performed using several basic circuits.

Bidirectional current-mode circuits are suitable for the multiple-valued coded RNS because both negative and

TABLE II
BASIC BIDIRECTIONAL CURRENT-MODE CIRCUITS

BASIC CIRCUIT	CURRENT SOURCE	CURRENT MIRRORS N-CH. TYPE	CURRENT MIRRORS P-CH. TYPE	THRESHOLD DETECTOR	BIDIRECTIONAL CURRENT INPUT
SCHEMATIC					
SYMBOL					
FUNCTION	$\begin{cases} Y=0 & \text{if } \bar{X}=\text{"1"} \\ Y=m & \text{if } \bar{X}=\text{"0"} \end{cases}$	$Y_i=-a_i x$ for $i=1,\ldots,n$ a_i: SCALE FACTOR		$\begin{cases} Y=0 & \text{if } x \le T \\ Y=m & \text{if } x>T \end{cases}$	$\begin{cases} x^+=x & \text{if } x \ge 0 \\ x^+=0 & \text{if } x<0 \end{cases}$ $\begin{cases} x^-=0 & \text{if } x \ge 0 \\ x^-=x & \text{if } x<0 \end{cases}$

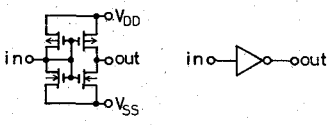


Fig. 2. Sign inverter: (a) circuit and (b) symbol.

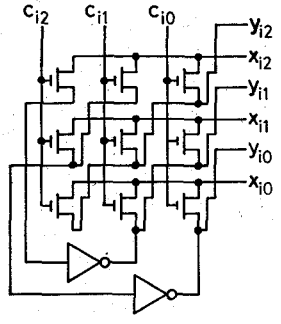
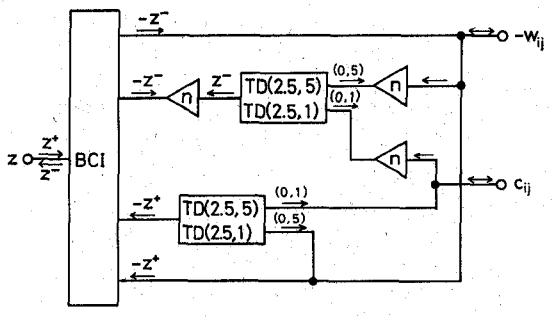


Fig. 3. Mod 7 multiplier.

positive values can be included in a single digit, and the radix-5 SD arithmetic can be implemented with reduced interconnections. Table II summarizes the basic multiple-valued bidirectional current-mode circuits. A detailed discussion of these circuits is found in [10].

B. Mod m_i Multiplier

Since mod m_i multiplication is performed by a shift operation, the mod m_i multiplier can be constructed using a barrel shifter with sign inverter. The circuit shown in Fig. 2 is the sign inverter used to invert the polarity of the input current. As an example, the mod 7 multiplier can be constructed using a 3×3 barrel shifter and sign inverters as shown in Fig. 3, where c_{i0}, c_{i1}, and c_{i2} are the control signals for the multiplier coefficients of 5^0, 5^1, and 5^2, respectively. The products for the multiplier coefficients of -5^0, -5^1, and -5^2 are also obtained by sign inversion of the corresponding outputs. The multiply time is always determined by the propagation delay time of the pass transistor and the sign inverter independently of the word length.



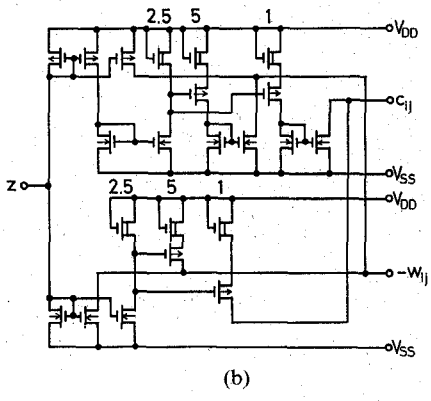


Fig. 4. Radix-5 SDFA: (a) block diagram and (b) circuit configuration.

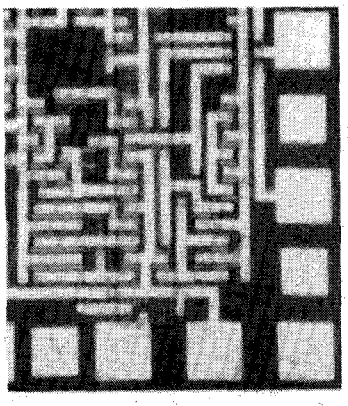


Fig. 5. Photomicrograph of radix-5 SDFA.

C. Mod m_i Adder

The mod m_i adder can be constructed using a radix-5 SDFA. In the bidirectional current-mode MOS technology, the addition steps of (9) and (11) can be performed as a wired sum. The main operation of the SDFA is defined by (10). The SDFA has been constructed using 22 transistors, and a block diagram and circuit configuration of the SDFA are shown in Fig. 4.

The SDFA has been fabricated at Tohoku University only to confirm the basic transfer characteristic using the usual 10-μm CMOS design rule. Fig. 5 shows a photomicrograph of the integrated SDFA. The effective size is 490×445 μm^2. The current transfer characteristics of the

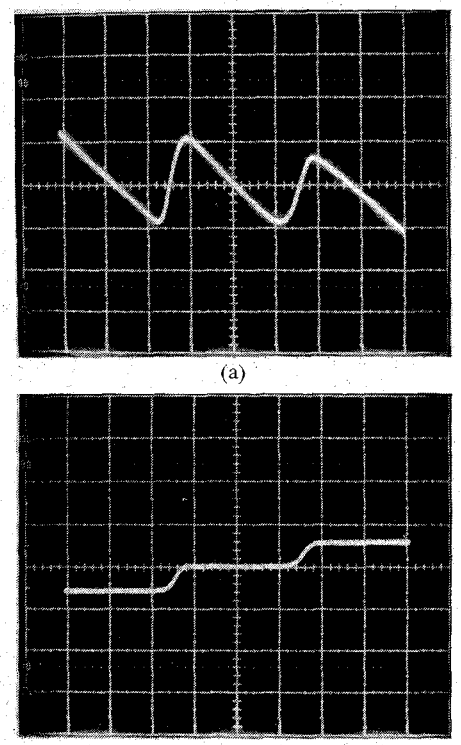


Fig. 6. Current-transfer curve of radix-5 SDFA. (a) Partial sum of SDFA. Horizontal axis: input z_{ij} (97.6 μA/div); vertical axis: partial sum output w_{ij} (97.6 μA/div). (b) Carry of SDFA. Horizontal axis: input z_{ij} (97.6 μA/div); vertical axis: carry output c_{ij} (97.6 μA/div).

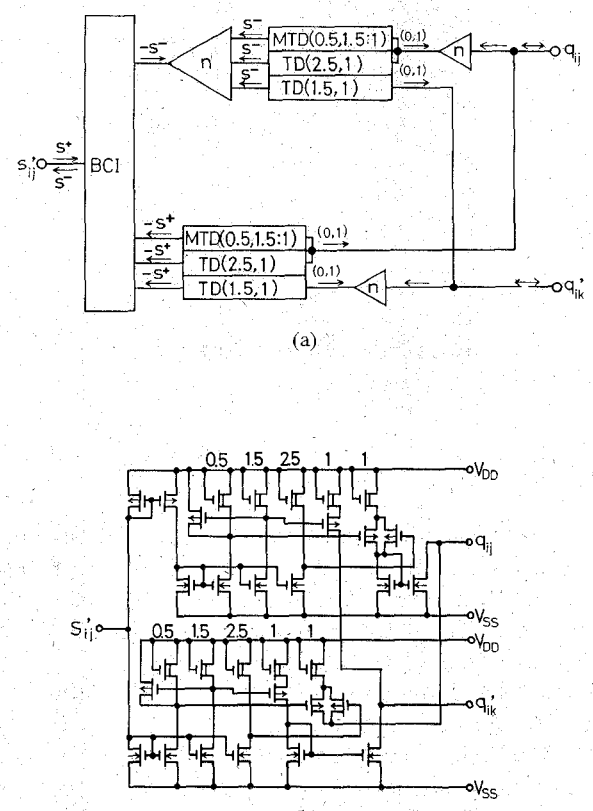


Fig. 7. Decoder: (a) block diagram and (b) circuit configuration.

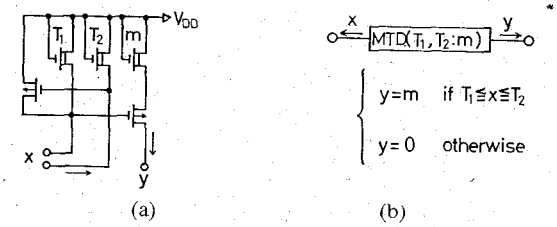


Fig. 8. Modified threshold detector: (a) circuit and (b) symbol.

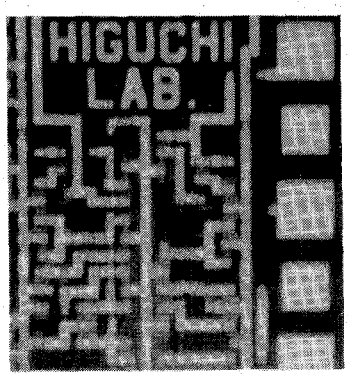


Fig. 9. Photomicrograph of decoder.

partial sum and the carry are shown in Fig. 6, where the unit current is approximately 50 μA. Although a slight deviation is observed around the threshold current, the characteristics following (10) are obtained in principle.

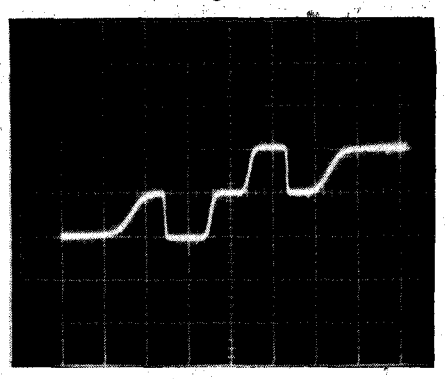
D. Decoder for Three-Operand Addition

The decoder of (17) is also effectively implemented using the bidirectional current-mode circuits. Since the radix-5 SDFA shown in Fig. 4 has no ability to restore signal levels, the current level should be quantized in the decoder. Fig. 7 shows a block diagram and the circuit of the decoder, using 32 transistors. In the decoder, a modified threshold detector (MTD), shown in Fig. 8, is used. The operation is defined as

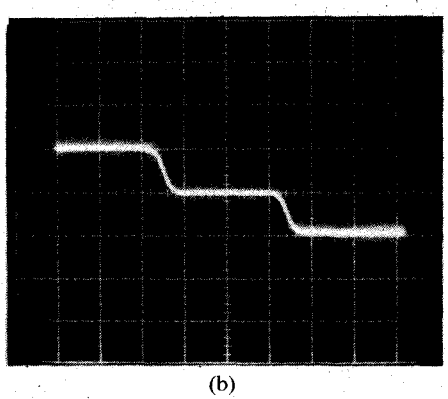
$$\mathrm{MTD}(T_1, T_2: m) = \begin{cases} m, & \text{if } T_1 \le x \le T_2 \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Fig. 9 shows a photomicrograph of the decoder implemented using the same 10-μm CMOS design rule. The effective size is 490×475 μm^2. The characteristics of the implemented circuits are shown in Fig. 10. Fig. 10(a) and (b) shows the current transfer curves for the output q_{ij} and q'_{ik} of the decoder, respectively.

Let us consider the noise margin with respect to a device parameter deviation. If the statistical variation from the desired output of the current source exceeds the noise margin, logical errors will occur. The variation of the current-source output current ΔI is mainly caused by the variation of the transistor threshold voltage ΔV_T. The variation is represented as $\Delta I = (\Delta V_T / V_T)(2 + \Delta V_T / V_T) \cdot I_a$, where I_a is the unit current. In the case where $\Delta V_T = 50$



(a)



(b)

Fig. 10. Current-transfer curve of decoder. (a) Output q_{ij} of decoder. Horizontal axis: input s'_{ij} (48.8 μA/div); vertical axis: output q_{ij} (48.8 μA/div). (b) Output q'_{ik} of decoder. Horizontal axis: input s'_{ij} (48.8 μA/div); vertical axis: output q'_{ik} (48.8 μA/div).

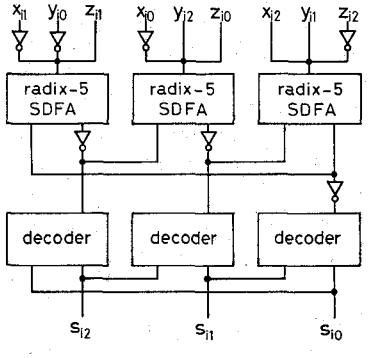


Fig. 11. Mod 7 three-operand multiply adder.

mV, $V_T = 2$ V, and $I_a = 50$ μA, ΔI becomes about 2.5 μA. This variation is not large enough to cause logical errors.

E. Implementation of the Mod 7 Three-Operand Multiply Adder

Fig. 11 shows a block diagram of the mod 7 three-operand multiply adder which is composed of sign inverters, radix-5 SDFA's, and decoders. In this circuit, only multiplication by a constant coefficient is required, such that $s_i = |2x_i + 3y_i - 2z_i|_7$, and it is composed of 190 transistors.

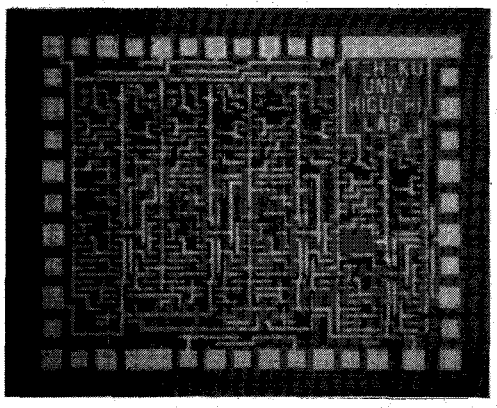


Fig. 12. Chip photomicrograph (2.00×2.50 mm^2).

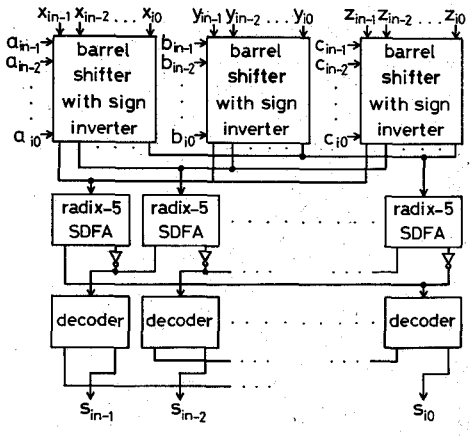


Fig. 13. Mod m_i three-operand multiply adder.

Fig. 12 shows a photomicrograph of the current-mode residue arithmetic chip based on 10-μm CMOS technology. This chip contains a mod 7 three-operand multiply adder, a radix-5 SDFA, and a decoder. The chip size is 2.00×2.50 mm^2 with a total of 244 transistors, and the effective circuit size of the mod 7 three-operand multiply adder is 1.55×1.99 mm^2.

VI. Evaluation

In order to demonstrate an advantage of the proposed residue arithmetic circuit, let us compare typical three-operand multiply adders.

Fig. 13 shows the mod m_i three-operand multiply adder based on the proposed residue arithmetic circuits. It is clear that the structure is very regular even if m_i is large.

The worst-case total delay time t_{total} of the mod m_i three-operand multiply adder is expressed as

$$t_{\text{total}} = t_m + t_s + t_d \tag{20}$$

where t_m, t_s, and t_d are the maximum propagation delay times of the mod m_i multiplier, the SDFA, and the decoder, respectively. By substituting these delay times ob-

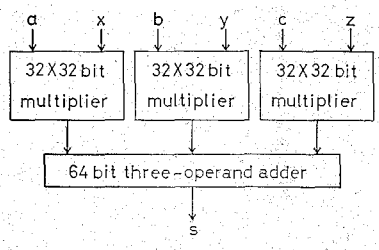


Fig. 14. Binary implementation of three-operand multiply adder.

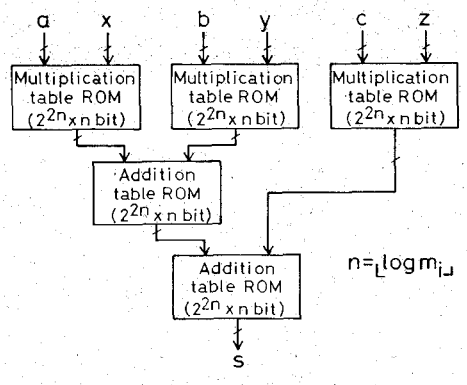


Fig. 15. ROM implementation.

tained from SPICE2 simulation using the 10-μm device parameters to (20), the operating time is estimated to be 270 ns. Using the 2-μm device parameters, the three-operand multiply-add time can be estimated as 10 ns. These results are quite natural from the speed measurement of the similar current-mode logic circuits presented in [10]. The delay time is equivalent to the total multiply-add time because of the parallelism of the multiple-valued residue arithmetic circuit.

Fig. 14 shows the 32×32-bit three-operand multiply adder constructed by the fastest 32×32-bit binary multiplier and the 64-bit three-operand binary adder based on 2-μm CMOS technology. The 64-bit three-operand binary adder is realized by a 64-bit carry-save adder and a 64-bit block carry-lookahead adder. The hardware complexity is increased to achieve high-speed operations.

Fig. 15 shows the ROM implementation of the mod m_i three-operand multiply adder based on residue arithmetic. This implementation requires $2^{2n}\times n$-bit memory capacity to store all possible outcomes of a binary operation, where $n=\lfloor \log m_i \rfloor$, and $\lfloor x \rfloor$ denotes the smallest integer such that $\lfloor x \rfloor \geq x$. The operating speed depends on the access time of ROM.

A comparison of the above various 32×32-bit three-operand multiply adders is shown in Table III. The set $\beta=\{7,11,17,19,23,37,43,47,53,59,73,79,83\}$ equivalent to approximately $65\cdot8$ bit is chosen as the moduli set of the residue arithmetic circuits. In the proposed residue arithmetic circuit, the number of transistors is approximately 34 percent of that for binary arithmetic. Also, the regularity of the layout greatly contributes to the reduction of interconnection area in the multiple-valued residue arithmetic circuit. As a result, the chip area will be greatly reduced compared with the binary arithmetic circuit.

TABLE III
COMPARISON OF THREE-OPERAND MULTIPLY ADDERS (2-μm CMOS TECHNOLOGY)

	RESIDUE ARITHMETIC		BINARY ARITHMETIC
	Multiple-valued	ROM	
Number of transistors	47,000	10^6	140,000
Multiply-add time (ns)	10	50*x3	85

*: access time of ROM

It is clear that a highly compact and high-speed residue arithmetic chip can be realized using multiple-valued logic.

VII. CONCLUSION

In this paper, we have discussed the design of a multiple-valued coded residue arithmetic circuit based on the bidirectional current-mode MOS technology. Because of its high degree of parallelism, both mod m_i multiplication and mod m_i addition can always be performed within a fixed delay time of the module. Furthermore, the hardware complexity is greatly reduced due to the regularity of the structure and the use of multiple-valued logic. Although residue arithmetic operations are restricted on integer arithmetic, the above high performance can hardly be achieved through the use of conventional binary arithmetic circuit. This highly parallel residue arithmetic chip will be of great use in many real-time applications.

ACKNOWLEDGMENT

The authors wish to thank Dr. S. Kawahito of Tohoku University for many helpful comments.

REFERENCES

[1] N. S. Szabo and R. I. Tanaka, *Residue Arithmetic and Its Applications to Computer Technology*. New York: McGraw-Hill, 1967.
[2] G. A. Jullien, "Residue number scaling and other operations using ROM arrays," *IEEE Trans. Comput.*, vol. C-27, pp. 325–336, Apr. 1978.
[3] M. Kameyama, T. Sekibe, and T. Higuchi, "Design of highly parallel residue arithmetic circuits based on multiple-valued bidirectional current-mode MOS technology," in *Proc Int. Symp. Multiple-Valued Logic*, May 1988, pp. 6–13.
[4] K. C. Smith, "Multiple-valued logic: A tutorial and appreciation," *IEEE Computer*, vol. 21, no. 4, pp. 17–27, Apr. 1988.
[5] D. Etiemble and M. Israel, "Comparison of binary and multivalued ICs according to VLSI criteria," *IEEE Computer*, vol. 21, no. 4, pp. 28–42, Apr. 1988.
[6] M. Kameyama and T. Higuchi, "A new scaling algorithm in symmetric residue number system based on multiple-valued logic," in *Proc. IEEE Int. Symp. Circuit Syst.* (Tokyo, Japan), July 1979, pp. 189–192.
[7] L. B. Wheaton and K. W. Current, "A quaternary threshold logic modulo-four multiplier circuit for residue number system nonrecursive digital filters," in *Proc. Int. Symp. Multiple-Valued Logic*, May 1981, pp. 48–53.
[8] M. A. Soderstrand and R. A. Escott, "VLSI implementation in multiple-valued logic of an FIR digital filter using residue number system arithmetic," *IEEE Trans. Circuits Syst.*, vol. CAS-31, pp. 5–25, Jan. 1986.
[9] S. Kawahito, M. Kameyama, and T. Higuchi, "VLSI-oriented bidirectional 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.
[10] S. Kawahito, M. Kameyama, T. Higuchi, and H. Yamada, "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.
[11] K. Hwang, *Computer Arithmetic*. New York: Wiley, 1979.

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.

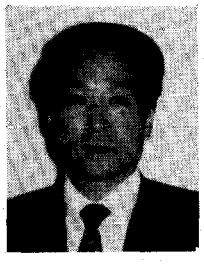
He is currently an Associate Professor in the Department of Electronic Engineering, Tohoku University. His general research interests include multiple-valued logic systems, VLSI systems, highly reliable digital systems, and robot electronics.

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 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.

Tsutomu Sekibe was born in Fukuoka, Japan, on October 15, 1964. He received the B.E. and M.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1987 and 1989, respectively.

He joined Matsushita Electric Industrial Company, Ltd., Osaka, Japan, in 1989. His main interests and activities include ultrahigh-speed arithmetic VLSI system.

Mr. Sekibe is a member of the Institute of Electronics, Information and Communication Engineers of Japan.



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-dimensional and two-dimensional finite word-length digital filters, multiple-valued logic 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 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 he was the Chairman of the Japan Research Group on Multiple-Valued Logic.