

# Design and Application of Pipelined Dynamic CMOS Ternary Logic and Simple Ternary Differential Logic

Chung-Yu Wu, *Member, IEEE*, and Hong-Yi Huang, *Student Member, IEEE*

**Abstract**—New dynamic CMOS ternary logic circuits, which can be used to form a pipelined system with the nonoverlapped two-phase clocks, are proposed and investigated. All the proposed new dynamic ternary gates do not have dc power dissipations and have full voltage swings. For complex ternary logic, a new circuit structure called the simple ternary differential logic (STDL) is also proposed and analyzed. The design procedure of the STDL is developed for the optimal implementation. An experimental chip has been fabricated in 1.2- $\mu\text{m}$  CMOS process and tested, which successfully verifies part of the logic functions of the proposed new dynamic ternary logic. A new binary pipelined multiplier is designed by using the proposed dynamic ternary logic circuits in the interior of the multiplier in the coding of radix-2 redundant positive-digit number. The new structure has the advantages of higher operating frequency as well as much less latency and total device count as compared with the conventional binary parallel pipelined multiplier. It has been shown that all the developed dynamic ternary logic circuits have certain advantages in speed, power dissipation, chip area, and clock complexity over other dynamic ternary logic circuits. Moreover, the pipelined structure is free from race problems.

## I. INTRODUCTION

IT IS known that as the chip integration increases toward VLSI/ULSI, the interconnections, both on chip and among chips, become a severe problem. The on-chip complex wire routing produces a heavy load that decreases the chip speed, whereas the increasing off-chip connections degrade the system speed performance. Being able to reduce the number of interconnection lines or nets and increase their information content, multivalued logic (MVL) becomes quite attractive in VLSI/ULSI applications. Its advantages have been confirmed in various applications, such as memories, communications, arithmetic circuits, signal processing, and supporting chips [1].

Among various types of MVL, the ternary logic receives more attention than others because of a lower interconnection cost estimation [2] and a simple electronic circuit implementation method [3]. Several static ternary logic circuits have been proposed so far. But they dissipate a lot of power [4]–[7], and require a complex process to obtain both depletion and enhancement devices [8], [9] or multithreshold voltages [10]. A dynamic four-value circuit has also been proposed [11], but only basic cells have been described. Recently, a low-power dynamic ternary logic [12] has been proposed using the Yoeli–Rosenfeld algebra [13]. However, the proposed dynamic

ternary logic still has many disadvantages. It requires four-phase clocks, which are too complicated in operation and require a lot of layout area. Moreover, the ternary inverters used in [13] are of the ratioed type. They still have dc power dissipation and do not have a full voltage swing. A new dynamic ternary logic was proposed to decrease the clock phases and the dc power dissipation [14]. However, the power supplies are too low, which may be easily influenced by noise and impulse spike.

In this paper, improved dynamic ternary logic circuits are developed and characterized. The maximum power supply is  $V_{DD}$  (5 V) and only one extra mask is added to the standard CMOS process. As compared to the previous versions [12], [14], the new circuits can improve the disadvantages described above. In the new design, the ternary inverters have no dc power dissipation and have full voltage swings. Besides, only nonoverlapped two-phase clocks are needed for the new dynamic ternary logic circuits and the new circuits can also be organized into a pipelined system. By using the NMOS differential tree, a simple ternary differential logic (STDL) is also proposed to realize complex logic functions. It is confirmed that the new dynamic ternary logic has a better performance in speed, power dissipation, and chip area than the previous dynamic ternary logic [12]. As an application example for the proposed ternary logic, a new binary pipelined multiplier is designed by using the new dynamic ternary logic circuits in the radix-2 positive-digit coding  $\{0, 1, 2\}$ . As compared with the conventional binary pipelined multiplier, the new structure has the advantages of much less latency and total device count. Through the SPICE simulations with the same power supplies and device model parameters, the new structure also shows an improvement in the operating frequency.

The basic ternary gate circuits and their operating principles are presented in Section II. The STDL and its design procedures are described in Section III. Experimental results to verify the logic function on an experimental chip are shown in Section IV. The new binary pipelined multiplier designed by using the new dynamic ternary logic circuits is presented in Section V. Finally, the conclusion is given.

## II. BASIC CIRCUIT STRUCTURES AND OPERATING PRINCIPLES

The Yoeli–Rosenfeld algebra [13], which is suitable for arithmetic operations [15], can be easily realized through electronic implementations to form the dynamic ternary logic family. Three basic inverter functions, namely simple ternary inverter (STI), negative ternary inverter (NTI), and positive

Manuscript received September 9, 1992; revised March 9, 1993.

The authors are with the Integrated Circuits and Systems Laboratory and the Department of Electronics Engineering and Institute of Electronics, National Chiao Tung University, Hsin-Chu, Taiwan 300, Republic of China.

IEEE Log Number 9209618.

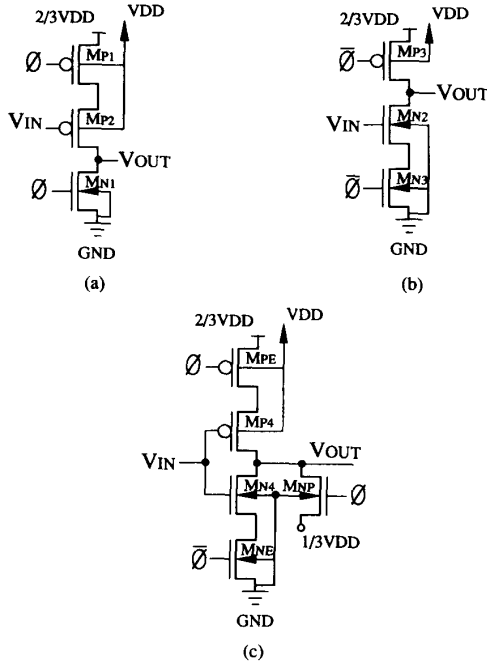


Fig. 1. Circuit structures of (a) dynamic NTI gate, (b) dynamic PTI gate, and (c) dynamic STI gate.

TABLE I  
POWER SUPPLIES AND IMPORTANT DEVICE  
PARAMETERS OF THE DYNAMIC TERNARY LOGIC

Parameters	V <sub>DD</sub>	2/3V <sub>DD</sub>	1/3V <sub>DD</sub>	GND	V <sub>TP0</sub>	V <sub>TN0</sub>	V <sub>CLK</sub>	V <sub>IN</sub>	V <sub>OUT</sub>
Voltage level or swing	5	3.3	1.65	0	-1.5	1.9	0-5	0-3.3	0-3.3

ternary inverter (PTI), form an operator set that is complete in the logic sense. All three inverters can be combined to realize ternary functions such as ternary NAND (TNAND) and ternary NOR (TNOR), with the minimum (AND) and maximum (OR) functions, respectively [13].

All three ternary inverters are designed in a dynamic form. The resultant new circuit structures in CMOS are shown in Fig. 1(a)-(c). It is seen that only enhancement PMOS and enhancement NMOS devices are used. The values of power supplies and important device parameters relevant to the explanation of the circuit operation are listed in Table I. The high, intermediate, and low logic levels are  $2/3V_{DD}$  (3.3 V),  $1/3V_{DD}$  (1.65 V), and GND (0 V), which represent the radix-2 positive-digit codes of 2, 1, and 0, respectively. The logic levels  $2/3V_{DD}$  and  $1/3V_{DD}$  can be generated on chip from the power supply  $V_{DD}$  (5 V).

By using a 1.2- $\mu\text{m}$  n-well CMOS process, the natural threshold voltages (the threshold voltages without the threshold adjustment implantation) for PMOS and NMOS devices are  $V_{TP} = -1.5$  V and  $V_{TN} = 0.2$  V, respectively. Generally, the standard CMOS process requires only one boron implantation to get the required threshold voltages  $V_{TP} = -0.8$  V and  $V_{TN} = 0.8$  V. In the new ternary logic circuits, the threshold implantation is only applied to the NMOS devices to increase

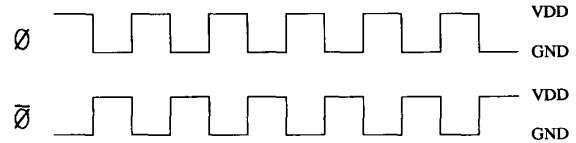


Fig. 2. Two-phase nonoverlapped clocks.

TABLE II  
TRUTH TABLE OF THE DYNAMIC NTI, PTI, AND STI

X	NTI	PTI	STI
0	2	2	2
1	0	2	1
2	0	0	0

the NMOS threshold voltage from 0.2 to 1.9 V, while keeping the PMOS at the natural threshold voltage  $-1.5$  V. The process requires only a different boron dose for channel implantation and an additional mask to prevent PMOS from being implanted. The threshold voltage  $V_{TN0}$  of the NMOS device under zero body-to-source bias is designed to be 1.9 V, and that of the PMOS device under the same bias,  $V_{TP0}$ , is  $-1.5$  V. In the clock generation circuit,  $V_{DD} = 5$  V is used. The substrate of PMOS devices in Fig. 1(a)-(c) is connected to  $V_{DD}$ . Thus the threshold voltage of the PMOS device with such a 1.7-V reverse source-to-substrate bias ( $V_{SB} = -1.7$  V) is increased to be about  $-1.9$  V, which is the same in amplitude as  $V_{TN0}$ .

The clocks used in the dynamic ternary logic are nonoverlapped two-phase clocks as shown in Fig. 2. The voltage swing  $V_{CLK}$  of the clocks  $\phi$  and  $\bar{\phi}$  is 5 V. In the dynamic NTI shown in Fig. 1(a), if  $\phi = 5$  V,  $M_{P1}$  is turned off and  $M_{N1}$  is turned on. Then the output is preset to 0 V in spite of the input. For  $\phi = 0$  V,  $M_{N1}$  is turned off and  $M_{P1}$  is turned on. Since the threshold voltage of  $M_{P2}$  is about  $-1.9$  V, it is turned off when the input voltage is 3.3 or 1.65 V. Then the output remains at the preset voltage 0 V. When the input is 0 V,  $M_{P2}$  is turned on and the output is pulled up to 3.3 V. Thus, the operation of this circuit is consistent with the truth table of the NTI as shown in Table II.

In the dynamic PTI as shown in Fig. 1(b), if  $\bar{\phi} = 0$  V,  $M_{P3}$  is turned on and  $M_{N3}$  is turned off. The output is preset to 3.3 V in spite of the input. For  $\bar{\phi} = 5$  V,  $M_{N3}$  is turned on and  $M_{P3}$  is turned off. When the input is 1.65 or 0 V,  $M_{N2}$  is turned off and the output remains at the preset voltage 3.3 V. When the input is 3.3 V,  $M_{N2}$  is turned on and the output is pulled down to 0 V. This verifies the truth table of the PTI as shown in Table II.

In the dynamic STI as shown in Fig. 1(c) [12], when  $\phi = 5$  V and  $\bar{\phi} = 0$  V,  $M_{PE}$  and  $M_{NE}$  are turned off and  $M_{NP}$  is turned on. The output is preset to 1.65 V. For  $\phi = 0$  V and  $\bar{\phi} = 5$  V,  $M_{PE}$  and  $M_{NE}$  are turned on and  $M_{NP}$  is turned off. The path to  $1/3V_{DD}$  is turned off and thus the output depends on the input state. As the input is 0 V,  $M_{P4}$  is turned on and  $M_{N4}$  is turned off. The output is pulled up to 3.3 V.

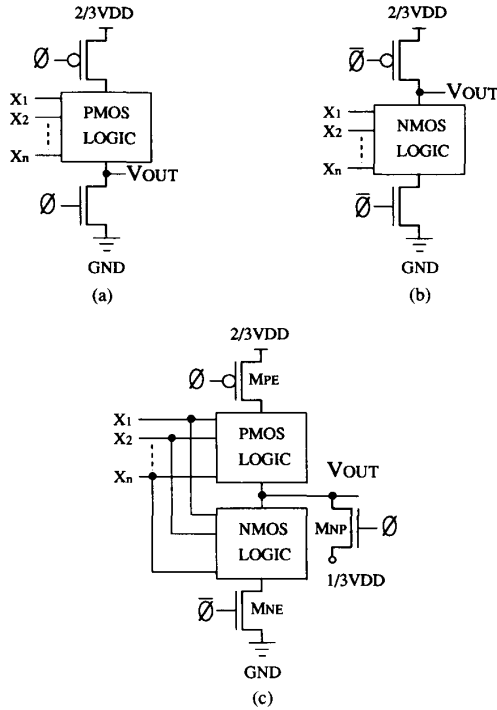


Fig. 3. Circuit structures of (a) basic dynamic negative ternary gate, (b) basic dynamic positive ternary gate, and (c) basic dynamic simple ternary gate.

As the input is 1.65 V, both  $M_{P4}$  and  $M_{N4}$  are turned off. Neither the path to 3.3 V nor the path to 0 V could conduct. The output remains at the preset voltage 1.65 V. As the input is 3.3 V,  $M_{P4}$  is turned off and  $M_{N4}$  is turned on. The output is pulled down to 0 V. This verifies the truth table of the STI as shown in Table II. The source of  $M_{NP}$  is at  $1/3V_{DD}$  level and its substrate is at the GND level, so that  $V_{BS} = -1.65$  V. The high voltage level of the clocks should be higher than  $V_{TN}(V_{BS} = -1.65 \text{ V}) + 1/3V_{DD}$ , which is about 3.6 V, to turn on the NMOS  $M_{NP}$ . In this system, therefore, 5 V is chosen as the high voltage of the clocks.

From the above description, it is realized that the dc power dissipation of these ternary inverters is very small, just as that of the conventional CMOS inverters. Moreover, all the ternary inverters have a rail-to-rail voltage swing independent of the MOS dimensions.

The positive ternary gate (PTG) and the negative ternary gate (NTG) have the same structure as the binary dynamic CMOS logic [16]. For the NTG shown in Fig. 3(a), there are a clocked NMOS device used to preset the output to 0 V and a PMOS logic circuit for logic implementation. For the PTG shown in Fig. 3(b), there are a clocked PMOS device used to preset the output to 3.3 V and an NMOS logic circuit for logic implementation. The simple ternary gate (STG) shown in Fig. 3(c) [12] is a combination of the conventional static binary CMOS gate, the presetting NMOS  $M_{NP}$ , the evaluating PMOS  $M_{PE}$ , and the evaluating NMOS  $M_{NE}$ .

As a demonstrating example, the two-input positive ternary NAND (PTNAND), negative ternary NAND (NTNAND), simple

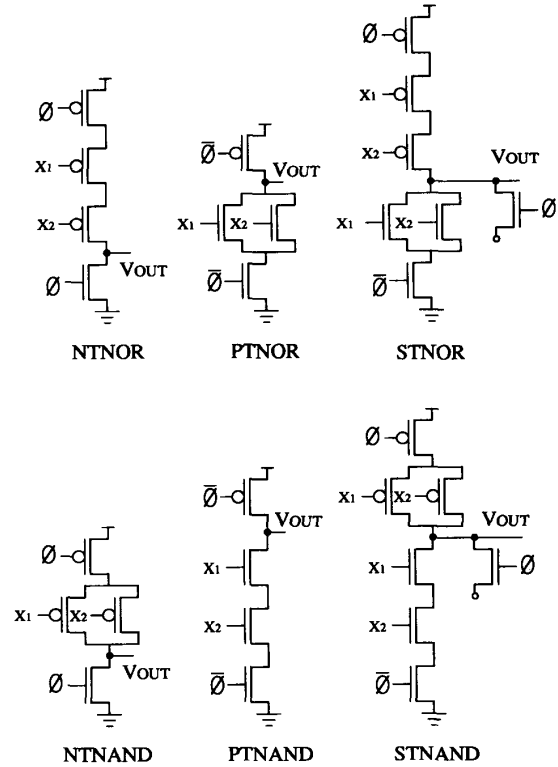


Fig. 4. Schematic diagrams of the two-input ternary NAND and NOR gates.

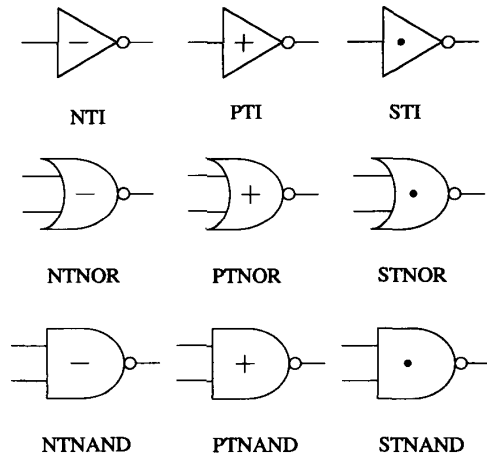


Fig. 5. Logic symbols of the ternary NAND and NOR gates.

ternary NAND (STNAND), positive ternary NOR (PTNOR), negative ternary NOR (NTNOR), and simple ternary NOR (STNOR) are shown in Fig. 4 [8]. Their logic symbols are shown in Fig. 5 whereas their truth tables are listed in Table III. The logic function verifications can be done similarly.

The PTG's, STG's, and NTG's are preset to 3.3, 1.65, and 0 V, respectively. Thus, direct connection of the same types of gates is not permitted because of the race problem. To avoid the race problem, the design rules should be set to form a

TABLE III  
TRUTH TABLE OF THE TWO-INPUT TERNARY NAND AND NOR

X	Y	STNAND	PTNAND	NTNAND	STNOR	PTNOR	NTNOR
0	0	2	2	2	2	2	2
0	1	2	2	2	1	2	0
0	2	2	2	2	0	0	0
1	0	2	2	2	1	2	0
1	1	1	2	0	1	2	0
1	2	1	2	0	0	0	0
2	0	2	2	2	0	0	0
2	1	1	2	0	0	0	0
2	2	0	0	0	0	0	0

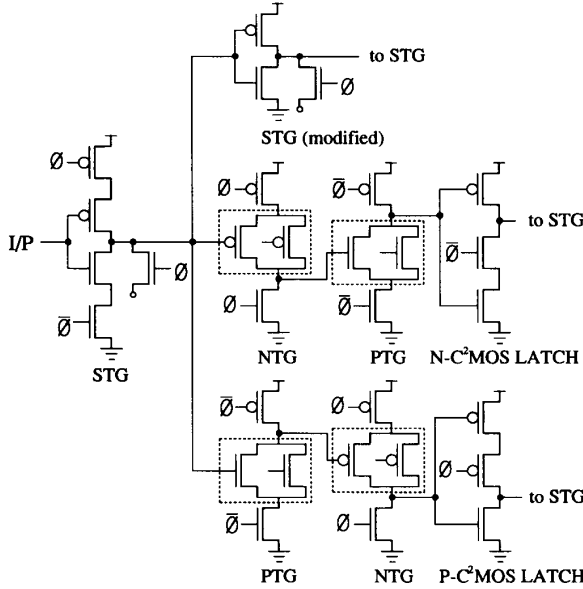


Fig. 6. The  $\phi$ -section circuit of the ternary NORA pipelined system.

race-free structure called the ternary NORA (TERN-NORA) system as shown in Fig. 6. In this system, both NTG's and PTG's can be connected after the STG's. When the STG's are preset to  $1/3V_{DD}$ , it cannot turn on either the PMOS logic gate in the NTG or the NMOS logic gate in the PTG. Thus, the internal race will not occur in this connection. Because the operating principles and the structures of the ternary dynamic logic NTG's (PTG's) are similar to those of the p-type (n-type) binary CMOS dynamic gates [17], the NTG's and the PTG's can be connected after each other. But the direct connection among PTG's is not permitted, and neither is the direct connection among NTG's. The mutual connection among STG's is permitted. In this case the two evaluating devices in the load STG can be omitted as shown in the upper part of Fig. 6.

C<sup>2</sup>MOS latch stages between PTG's and STG's and between NTG's and STG's must be added to guarantee a fully race-free operation in a pipelined system. Based upon the same considerations as in [18]–[20], the C<sup>2</sup>MOS latch stage connected after a PTG can be replaced by a N-C<sup>2</sup>MOS latch stage as shown in Fig. 6, because the clocked PMOS in

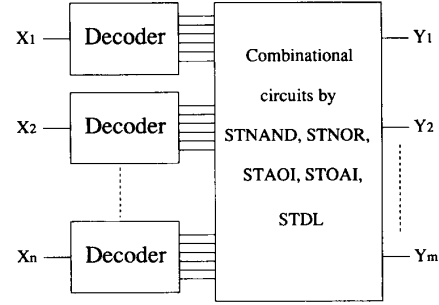


Fig. 7. Basic building block of the dynamic ternary logic circuit.

TABLE IV  
TRUTH TABLE OF THE TERNARY LITERALS

X	X <sup>0</sup>	X <sup>1</sup>	X <sup>2</sup>	X <sup>01</sup>	X <sup>12</sup>	X <sup>02</sup>	$\bar{X}$
0	2	0	0	2	0	2	2
1	0	2	0	2	2	0	1
2	0	0	2	0	2	2	0

the C<sup>2</sup>MOS latch is redundant. Similarly, the C<sup>2</sup>MOS latch connected after a NTG can be replaced by a P-C<sup>2</sup>MOS latch. If the dynamic ternary gates are connected according to the above rules, no glitch or race problem could occur.

Every ternary function can be implemented by the building block shown in Fig. 7. Every input  $X_i$  requires a decoder to convert the three-state input to the two-state literals [13]. Then the ternary function can be implemented by STNAND, STNOR, STAOI, STOI [12], and STDL. The STDL will be described later in the next section. Both the map method and the Quine method [13], [15] can be applied to minimize ternary functions.

The truth table of the ternary literals is listed in Table IV. Not all these literals are required in implementing the ternary function, because there exist certain relationships among them:

$$X^{01} = X^0 + X^1 \quad (1)$$

$$X^{12} = X^1 + X^2 \quad (2)$$

$$X^{02} = X^0 + X^2 \quad (3)$$

$$X^{01} \cdot X^{12} = X^1 \quad (4)$$

where "+" means "max" (OR) and "." means "min" (AND).

The ternary decoder used in Fig. 7 has a circuit structure as shown in Fig. 8. The decoder is composed of some unary operators, PTI's, NTT's, PTNOR, NTNAND, P-C<sup>2</sup>MOS latch stages, and N-C<sup>2</sup>MOS latch stages and can convert the three-state input  $X$  to the two-state output literals. These literals are input to the STG's. The STG's perform the desired logic operations and send out the three-state outputs. It is important to choose the most suitable literals to implement the ternary logic circuits, so that the amount of logic gates used in the decoders and the stacked devices used in the STG's are minimum.

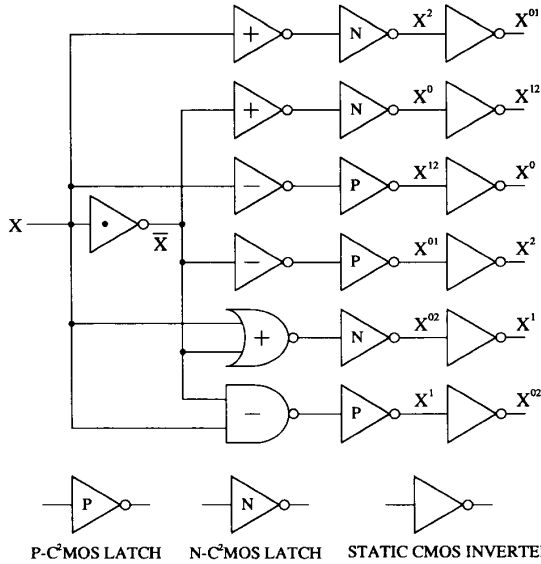


Fig. 8. Block diagram of the dynamic ternary decoder composed of unary operators.

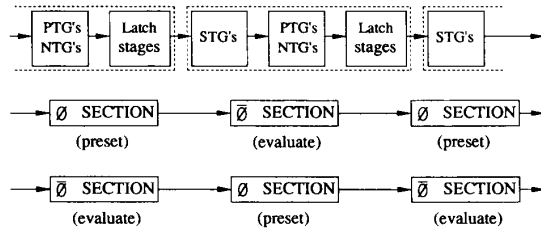


Fig. 9. Block diagram of the dynamic ternary pipelined system.

A ternary pipelined system thus can be built up by cascading the building blocks shown in Fig. 7. A string of PTG's (NTG's), latch stages, and STG's is shown in Fig. 9, which forms the ternary pipelined system. The inputs of the ternary system should be applied to PTG's and NTG's. Define the first PTG's (NTG's) and the latch stages as the first  $\phi$  section ( $\bar{\phi}$  section). Then the following STG's, PTG's (NTG's), and latch stages are defined as the normal  $\bar{\phi}$  section ( $\phi$  section). Then, a normal  $\phi$  section ( $\bar{\phi}$  section) composed of a string of STG's, PTG's (NTG's), and latch stages is cascaded after the normal  $\bar{\phi}$  section ( $\phi$  section), and so on. The clock used for  $\bar{\phi}$  section is the complement of that used in the  $\phi$  section.

When the  $\phi$  section is in the preset phase, the outputs of all the PTG's, STG's, and NTG's in the  $\phi$  section are preset to 3.3, 1.65, and 0 V, respectively. At this time, the inputs to the  $\phi$  section have been set up and kept unchanged in this phase. As the  $\phi$  section is in the evaluation phase, the outputs of all the dynamic blocks are evaluated as a function of the  $\phi$ -section inputs. Among these output results, those which must be transferred to the next pipelined section are stored in P-C<sup>2</sup>MOS latch and N-C<sup>2</sup>MOS latch stages [17].

The operating principle of the pipelined ternary system can be further explained by an example circuit which is formed by

cascading two stages of cycling gates, as shown in Fig. 10. The truth table of a cycling gate is listed in Table V. A sequence of  $\phi$ ,  $\bar{\phi}$ , and  $\phi$  sections forms the pipelined two-stage cascaded cycling gate. Note that the two STG circuits, which consist of the devices  $P_{10}$ ,  $P_{11}$ ,  $N_{7-10}$ ,  $P_{21}$ ,  $P_{22}$  and  $N_{17-20}$ , are slightly modified such that the least number of devices is required.

### III. STDL AND DESIGN PROCEDURES

Conceptually an STG is a combination of a conventional static binary CMOS gate with presetting and evaluating devices. Based upon this concept, a new dynamic ternary differential logic (STDL) is proposed as shown in Fig. 11(a). The structure of STDL is similar to the binary enabled/disabled CMOS differential logic (ECDL) [21] and the latched CMOS differential logic (LCDL) [22]. The STDL has the same advantages as the binary differential logic, such as a shorter circuit delay, less layout area, less power dissipation, and an increase of logic flexibility. Besides, two differential outputs also induce less current spike in the power supplies and less coupling noise because of the  $1/3V_{DD}$  voltage excursion.

As may be seen from Fig. 11(a), there are three major components in the STDL. First, there is a sensing latch consisting of the transistors  $P_2$ ,  $P_3$ ,  $N_2$ , and  $N_3$  and the clocked transistors  $P_1$  and  $N_1$ . It is the same as the sense amplifier (SA) used in a CMOS dynamic random access memory (DRAM) [23]. Second, there is an NMOS differential network ternary logic tree with an evaluating transistor  $N_6$  to realize the ternary logic functions. Finally, there are two presetting transistors  $N_4$  and  $N_5$  for the two output nodes to be preset to  $1/3V_{DD}$ . When  $\phi = 5$  V and  $\bar{\phi} = 0$  V, the clocked transistors  $P_1$ ,  $N_1$ , and  $N_6$  are turned off, so that the SA is disabled, having no  $2/3V_{DD}$  or GND connections. Meanwhile, both outputs  $Q$  and  $\bar{Q}$  are preset to  $1/3V_{DD}$  by  $N_4$  and  $N_5$ . As  $\phi = 0$  V and  $\bar{\phi} = 5$  V, the evaluation devices  $P_1$ ,  $N_1$ , and  $N_6$  are turned on, which enables the SA. Depending on the inputs of the ternary differential logic tree, a path may exist from one of the output nodes to GND, while the other node is pulled up to  $2/3V_{DD}$ . The other possible case is that there is no current path from both output nodes to GND. Thus both output nodes are held at  $1/3V_{DD}$ . There exists a dead band from 1.4 to 1.9 V, within which the SA is still disabled because the devices of the SA are not turned on. It is a good protection for the charge-sharing or the noise problems.

An improved STDL circuit is shown in Fig. 11(b) where the transmission gates instead of the presetting NMOS's are used to preset the outputs nodes. This can reduce the clock feedthrough effect. The other improvement in the improved STDL is the use of multi-preset devices for internal precharging [24]. Because the internal nodes of NMOS differential network tree may not be able to preset to 1.65 V completely, the charge-sharing problem could occur in the evaluation phase among the directly cascading STDL's or STG's. The internal precharging can reduce the problem. Dimension optimization for the devices  $P_1-P_3$  and  $N_1-N_3$  shown in Fig. 11(a) is required to obtain the best performance [21], [22].

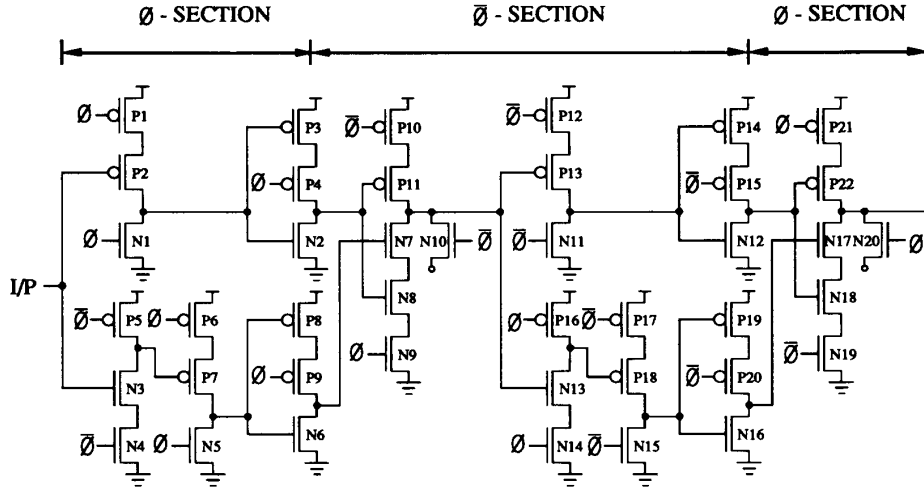


Fig. 10. Circuit configuration of a two-stage cascaded dynamic ternary cycling gate.

TABLE V  
TRUTH TABLE OF A TERNARY CYCLING GATE

IN	OUT
0	2
1	0
2	1

In the realization of ternary functions, the state "1" means that the gates of the MOS transistors are connected to  $1/3V_{DD}$ . Thus, these devices are absolutely off. Such devices and those that are in series with them are redundant, because they are never turned on to form a conducting path to  $2/3V_{DD}$  or GND. Thus, these devices can be neglected and only the states "2" and "0" are considered in the Karnaugh map.

For  $n$  input variables  $X_1, X_2, \dots, X_n$  of an STD L tree, every input variable  $X_i$  can be decoded to the literal  $X_i^a$ , where  $a \in \{01, 02, 12, 0, 1, 2\}$ . There are negation relationships among these literals:

$$X_i^{01} = \overline{X_i^2} \quad (5)$$

$$X_i^{02} = \overline{X_i^1} \quad (6)$$

$$X_i^{12} = \overline{X_i^0} \quad (7)$$

where  $\overline{X_i^a}$  is called the negation of  $X_i^a$ .

A cube is a set  $P$  of literals  $X_i^a$  such that  $X_i^a \in P$  implies  $X_i^b \notin P$ , in which  $X_i^a \neq X_i^b$  and  $a, b \in \{01, 12, 02, 0, 1, 2\}$ . For example, consider a ternary function  $F = X_1^2 X_2^{01} X_3^1 + X_1^1 X_2^2$ . There are two cubes  $X_1^2 X_2^{01} X_3^1$  and  $X_1^1 X_2^2$ . The cube  $X_1^2 X_2^{01} X_3^1$  is a set of literals  $X_1^2$ ,  $X_2^{01}$ , and  $X_3^1$  whereas the cube  $X_1^1 X_2^2$  is a set of literals  $X_1^1$  and  $X_2^2$ .

In a Karnaugh map of  $n$  variables, there are  $3^n$  cells, each of which represents a cube consisting of exactly  $n$  literals.

Cells that contain the state "2" ("0") are called the 2-cells (0-cells) as shown in Fig. 12(a). A 2-loop (0-loop) is defined by encircling three adjacent 2-cells (0-cells). It represents a cube with one less literal than each of the cubes representing the original 2-cell (0-cell) as shown in Fig. 12(b). For example, the 2-loop is composed of three 2-cells, each of which represents the cubes  $A^2 B^0$ ,  $A^2 B^1$ , and  $A^2 B^2$ . Two literals are required in each cube. The 2-loop is also a cube  $A^2$ , which requires only one literal. Thus the 2-loop is a cube with one less literal than the cube representing each of the three 2-cells in the 2-loop. Adjacent 0-cell and 2-cell form a 02-cell and adjacent 0-loop and 2-loop form a 02-loop, as shown in Fig. 12(c).

The K-map design procedure for the STD L consists of seven steps [25]:

1. Identify three different types of cells in the K-map, namely, 0-, 2-, and 02-cells.
2. Find a minimal cover for the 02-cells.
3. Find a minimal cover for the remaining 0-cells and 2-cells.
4. Consider the required literals from the above chosen cells to implement the ternary functions. Go to (1) and reidentify the types of the cells, so that the required number of literals is minimum.
5. Construct the tree corresponding to the minimal cover of 02-cells chosen above. The variables  $X_i$ 's in each of the tree branches are arranged from top to bottom in ascending order with the magnitude of  $i$ . Always construct tree branches corresponding to loops with a smaller size first. The top pair of control inputs is  $X_i^a$  associated with  $Q$  and  $X_i^b$  associated with  $\overline{Q}$ , where  $X_i^a$  corresponds to "0" and  $X_i^b$  corresponds to "2." The sources of the transistors with their gates driven by  $X_i^a$  and  $X_i^b$  are connected together.
6. Construct the tree corresponding to the cover of 0-cells chosen above. Always look for the sharing of tree branches. The root of the 0-tree is connected to the node  $Q$ .

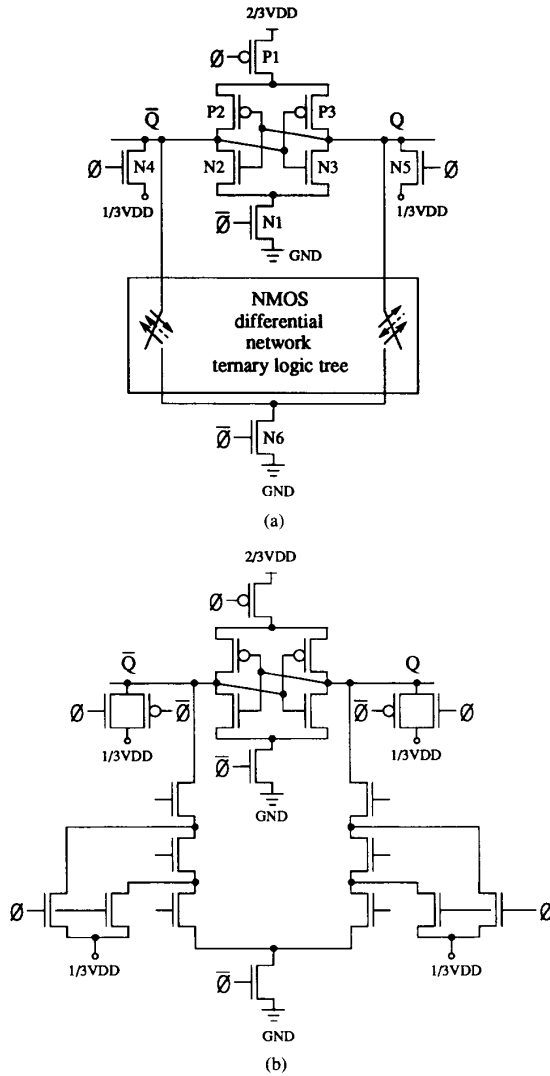


Fig. 11. General circuit configuration of (a) the STDL and (b) the modified STDL.

7. Construct the tree corresponding to the cover of 2-cells chosen above. Always look for the sharing of tree branches. The root of the 2-tree is connected to the node  $\bar{Q}$ .

This procedure may create different tree structures if  $X_i$ 's are permuted (e.g.,  $X_1$  and  $X_2$  variables are interchanged). Also there may be several ways to choose a minimal cover and to share tree branches. Generally, the reduction of the number of devices by tree sharing does not necessarily cause an increase of stacked levels. In fact, the heuristic procedures tend to optimize both devices count and number of stacked levels.

An example is given to demonstrate the above ideas. The K-map shown in Fig. 13(a) has two types of encirclement, namely, 0- and 2-loops, and two types of cell, namely, 0-cell

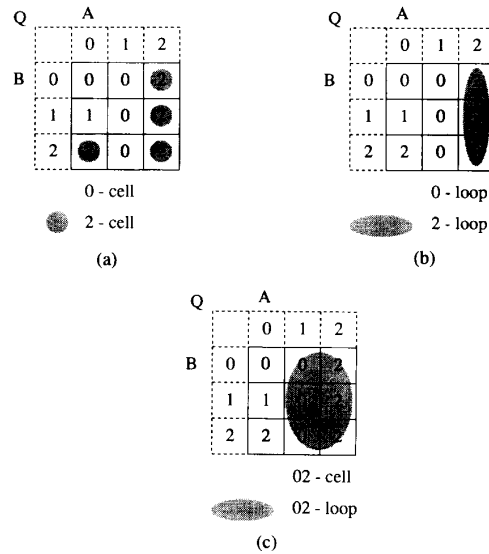


Fig. 12. The K-maps with (a) 0-cell and 2-cell, (b) 0-loop and 2-loop, and (c) 02-cell and 02-loop.

and 02-cell. The shared tree corresponding to the 02-cell is first constructed as shown in Fig. 13(b). Then more branches corresponding to the 2-cell, 2-loop, and 0-loop are added to form a complete STDL tree, as shown in Fig. 13(c).

#### IV. EXPERIMENTAL VERIFICATIONS

Experimental circuits were designed and fabricated to verify part of the functions of the proposed ternary logic circuits. The experimental circuits were fabricated in a  $1.2\text{-}\mu\text{m}$  single-metal single-poly n-well CMOS process with special threshold implantation. The measured threshold voltages are  $V_{TP0} = -1.5\text{ V}$  and  $V_{TN0} = 2.35\text{ V}$ . The test circuits of the STDL is a three-input STNAND gate as shown in Fig. 11(b). It is designed with a single NMOS as the presetting device but without the multi-preset devices. Fig. 14(a) shows the chip photomicrograph of the test circuits. The measured waveforms are shown in Fig. 14(b) where the upper (lower) waveform is the waveform at the output node  $Q$  ( $\bar{Q}$ ). The three different voltage levels can be seen clearly. The operating frequency of the test chip without an on-chip output buffer can be as high as 30 MHz. If a suitable output buffer to drive the output pad is added, the operating frequency could reach the simulated maximum frequency.

The other test circuit is a three-input STNAND gate with the threshold voltage of the presetting NMOS devices being the natural threshold voltage  $V_{TN} = 0.2\text{ V}$ . Since the sources of the presetting NMOS are connected to  $1/3V_{DD}$  and the bulk connected to GND, the threshold voltage of the presetting NMOS devices with 1.65-V reverse source-to-substrate bias ( $V_{SB} = -1.65\text{ V}$ ) is increased to be about 0.5 V. The amplitude of the clocks used in such a case can be from 0 to 3.3 V. Since the sources of the presetting NMOS in the STG and STDL are connected to 1.65 V, a 3.3-V clock leads to 1.65-V gate-to-source voltage to turn on the presetting NMOS.

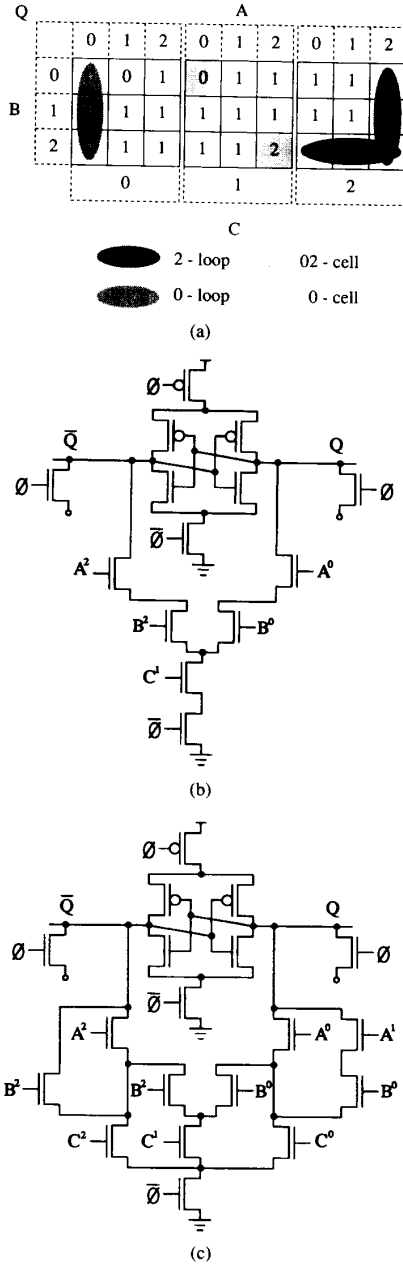


Fig. 13. (a) K-map with 0-, 2-, and 02-loops encirclement. (b) The STDL circuits corresponding to the 02-loop. (c) The complete STDL circuit.

The 5-V power supply is only used for the n-substrate bias to increase the PMOS threshold voltage. This test circuit also works well at 30 MHz. Thus, the logic function is verified through the experimental chip.

#### V. APPLICATIONS

The proposed new ternary logic can be applied to the binary pipelined multiplier. Fig. 15 shows the block diagram of the pipelined multiplier with binary input and output, which is designed by using the dynamic ternary logic circuits in the

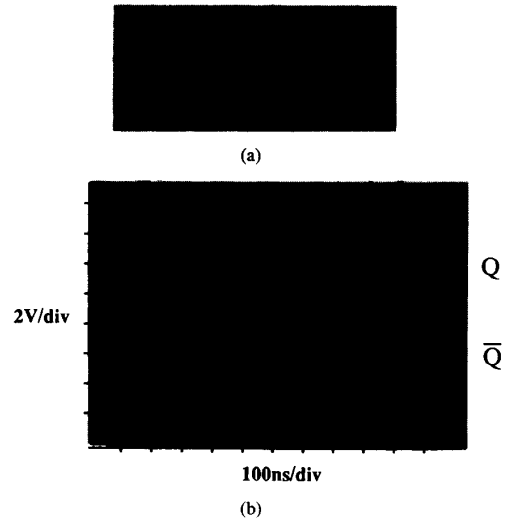


Fig. 14. (a) The chip photomicrograph. (b) The measured waveforms of the fabricated three-input STDL STNAND gate. (Vertical scale: 2 V/div; horizontal scale: 100 ns/div.)

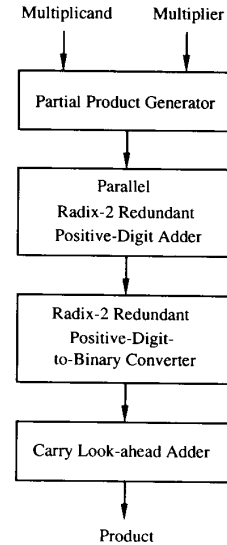


Fig. 15. Block diagram of the new parallel pipelined multiplier.

internal part. The multiplier is composed of a partial-product generator, a binary-to-radix-2 redundant positive-digit number converter, a parallel radix-2 redundant positive-digit adder, a radix-2 redundant positive-digit-number-to-binary converter, and a carry lookahead adder [26].

#### A. Partial-Product Generator and Binary-to-Radix-2 Converter

In the radix-2 positive-digit number, any  $n$ -digit positive integer  $S$  is denoted as  $S = (s_{n-1} \cdots s_1 s_0)$  and has the value

$$S = \sum_{i=0}^{n-1} s_i 2^i \quad (8)$$

where  $s_i \in \{0, 1, 2\}$ .



		$p_{ij}$	
$s$		0	1
$p_{mn}$	0	0	1
	1	1	2
(a)			
$c_i^{(2)}$		$x_i$	
		0	1
$y_i$	0	0	0
	1	0	0
	2	0	1
$c_i^{(1)}$		$x_i$	
		0	1
$y_i$	0	0	0
	1	0	1
	2	1	1
$w_i$		$x_i$	
		0	1
$y_i$	0	0	1
	1	1	0
	2	0	1
(b)			
$d_i$		$c_{i-1}^{(1)}$	
		0	1
$w_i$	0	0	0
	1	0	1
$v_i$		$c_{i-1}^{(1)}$	
		0	1
$w_i$	0	0	1
	1	1	0
(c)			
$s_i$		$c_{i-2}^{(2)} d_{i-1}$	
		00	01
$v_i$	0	0	1
	1	1	2
	2	2	x
x : don't care			
(d)			
$s_i$		$a_i$	$b_i$
		0	0
$v_i$	0	0	0
	1	1	0
$v_i$	2	0	1
(e)			

Fig. 16. Truth tables of (a) radix-2 redundant positive-digit partial product generator, (b) intermediate sum digit and carry digits, (c) intermediate sum digit and carry digit, (d) final sum, and (e) radix-2 redundant positive-digit-number-to-binary converter in the new parallel multiplier.

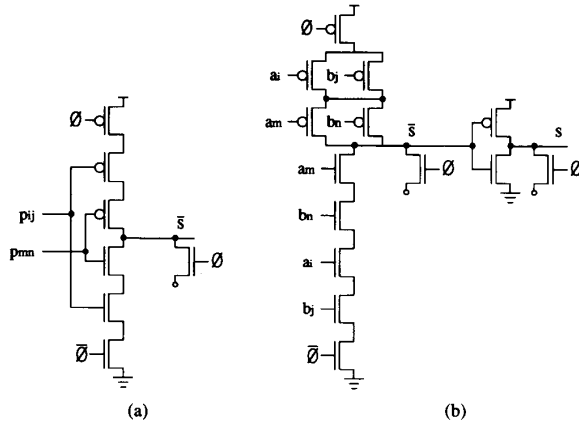


Fig. 17. Circuit configurations of the partial product generation for (a) the addition of two binary partial products, and (b) the direct implementation with binary multiplicand and multiplier as the inputs.

The binary-to-radix-2 conversion is designed by the addition of the two partial product terms  $p_{ij}$  and  $p_{mn}$ . The truth table is shown in Fig. 16(a). The logic functions are

$$s = p_{ij} + p_{mn} \quad (9)$$

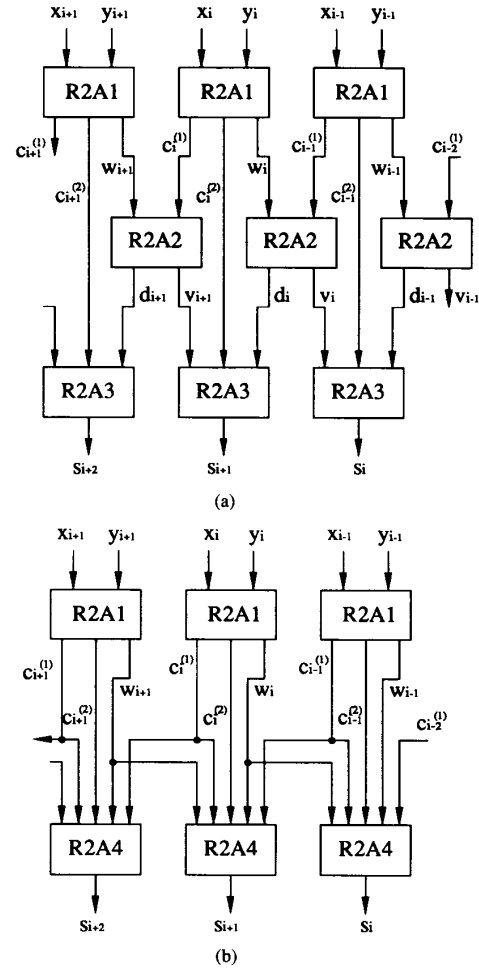


Fig. 18. Block diagrams of (a) the parallel radix-2 redundant positive-digit adders, and (b) the final sum directly derived from the intermediate sum digit and the carry digits.

$$p_{ij} = a_i b_j \quad (10)$$

$$p_{mn} = a_m b_n \quad (11)$$

where  $s \in \{0, 1, 2\}$  and  $p_{ij}, p_{mn}, a_i, b_j, a_m, b_n \in \{0, 1\}$ .

Fig. 17(a) shows the STG adder for the addition of the two partial products  $p_{ij}$  and  $p_{mn}$ . Fig. 17(b) shows the STG adder for the direct implementation with the binary inputs  $a_i, b_j, a_m$ , and  $b_n$  from both multiplicand and multiplier. Thus the partial products are directly generated and used to form the radix-2 redundant positive-digit number. For an  $n \times n$ -bit multiplier, the total partial-product operands are reduced to  $n/2$  without using the modified Booth algorithm.

### B. Parallel Radix-2 Redundant Positive-Digit Adder

The truth tables of the radix-2 redundant positive-digit adder are shown in Fig. 16(b)–(d). The parallel addition of the two  $n$ -bit radix-2 redundant positive-digit numbers

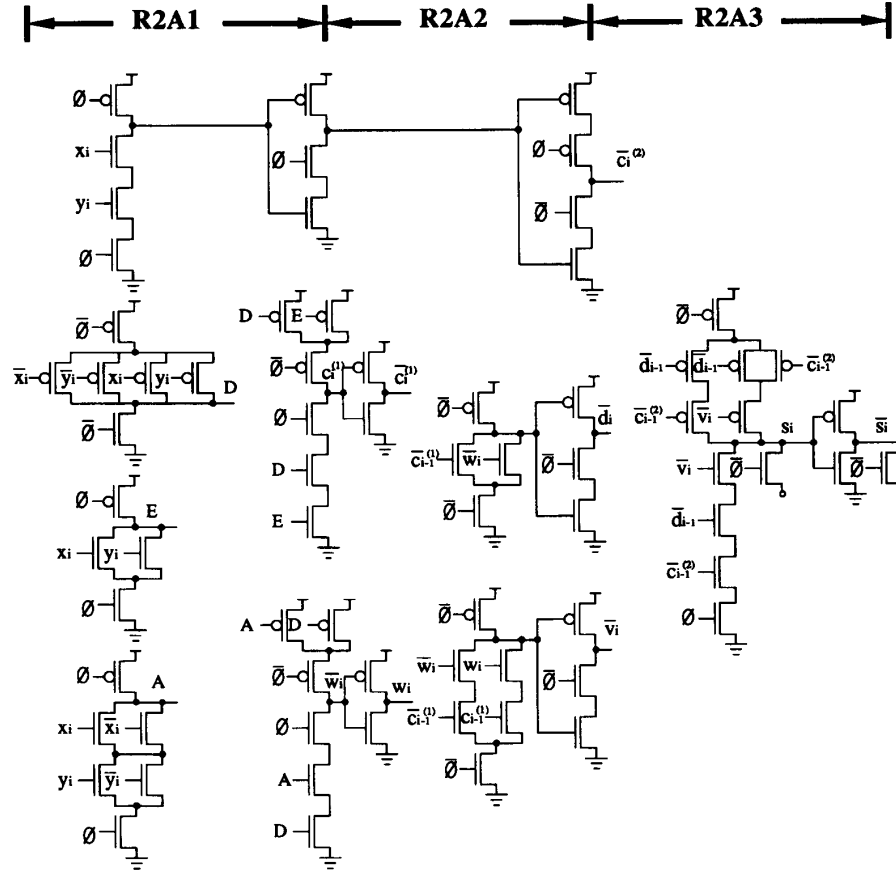


Fig. 19. Circuit configuration of Fig. 18(a).

$X = (x_{n-1} \cdots x_1 x_0)$  and  $Y = (y_{n-1} \cdots y_1 y_0)$  is described by the following three steps [26]:

1. generate the intermediate sum digit  $w_i$  and the two carry digits  $c_i^{(1)}$  and  $c_i^{(2)}$ ;
2. generate the intermediate sum digit  $v_i$  and the carry digit  $d_i$ ;
3. linearly add up  $v_i$ ,  $d_{i-1}$ , and  $c_{i-1}^{(2)}$  to obtain the final sum  $s_i$ ,

where  $w_i, c_i^{(1)}, c_i^{(2)}, v_i, d_i \in \{0, 1\}$  and  $s_i \in \{0, 1, 2\}$ .

Fig. 18(a) shows the structure of the parallel addition for the two operands, which are the radix-2 redundant positive-digit numbers.  $R2A1$ ,  $R2A2$ , and  $R2A3$  denote the cells that perform the operations in step 1, step 2, and step 3, respectively. The final sum  $s_i$  depends on the input digits  $(x_i, y_i)$ ,  $(x_{i-1}, y_{i-1})$ , and  $(x_{i-2}, y_{i-2})$ . The parallel adder cell is designed as shown in Fig. 19. The cell  $R2A1$  is designed to derive the intermediate sum digit  $w_i$  and the carry digits  $c_i^{(1)}$  and  $c_i^{(2)}$  and to form the first pipelined stage. The NAND gate structures of C<sup>2</sup>MOS latch stages in the cell  $R2A1$  are used for race-free operations and to form the pipelined system. In Fig. 19, the radix-2 redundant positive-digit numbers  $x_i$

and  $y_i$  are connected to the 2-b decoders in the cell  $R2A1$  for the optimized design with the least interconnections and devices [27]–[29]. The cell  $R2A2$  is designed to derive the intermediate sum digit  $d_i$  and the carry digit  $v_i$  and to form the second pipelined stage.

The Boolean functions in Fig. 19 are described as follows:

$$A = x^0 y^0 + x^0 y^2 + x^2 y^0 + x^2 y^2 \quad (12)$$

$$\begin{aligned} D &= x^0 + y^0 + x^2 + y^2 \\ &= x^{02} + y^{02} \\ &= \overline{x^1} + \overline{y^1} \\ &= \overline{x^1 y^1} \end{aligned} \quad (13)$$

$$E = \overline{x^2 + y^2} \quad (14)$$

$$c_i^{(2)} = \overline{x^2 y^2} \quad (15)$$

$$\begin{aligned} c_i^{(1)} &= x^1 y^1 + x^2 + y^2 \\ &= \overline{D} + \overline{E} \\ &= \overline{DE} \end{aligned} \quad (16)$$

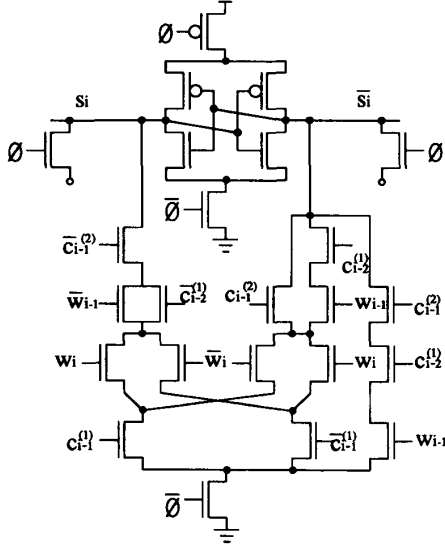


Fig. 20. Circuit configuration of R2A4 in Fig. 18(b) designed by using a STDL gate.

$$\begin{aligned}
 \bar{w}_i &= x^0 y^0 + x^0 y^2 + x^2 y^0 + x^2 y^2 + x^1 y^1 \\
 &= \bar{A} + \bar{D} \\
 &= \overline{AD}.
 \end{aligned} \tag{17}$$

The final sum is implemented by the cell R2A3 using an STG. It combines with the cell R2A1 of the next adders to form the other pipelined stage. Thus, two pipelined stages are required in each adder of the structure in Fig. 18(a). Fig. 18(b) shows the other structure of the pipelined radix-2 redundant positive-digit adder. In this structure, the final sum is implemented by using the cell R2A4 whose inputs are the intermediate sum digits  $w_i$  and  $w_{i-1}$  and the carry digits  $c_{i-1}^{(1)}$ ,  $c_{i-1}^{(2)}$ , and  $c_{i-2}^{(1)}$ . The cell R2A4 is implemented by the STDL as shown in Fig. 20. It combines with the cell R2A1 in the next adder to form a pipelined stage. Thus, only one pipelined stage is required for each adder.

### C. Radix-2 Redundant Positive-Digit-Number-to-Binary Conversion

The parallel radix-2 redundant positive-digit adders reduce the  $n/2$  partial-product operands to a single operand in the digit set  $\{0, 1, 2\}$ . Fig. 21 shows the implementation of the radix-2 redundant positive-digit-number-to-binary converter. The truth table is shown in Fig. 16(e). The NAND and NOR structures of C<sup>2</sup>MOS latch stages are used for race-free operations and to derive the carry propagate  $p_i$  and the carry generate  $g_i$  for the following carry lookahead adders.

### D. Carry Lookahead Adders

The final product is obtained from the addition of the last two operands, which is implemented by the binary carry lookahead adders. Every 4-b carry lookahead adder is connected to a latch to form a pipelined stage.

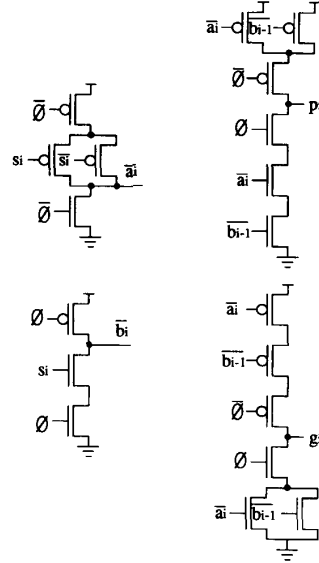


Fig. 21. Circuit configuration of radix-2 redundant positive-digit-number-to-binary converter, carry propagate, and carry generate.

TABLE VI  
COMPARISONS OF THE NEW PIPELINED MULTIPLIER BY USING  
INTERNAL RADIX-2 REDUNDANT POSITIVE-DIGIT NUMBER WITH  
CONVENTIONAL BINARY PARALLEL PIPELINED MULTIPLIER

Comparisons Structures	Number of bits	Pipelined stages (latency)	Total device counts	Operating frequency (unscaled)	Operating frequency (scaled)
Radix-2 redundant Positive-digit number	16 × 16	12	7700	50 MHz	75 MHz
	32 × 32	21	23800	50 MHz	75 MHz
	64 × 64	38	75200	50 MHz	75 MHz
binary number	16 × 16	32	11500	40 MHz	50 MHz
	32 × 32	64	46600	40 MHz	50 MHz
	64 × 64	128	187400	40 MHz	50 MHz

Table VI shows the comparisons between the radix-2 positive-digit pipelined multiplier in Fig. 15 with the structure of Fig. 18(b) and the conventional binary parallel pipelined multiplier designed by using the two-phase NORA system [17]. They are simulated with the same power supply  $2/3V_{DD}$  and device model parameters. As seen in Table VI, the pipelined multiplier internally using the radix-2 redundant positive-digit adder is superior in terms of the total delay of the pipelined stages (latency) and the device count. The total system delay will be large if the latency of the multiplication is large. The long system delay will weaken the environment tracing capability of the system, which is a critical issue in many real-time applications. Meanwhile, large total device count increases the chip area and cost. Therefore, low latency and small device count could lead to certain advantages in VLSI design.

## VI. CONCLUSION

In this paper, new dynamic ternary logic circuits have been developed. The developed ternary inverters have no dc power dissipation. Moreover, a new ternary differential logic gate called the simple ternary differential logic (STDL) and

its design procedures have also been developed. A suitable algebra is selected for easy implementation. The new dynamic ternary logic circuits are shown to have better performance in speed, power dissipation, layout area, and clock complexity than other dynamic ternary logic circuits. In particular, the dynamic ternary logic can form a pipelined structure without the race problem. The performance of the proposed STDL has been partly verified through an experimental chip. A new structure of binary pipelined multiplier is designed by using the new dynamic ternary logic circuits. It is shown that the new structure has much less latency and total device count than the corresponding binary parallel pipelined multiplier. Moreover, the operating frequency has also been improved.

#### ACKNOWLEDGMENT

The authors would like to thank Dr. J.-S. Wang for his fruitful discussions, and the Winbond Electronics Corporation, Taiwan, R.O.C., for the fabrication of the experimental chip.

#### REFERENCES

- [1] K. C. Smith, "The prospects for multivalued logic: A technology and applications view," *IEEE Trans. Comput.*, vol. C-30, pp. 619-634, Sept. 1981.
- [2] S. L. Hurst, "Multivalued logic—Its status and its future," *IEEE Trans. Comput.*, vol. C-33, pp. 1160-1179, 1984.
- [3] H. T. Mouftah and I. B. Jorden, "Integrated circuits for ternary logic," in *IEEE Proc. ISMVL*, May 1974, pp. 285-302.
- [4] H. T. Mouftah and K. C. Smith, "Design and implementation of three-valued logic systems with m.o.s. integrated circuits," *Proc. Inst. Elec. Eng.*, vol. 127, pt. G, pp. 165-168, Aug. 1980.
- [5] H. T. Mouftah and K. C. Smith, "Injected voltage low-power CMOS for three-valued logic," *Proc. Inst. Elec. Eng.*, vol. 129, pt. G, pp. 270-272, Dec. 1982.
- [6] M. Li and W. N. Gu, "The new method of implementation for ternary logic," in *IEEE Proc. ISMVL*, May 1983, pp. 56-60.
- [7] H. M. Aytac, "Ternary logic based on a novel MOS building block circuit," in *IEEE Proc. ISMVL*, May 1986, pp. 20-25.
- [8] P. Balla and A. Antoniou, "Low power dissipation MOS ternary logic family," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 739-749, 1984.
- [9] A. Hueng and H. T. Mouftah, "Depletion/enhancement CMOS for a low power family of three-valued logic circuits," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 609-616, Apr. 1985.
- [10] X. W. Wu and F. P. Prosser, "CMOS ternary logic circuits," *Proc. Inst. Elec. Eng.*, vol. 137, pt. G, pp. 211-217, Feb. 1990.
- [11] J. L. Huertas, A. Barriga, and G. Sanchez-Gomez, "Multivalued dynamic circuits," *Electron. Lett.*, vol. 23, pp. 502-504, 1987.
- [12] J. S. Wang, C. Y. Wu, and M. K. Tsai, "Low power dynamic ternary logic," *Proc. Inst. Elec. Eng.*, vol. 135, pt. G, pp. 221-230, Dec. 1988.
- [13] M. Yoeli and G. Rosenfeld, "Logical design of ternary switching circuits," *IEEE Trans. Comput.*, vol. C-14, pp. 19-29, Feb. 1965.
- [14] C. Y. Wu and H. Y. Huang, "A new two-phase pipelined dynamic CMOS ternary logic," in *IEEE Proc. ISCAS*, May 1990, pp. 582-586.
- [15] I. Halpern and M. Yoeli, "Ternary arithmetic unity," *Proc. Inst. Elec. Eng.*, vol. 115, pt. G, pp. 1385-1388, Oct. 1968.
- [16] V. Friedman and S. Liu, "Dynamic logic CMOS circuits," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 263-266, Apr. 1984.
- [17] N. F. Goncalves and H. J. De Man, "NORA: A race free dynamic CMOS technique for pipelined logic structures," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 261-266, June 1983.
- [18] Y. Jiren, I. Karlsson, and C. Svensson, "A true single-phase-clock dynamic CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 899-901, Oct. 1987.
- [19] Y. Jiren and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 24, pp. 62-70, Feb. 1989.
- [20] M. Afghahi and C. Svensson, "A unified single-phase clocking scheme for VLSI system," *IEEE J. Solid-State Circuits*, vol. 25, pp. 225-233, Feb. 1990.
- [21] S. L. Lu, "Implementation of iterative network with CMOS differential logic," *IEEE J. Solid-State Circuits*, vol. 23, pp. 1013-1017, Aug. 1988.
- [22] C. Y. Wu and K. H. Cheng, "Latched CMOS differential logic (LCDL) for complex high-speed VLSI," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1324-1328, Sept. 1991.
- [23] N. C. Lu and H. H. Chao, "Half-VDD bit-line sensing scheme in CMOS DRAM's," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 451-454, Aug. 1984.
- [24] J. A. Pretorius, A. S. Schubat, and C. A. Salama, "Charge redistribution and noise margins in domino CMOS logic," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 786-793, Aug. 1986.
- [25] K. M. Chu and D. L. Pulfrey, "Design procedures for differential cascode voltage switch circuits," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 1082-1087, Dec. 1986.
- [26] S. Kawahito, Y. Mitsui, M. Ishida, and T. Nakamura, "Parallel hardware algorithms with redundant number representations for multi-valued arithmetic VLSI," in *IEEE Proc. ISMVL*, May 1992, pp. 337-345.
- [27] T. Sasao, "Input variable assignment and output phase optimization of PLA's," *IEEE Trans. Comput.*, vol. C-33, pp. 879-894, Oct. 1984.
- [28] T. Sasao, "Multiple-valued logic and optimization of programmable logic arrays," *IEEE Comput.*, pp. 71-80, Apr. 1988.
- [29] T. Sasao, "On the optimal design of multiple-valued PLA's," *IEEE Trans. Comput.*, vol. 38, pp. 582-592, Apr. 1989.



**Chung-Yu Wu** (S'75-M'77) was born in Chiayi, Taiwan, Republic of China, in 1950. He received the B.S. degree from the Department of Electrophysics, and the M.S. and Ph.D. degrees from the Institute of Electronics, National Chiao-Tung University, Hsinchu, Taiwan, in 1972, 1976, and 1980, respectively.

During 1975-1976 he studied ferroelectric films on silicon and their device applications. During 1976-1979 he engaged in the development of integrated differential negative resistance devices and their circuit applications, with support from the National Electronics Mass Plan (Semiconductor Devices and Integrated Circuit Technologies) of the National Science Council. From 1980 to 1984 he was an Associate Professor at the Institute of Electronics, National Chiao-Tung University. During 1984-1986 he was an Associate Professor in the Department of Electrical Engineering, Portland State University, Portland, OR. Presently he is a Professor in the Department of Electronics Engineering and Institute of Electronics, National Chiao-Tung University. His research interests have been in analog and digital integrated circuits and systems, special semiconductor devices, and neural networks.

Dr. Wu is a member of Eta Kappa Nu and Phi Tau Phi.



**Hong-Yi Huang** (S'92) was born in Changhua, Taiwan, Republic of China, in 1965. He received the B.S. degree from the Department of Nuclear Engineering, National Tsing-Hua University, Hsinchu, Taiwan, and the M.S. degree from the Institute of Electronics, National Chiao-Tung University, Hsinchu, Taiwan, in 1987 and 1989, respectively. He is currently working toward the Ph.D. degree at the same institute. His main research interests have been in multivalued logic and high-speed digital integrated circuits and systems.