

Transactions Briefs

A 1.0-GHz 0.6- μm 8-bit Carry Lookahead Adder Using PLA-Styled All-N-Transistor Logic

Chua-Chin Wang, Chenn-Jung Huang, and Kun-Chu Tsai

Abstract—This article presents a high-speed 8-bit carry-lookahead adder (CLA) using two-phase clocking dynamic CMOS logic with modified noninverting all-N-transistor (ANT) blocks which are arranged in a programmable logic array design style. Detailed simulation reveals appropriate L/W guidelines for the ANT block design. The area (transistor count) tradeoff is also analyzed. The operating clock frequency is 1.0 GHz, while the output of the addition of two 8-bit binary numbers is completed in two cycles. Simulation results confirm that the proposed design methodology is appropriate for the long adders, e.g., 64-bit adders, while the correct output is available after four cycles if the 64-bit adder is composed of nine hierarchical 8-bit CLA's.

Index Terms—Noninverting block, PLA-styled design, two-phase clocking.

I. INTRODUCTION

Improving adder designs has received considerable attention [1]. CMOS dynamic logic is one of the options promising to challenge gigahertz operation in adder design [2]. However, all-N-logic [2] and robust single-phase clocking [3] cannot operate correctly under clocks with short rise or fall time. While single-phased clocked logic-flip-flops are pipelined to assemble it, an adder can only process short data words [1]. Therefore, this work presents an all-N-transistor (ANT) noninverting function block for the high-speed design. Also proposed herein is an 8-bit carry-lookahead adder (CLA) using ANT's arranged in a programmable logic array (PLA)-like structure and triggered by a single clock. This design methodology is advantageous in that it is scalable such that long data words, e.g., 64-bit binary data, can also be processed. The 8-bit CLA using PLA-styled ANT logic is measured to be fully functional at up to 1.0 GHz with a 5.0-V power supply, and the precise result of the addition is available after two cycles.

II. HIGH-SPEED 8-BIT CLA

A. ANT Function Unit

Although possessing high speed, the N-block dynamic logic [1] is inefficient for operation in the gigahertz range for two reasons: the slopes of the clock's edges are not gentle, and the number of stacks in the evaluation N-block significantly affects the size of all of the transistors in the unit. Therefore, a modified dynamic logic is presented in Fig. 1. The main feature of this modification is the presence of feedback transistor pair, P3 and N3, between the evaluation block and the output. In short, P3 and N3, respectively, provide an extra charging and discharging path, thus accelerating the evaluation. Notably, the ANT in

Fig. 1 is noninverting. The detailed operation of the ANT is described as follows.

- 1) When $clk = 0$, P1 is on and the gate at P2 is precharged to V_{dd} . Then, P2 is off and N4 is off. This maintains the output in the previous state.
- 2) When $clk = 1$ and the N-block is evaluated as "pass," the charge at node a should theoretically be grounded through the N-block and N1. Note that N4 is on and N2 is also on at the beginning. If the previous output is high, then N3 will be turned on via N4. This finding implies that N3 provides another fast discharging path for the charge at node a . When the voltage at node a is decreased below the threshold voltage of pMOS, P2 and P3 turn on. The output is then charged to V_{dd} through paths P2 and P3-N4.
- 3) When $clk = 1$ and the previous output is low, and the N-block is evaluated as "pass," the voltage at node a starts to fall. When $V_a - V_{dd} > V_{tp}$, P3 is on, and the gate of N3 is charged to V_{dd} . The charge at node a is discharged faster, and the output is charged to high through P2 and N4. In summarizing 2) and 3) above, the output is high when the N-block is evaluated as "pass," i.e., "1," during $clk = 1$.
- 4) When $clk = 1$ and the N-block is evaluated as "stop," the charge at node a is maintained if the previous output is low. There is no discharging path for node a because N3 is turned off via N4. If the previous state is high, the output is grounded via N4 and N2 before the voltage at node a starts to fall.

Therefore, the output will be low when the N-block is evaluated as "stop," i.e., "0" during $clk = 1$. The functioning of the ANT logic block, thus, is conclusively correct and noninverting. The feedback transistor pair, N3 and P3, indeed provides extra paths for the operations.

B. Sizing Problem

A reason why other high-speed logics can not run correctly given clocks with short rise time or fall time is that the size of each transistor can not be tuned properly. Both [2] and [3] possess this inherent shortcoming, accounting for why they cannot use normal square-wave clocks in the gigahertz range. Thus, the sizing problem of the transistors in the ANT, let alone those in the N-block, significantly affects the speed.

Theoretically, wider transistors have lower resistance. Herein, we conducted several simulations to obtain the optimal figure for the sizing of each transistor given in Fig. 1 using TSMC 0.6- μm SPDM technology; the results are summarized in Table I.

C. PLA-Styled 8-Bit CLA Design

The formulation of an 8-bit CLA is represented by the following equations:

$$\begin{aligned} S_i &= C_{i-1} \oplus P_i \\ C_i &= G_{i-1} + P_{i-1}G_{i-2} + P_{i-1}P_{i-2}G_{i-3} + \cdots + P_{i-1} \cdots P_1P_0C_0. \end{aligned} \quad (1)$$

where A_i , B_i , $i = 0, \dots, 7$ are inputs, and $P_i = A_i \oplus B_i$, $G_i = A_i \cdot B_i$ are propagate and generate signals, respectively.

If the P_i 's and G_i 's are produced by combinatorial logic function blocks before they are fed into the function blocks for S_i 's and C_i 's, then (1) implies that a two-level AND-OR logic function block is a pos-

Manuscript received October 1997; revised October 1999. This paper was recommended by Associate Editor Y. Leblebici.

The authors are with the Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan 80424 R.O.C.

This work was supported in part by the National Science Council under Grant NSC 88-2219-E-110-001.

Publisher Item Identifier S 1057-7130(00)01458-0.

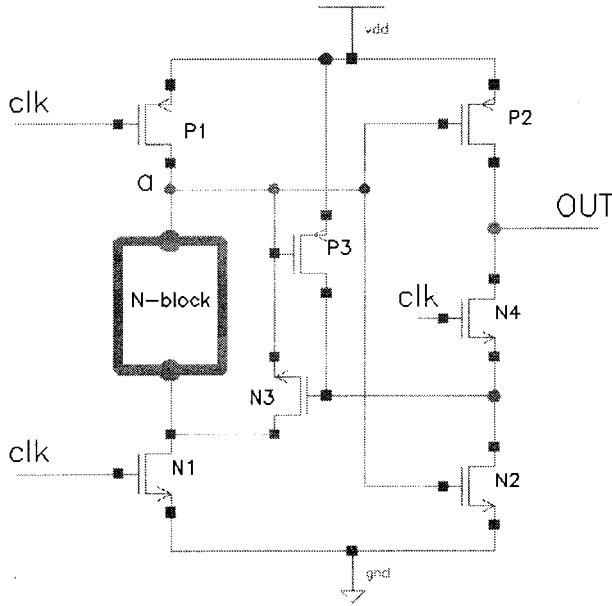


Fig. 1. The schematic diagram of the modified dynamic logic.

TABLE I
SIZES OF ANT LOGIC BLOCK

Transistor	L (μm)	W (μm)
N1	0.6	15
N2	0.6	10
N3	0.6	3
N4	0.6	10
P1	0.6	20
P2	0.6	20
P3	0.6	6
N-block	0.6	10

sible solution to achieve high speed operations. Thus, the PLA-styled design is appropriate for such a function block.

A conceptual PLA-styled design for CLA is presented in Fig. 2. A typical PLA consists of an AND array and an OR array. As is well known, the series nMOS in the evaluation block of NAND or AND gates produces long discharging delays, subsequently decelerating the entire circuit. The noninverting feature of the ANT logic can be taken advantage of to utilize a NOT-OR-NOT-OR configuration instead of the typical AND-OR style, where the two OR planes are made of ANT logic blocks. The noninversion feature can also minimize the series transistor count in the evaluation block. The OR array consists of ANT logic with a predefined evaluation block. The inputs to the first OR array are the inverted P_i (propagate) and G_i (generate) signals which are also produced by other ANT logic units. Notably, the propagate signals are defined differently from the traditional $1P_i = A_i + B_i$ because $P_i = A_i \oplus B_i$ can be reused to generate the sum term S_i .

D. Speed and Area Analysis

Speed: the critical path of an adder resides in the generation of carry signals, i.e., C_7 in the 8-bit adder. When the binary data are ready, the generation of P_i 's and G_i 's using ANT logic requires the high half of a full cycle. The inverted P_i 's and G_i 's are then be fed into the first OR plane of the ANT-based PLA. The inverted outputs of the first OR plane are presented to the second OR in the high half of the second cycle. The final C_i results are then ready in the low half of the second cycle. As soon as it is generated, each C_i is inverted and fed into the S_i 's function

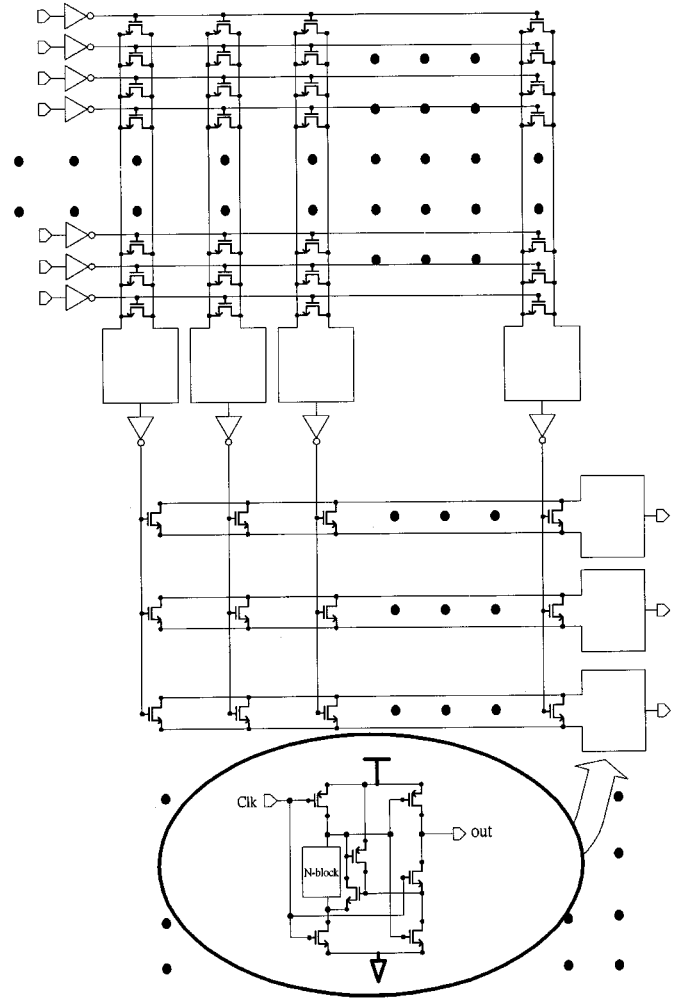


Fig. 2. A conceptual PLA-styled design for CLA.

blocks. Another half cycle is then required to produce all of the S_i 's. The final result is available after two cycles.

Area: for an 8-bit CLA adder using PLA-styled ANT logic, the total transistor count is 928.

E. Design of n -Bit CLA Adders

The delay is the same, i.e., two cycles, even if n is large, as long as the PLA-styled design is adopted, because the critical path from input to output is unchanged. The transistor count of the PLA-styled implementation for CLA using ANT logic strictly adheres to the following general forms.

Carry Out: The number of total product terms in the first OR plane is summarized as

$$T_p = 1 + 2 + 3 + \cdots + n = \sum_{i=1}^n i = \frac{1}{2}n(n+1). \quad (2)$$

The number of the nMOS residing in the evaluation blocks for these product terms is

$$\begin{aligned} N_p &= \sum_{i=2}^{n+1} \sum_{j=2}^i j = \sum_{i=2}^{n+1} \left[\frac{1}{2}i(i+1) - 1 \right] \\ &= \frac{1}{6}(n+1)(n+2)(n+3) - (n+1). \end{aligned} \quad (3)$$

Now, the total number of transistors in the first OR plane, excluding the inverters required between the first and second OR planes, is $T_p \cdot 7 +$

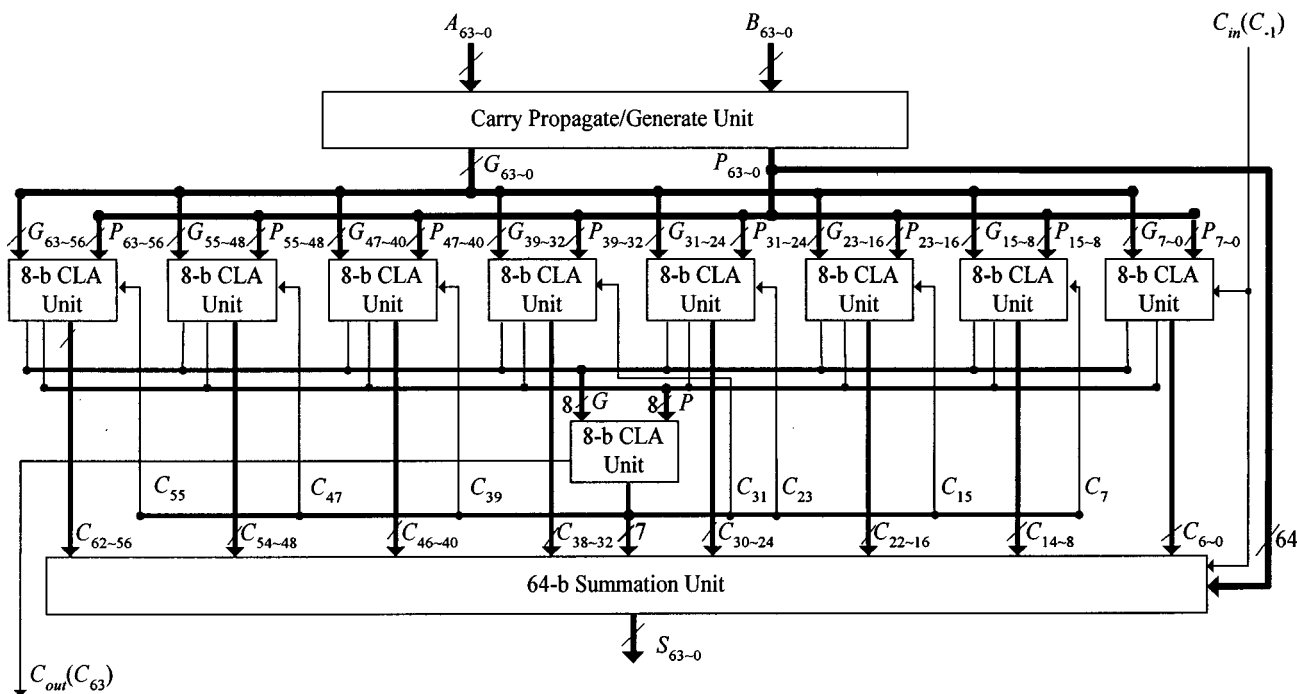


Fig. 3. The hierarchical architecture of 64-bit CLA.

N_p , where each ANT logic block requires seven transistors according to Fig. 1. The number of output terms in the second OR plane is n . Now, the total transistor count for the second OR plane is $n \cdot 7 + T_p + n$.

Inverters: signals which must be inverted include the input data words, clock, C_{in} , P_i 's, and G_i 's, and the outputs of the first OR plane. Therefore, the total number of transistors for the inverters is $I = 2 \cdot (n \cdot 4 + (1/2)n(n+1) + 2)$.

P_i 's and G_i 's generation: the total number of transistors is $n \cdot (2 + 7) + n \cdot (4 + 7)$.

S_i 's generation: the total transistor count is $n \cdot (4 + 7) + n \cdot 4 = 15n$, where $n \cdot 4$ is the number of transistors required to invert C_i 's and P_i 's.

In sum, the total number of transistors required to implement an n -bit CLA with PLA-styled design using ANT logic is

$$\begin{aligned}
 T_{total} &= [T_p \cdot 7 + N_p] + [n \cdot 7 + T_p + n] \\
 &+ \left[2 \cdot \left(n \cdot 4 + \frac{1}{2}n(n+1) + 2 \right) \right] \\
 &+ [n \cdot (2 + 7) + n \cdot (4 + 7)] + [n \cdot (4 + 7) + n \cdot 4] \\
 &= \frac{1}{6}(n+1)(n+2)(n+3) + 5n(n+1) + 50n + 3. \quad (4)
 \end{aligned}$$

Notably, a 64-bit adder can theoretically be constructed using the same PLA-style design methodology, and a delay of two cycles (nanoseconds) is expected. However, the number of transistors would then be over 70 000, which is very large. Another alternative is to use a hierarchical design, employing nine 8-bit CLA's, as shown in Fig. 3. The tradeoff is that the delay would then be four cycles (nanoseconds).

III. PERFORMANCE SIMULATIONS AND COMPARISON

The performance of the 8-bit CLA is verified using the ANT logic in a PLA-styled design. The TSMC 0.6- μm SPDM technology is adopted to simulate several adder designs using different logics. The clock rate is 1.0 GHz with a 0.01-ns rise time and the same fall time. The results are presented in Table II.

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT DESIGNS

Logic	Delay	# transistors	Technology
8-b PLA-ANT CLA	2.0 ns	928	0.6 μm
64-b PLA-ANT CLA	2.0 ns	71908	0.6 μm
64-b PLA-ANT hierarchical CLA	4.0 ns	8352	0.6 μm
32-b EMODL adder [4]	2.7 ns	1537(gates)	1.2 μm
8-b TSPC adder (1 μm) [1]	7.5 ns	1832	1.0 μm
All-N-logic [2]	Failed	2062	0.8 μm

IV. CONCLUSION

This work has proposed a novel high-speed PLA-styled ANT logic design for the implementing adders. Simulation results not only verify the accuracy of the function in the gigahertz range, but also indicate that the proper size of each transistor is adjusted such that a usual square-wave clock can be used to run the 64-bit long adder. The PLA-styled ANT-based structure, using only one clock, causes the result of an 8-bit adder to be obtained in two cycles (2.0 ns if the 1.0-GHz clock is used), or that of a hierarchical 64-bit adder to be available after four cycles. We also estimated the number of transistors (area) for larger long adders using the proposed approach.

REFERENCES

- [1] R. Rogenmoser and Q. Huang, "An 800-MHz 1 mm CMOS pipelined 8-bit adder using true single phase clocked logic-flip-flops," *IEEE J. Solid-State Circuits*, vol. 31, pp. 401–409, Mar. 1996.
- [2] R. X. Gu and M. I. Elmasry, "All-N-logic high-speed true-single-phase dynamic CMOS logic," *IEEE J. Solid-State Circuits*, vol. 31, pp. 221–229, Feb. 1996.
- [3] M. Afghahi, "A robust single phase clocking for low power, high-speed VLSI applications," *IEEE J. Solid-State Circuits*, vol. 31, pp. 247–253, Feb. 1996.
- [4] Z. Wang, G. A. Jullien, W. C. Miller, J. Wang, and S. S. Bizzan, "Fast adders using enhanced multiple-output domino logic," *IEEE J. Solid-State Circuits*, vol. 32, pp. 206–214, Feb. 1997.