
ENMODL을 이용한 32 비트 CLA 설계

김강철*, 이효상**, 송근호**, 서정훈***, 한석봉****

Design of 32-bit Carry Lookahead Adder Using ENMODL

Kang-Chul Kim, Sang-Hyo Lee, Geun-Ho Song,
Jeong-Hun Seo, Seok-Bung Han

요 약

본 논문에서는 기존의 동적 CMOS 논리회로보다 동작속도가 빠르고 면적이 작은 새로운 ENMODL (enhanced NORA MODL)의 설계방법을 제시하고, 이를 이용하여 32 비트 CLA(carry lookahead adder)를 구현하였다. 제안된 회로는 MODL(multiple output domino logic)의 출력 인버터를 제거하여 면적을 줄이고 동작속도를 증가시킬 수 있다. 0.8 μ m 이중금속 CMOS 공정으로 구현된 CLA는 시차문제가 발생하지 않았고, 3.9nS 이내에 32 비트 연산이 가능하였다.

Abstract

This paper presents an ENMODL(enhances NORA MODL) circuit and implements a high-speed 32 bit CLA(carry lookahead adder) with the new dynamic logics. The proposed logic can reduce the area and the propagation delay of carry because output inverters and a clocking PMOS of second stage can be omitted in two-stage MODL(multiple output domino logic) circuits. The 32-bit CLA is implemented with 0.8 μ m double metal CMOS process and the carry propagation delay of the adder is about 3.9 nS. The ENMODL circuits can improve the performance in the high-speed computing circuits depending on the degree of recurrence.

* 여수대학교 컴퓨터공학과 조교수

** 경상대학교 전자공학과 박사과정

*** 창원전문대학 전자통신과 전임강사

**** 경상대학교 전자공학과 교수

접수일자 : 1999년 11월 17일

1. 서론

CMOS 소자는 전력소비가 적고 집적도가 높으므로 대부분의 VLSI 칩에서 사용되고 있다. 특히 domino CMOS, NORA CMOS, Zipper CMOS 등과 같은 동적 CMOS는 기존의 정적 CMOS에 비하여 동작 속도가 빠른 장점을 갖는 동시에 소비 전력은 거의 동일하므로 이에 대한 연구가 많이 수행되고 있다.[1-4] 그러나 동적 CMOS를 사용하여 회로를 설계할 경우 내부 신호 지연에 의한 시차 문제(race problem), 누설전류(leakage current), 내부 함수블록 내에서 발생하는 전하 결합(charge coupling), 전하 재분배(charge redistribution) 현상 등으로 회로의 안정도가 떨어지며, 외부의 잡음 등에 쉽게 영향을 받는 문제점들이 있다.[5,6]

1988년에 동적 CMOS의 문제점들을 해결하면서 면적이 작고 빠른 동작속도를 얻을 수 있는 MODL(multiple output domino logic) CMOS 설계 방식이 제안되었다.[7,8] 이 방식은 기존의 domino CMOS 논리 구조에 프리차지(precharge) 소자와 출력 인버터(inverter)를 사용하여 다중 함수를 생성할 수 있도록 변형한 것이다. MODL 회로는 한 함수블록에 포함되는 부함수블록을 따로 만들지 않음으로 게이트 수를 줄일 수 있어 작은 면적에 빠른 연산 속도를 갖는 회로의 설계가 가능하고, 동적 CMOS의 누설전류나 전하 재분배 현상 등을 해결할 수 있다.[9]

신호처리나 마이크로프로세서 칩 등에서 가산기의 계산속도는 칩의 성능뿐만 아니라 그 칩을 사용한 시스템의 성능을 좌우하게 된다. 가산기는 어떠한 캐리 생성 방식을 사용하느냐에 따라 그 성능이 결정될 수 있으므로 면적과 속도를 고려하여 CSA(carry selection adder), RCA(ripple carry adder), CLA(carry look ahead adder) 등이 이용된다.[10-12] CLA는 RCA에 비해 게이트 수가 약 4 배이고 비트 수가 클수록 회로가 매우 커지고 복잡하게 되지만 고속연산이 필요한 경우에 사용된다. 따라서 마이크로프로세서나 DSP 칩 등에 사용되는 고성능 연산기의 설계를 위해서는 집적도가 높고, 소비전력이 작은 논리소자를 사용하여야 하며, 빠른 동작속도를 갖는 연산회로의 설계가 필수

적이다.

본 논문에서는 MODL 방식의 장점을 유지하면서 MODL 회로보다 면적과 신호 전파속도를 줄일 수 있는 ENMODL 회로 설계방식을 제안하고, ENMODL을 이용하여 32 비트 CLA를 설계하였다. HSPICE를 사용하여 ENMODL의 정상동작을 확인하고 캐리지연 시간을 측정하였다. 그리고 0.8 μ m 이중 금속 CMOS 공정으로 구현하여 테스트한 결과에서 32 비트 연산에 소요되는 시간은 약 3.9nS로 측정되었고, ENMODL 회로의 안정된 동작과 성능을 확인하였다..

본 논문의 II장에서는 ENMODL의 구조 및 동작에 대하여 설명하고, III 장에서는 이를 이용하여 32 비트 가산기를 설계한다. 그리고 IV 장에서는 ENMODL로 구현된 가산기의 연산결과를 측정하여 그 성능을 확인하고 V 장에서 결론을 기술한다.

II. ENMODL의 구조

식 (1)과 (2)와 같은 함수를 설계할 때, MODL 회로를 이용하면 1개의 논리 게이트로 2 개의 함수를 동시에 설계할 수 있으므로 부함수를 별도로 설계할 필요가 없어 면적을 감소시킬 수 있다. 이 경우에 회로가 동적 CMOS로 구성되므로 트랜지스터의 크기가 커져 트랜지스터 수의 감소에 의한 면적을 줄일 수 없으나, 게이트 수의 감소에 따른 금속 도선 등의 감소로 인하여 전체 면적이 줄어들 수 있고, 지연시간을 줄일 수 있다.

$$F1 = A + B \cdot C \dots\dots\dots(1)$$

$$F2 = D \cdot F1 = D \cdot (A + B \cdot C) \dots\dots\dots(2)$$

이 회로의 설계방식은 캐리 생성과 같은 반복적인 구조를 갖는 논리회로에 대해 적합한데, CLA와 같은 회로는 반복성이 매우 크므로 MODL 회로를 이용하여 회로를 설계하면 domino 논리 회로를 사용하여 설계한 것보다 트랜지스터 수를 많이 감소시킬 수 있다.

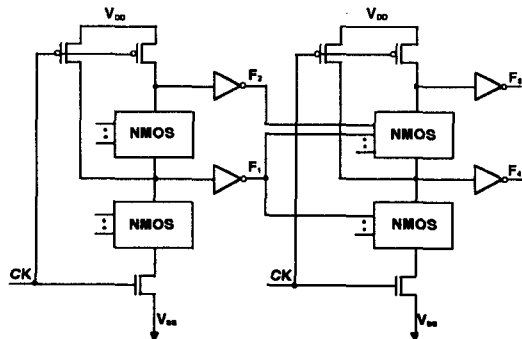
MODL은 그림 1(a)에서와 같이 2 단으로 구성되는 회로에서도 같은 논리 구조가 반복된다는 것을 알 수 있다. 그러나 MODL 회로의 첫 단 출력

의 인버터를 제거하고, 다음 단의 N 블록을 P 블록으로 바꾸면 다음 단의 인버터가 제거되어 면적을 줄일 수 있고, 2 게이트의 지연시간을 줄일 수 있는데 이러한 구조를 NMODL(NORA-MODL)이라 한다.

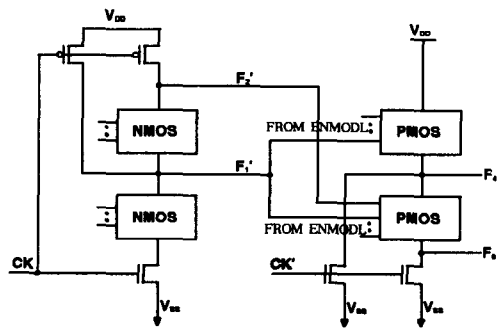
기본적인 동적 CMOS 게이트는 CK가 0 일 때 (precharge phase) CK에 연결된 PMOS를 통하여 프리차지 시키고, CK가 1 일 때 (evaluation phase) CK에 연결된 NMOS를 통하여 디스차지(discharge) 시킨다. 그러나 NMODL 회로에서는 두 번째 단계 사용되는 P 블록의 모든 입력이 앞 단 NMODL 게이트의 출력으로부터만 들어오는 경우 프리차지 시간 동안에 입력이 모두 HIGH가 되므로 클럭킹 소자의 PMOS는 회로의 동작에 아무런 영향을 미

치지 못한다. 따라서 CK에 연결되어 있는 클럭킹 소자의 PMOS를 그림 1(b)와 같이 제거할 수 있는데 이러한 구조를 ENMODL이라 한다.

ENMODL 회로는 함수블록 내부 노드에 다출력을 위한 클럭킹 프리차지 소자를 사용하여 내부 회로간에 발생하는 누설전류나 전하 재분배 현상 및 전하결합 등의 문제를 해결하여 출력 전압의 변동이 제거되고 잡음여유가 증가되어 안정된 회로로 동작할 수 있다. 여러 단으로 구성된 ENMODL 회로에서는 각 단의 함수 블록들이 홀짝에 따라 NMOS와 PMOS 블록이 교대로 사용되므로 프리차지 시간 동안에 홀수 단의 클럭킹 소자의 PMOS가 ON 되어 출력을 HIGH로 프리차지시키고, 짝수 단의 클럭킹 소자의 NMOS가 ON 되어 P 블록의 출력은 LOW로 프리차지되어 신호 지연에 의한 시차 문제를 해결할 수 있다. 또한 PMOS 블록에서 디스차지 소자인 PMOS를 제거하여도 그 블록의 입력이 인버터를 제거한 다이내믹 CMOS의 NMOS 블록으로부터만 들어온다면 프리차지 기간에는 각 트랜지스터가 OFF 상태를 유지한다. 따라서 입력 신호의 시차문제에 영향을 받지 않고 안정적인 동작을 할 수 있다.



(a) 2 단으로 구성된 MODL 회로
(a) MODL circuit designed by two-stage MODL



(b) ENMODL 회로
(b) ENMODL circuit

그림 1. MODL과 ENMODL 회로
Fig. 1. MODL and ENMODL circuits

III. 32 비트 가산기 설계

CLA는 캐리의 전파지연을 제거하여 속도를 증가시키므로 캐리생성항을 G_i , 캐리파항을 P_i , i 단의 입력 캐리를 C_i , 출력 캐리를 C_{i+1} 라 하면 다음 식으로 표시된다.[13]

$$G_i = A_i \cdot B_i \dots\dots\dots (3)$$

$$P_i = A_i \cdot B_i' + A_i' \cdot B_i \dots\dots\dots (4)$$

$$C_{i+1} = G_i + P_i \cdot C_i \dots\dots\dots (5)$$

i 가 커지게 되면 캐리발생회로가 커지게 되므로 분할하여 그룹캐리 생성항과 그룹캐리 전파항을 설계하여야 한다. 8 비트로 분할하여 이에 대한 그룹캐리 생성항(EGG_{i+7})과 그룹캐리 전파항(EGP_{i+7})을 ENMODL에서 사용하기 위하여 PMOS에 대한 부울함수로 표현하면 아래와 같다.

$$GP_{i+3} = EP_{i+3}' + EP_{i+1}' \dots\dots\dots (6)$$

$$GG_{i+3} = EG_{i+3} \cdot (EP_{i+3}' + EG_{i+1}') \dots\dots\dots (7)$$

$$EGP_{i+7} = EP_{i+7}' + EP_{i+5}' + EP_{i+3}' + EP_{i+1}' \dots\dots\dots (8)$$

$$EGG_{i+7} = EG_{i+7}' \cdot (EP_{i+7}' + EG_{i+5}' \cdot (EP_{i+5}' + EG_{i+3}' \cdot (EP_{i+3}' + EG_{i+1}')))) \dots\dots\dots (9)$$

$$\begin{aligned} EG_{i+1} &= G_{i+1} + G_i \cdot P_{i+1}, & EP_{i+1} &= P_{i+1} \cdot P_i \\ GP_{i+3} &= EP_{i+3} \cdot EP_{i+1}, & EP_{i+3} &= P_{i+3} \cdot P_{i+2} \\ EG_{i+3} &= G_{i+3} + G_{i+2} \cdot P_{i+3}, \\ GG_{i+3} &= EG_{i+3} + EP_{i+3} \cdot EG_{i+1} \\ EG_{i+5} &= G_{i+5} + G_{i+4} \cdot P_{i+5}, & EP_{i+5} &= P_{i+5} \cdot P_{i+4} \\ EG_{i+7} &= G_{i+7} + G_{i+6} \cdot P_{i+7}, & EP_{i+7} &= P_{i+7} \cdot P_{i+6} \end{aligned}$$

위에서 GP_{i+3} 와 GG_{i+3} 는 4 비트 그룹캐리 생성항과 전파항이다. 32 비트의 경우에 8 비트로 분할하여 I=8, 16, 24, 32에 대하여 캐리를 아래 식과 같은 부울함수로 나타낼 수 있다.

$$C_8 = EGG_7 + EGP_7 \cdot C_0 \dots\dots\dots (10)$$

$$C_{16} = EGG_{15} + EGP_{15} \cdot C_8 \dots\dots\dots (11)$$

$$C_{24} = EGG_{23} + EGP_{23} \cdot C_{16} \dots\dots\dots (12)$$

$$C_{32} = EGG_{31} + EGP_{31} \cdot C_{24} \dots\dots\dots (13)$$

본 논문에서 설계한 CLA는 캐리 생성 블럭과 합(SUM)을 구하는 블럭으로 구성된다. 합을 구하는 블럭은 기존의 동적 CMOS를 사용하여 XOR 게이트로 설계하였으며, 캐리 생성 블럭은 다중 함수를 출력하는 ENMODL 회로를 사용하여 설계하였다.

그림 2는 32 비트 CLA의 전체 블럭도를 나타낸 것으로 4 개의 8 비트 가산기로 구성된다. 계층(level) 1에는 입력 A, B를 받아 캐리 생성항과 캐리 전파항을 발생시키고, 계층 3으로부터 받아들이는 캐리를 이용하여 합을 발생시킨다. 계층 2에서는 계층 1로부터 얻어진 캐리 생성항과 전파항을 이용하여 8 비트 그룹캐리 생성항과 전파항을 만들어낸다. 이 계층은 P 블럭으로 이루어지므로 CK'가 사용된다. 계층 3은 캐리를 발생하는 부분으로 계층 1로부터 들어오는 입력를 이용하여 캐리(C₀-C₃₁)들을 발생하는 부분과 계층 2로부터 얻은 그룹캐리 생성항과 전파항을 이용하여 그룹캐리

C₈, C₁₆, C₂₄, C₃₂를 만드는 부분으로 이루어진다. C₃₂는 64 비트 가산기를 만들 때 이용될 수 있다.

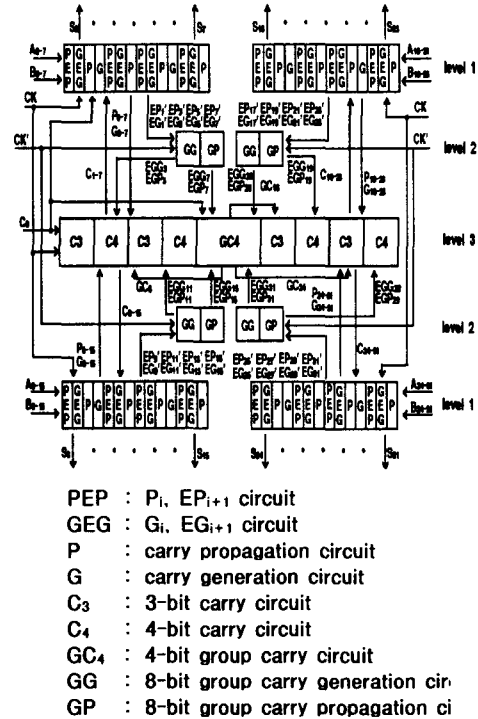
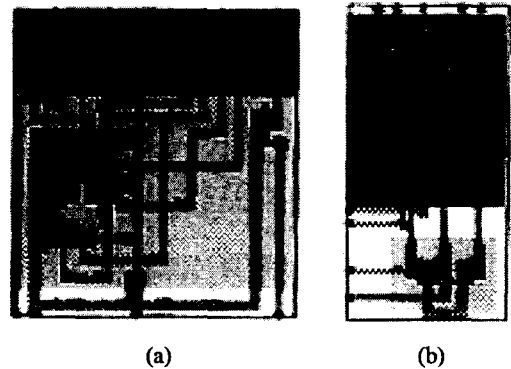
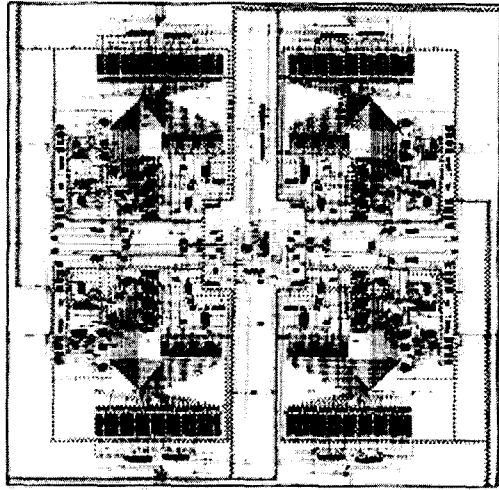


그림 2. 32 bit CLA의 블럭 다이어그램
Fig. 2. Block diagram of 32-bit CLA

그림 3은 ENMODL로 설계된 CLA에서 사용되는 게이트와 전체 레이아웃을 나타낸 것이다. (a)는 1, 2 비트 캐리전파 항(P)을 구하는 회로를 나





- (a) 1-, 2-비트 전파회로
- (a) 1-, 2-bit propagation circuit(Pi, Epi+1')
- (b) 4-, 8-비트 그룹캐리 전파회로
- (b) 4-, 8-bit group carry generation circuit(EGGi+3, EGGi+7)
- (c) CLA 레이아웃
- (c) CLA adder layout

그림 3. 32 비트 CLA의 레이아웃
Fig. 3. Layouts of gates and 32-bit CLA

타낸 것이고, 그림 (b)는 그룹캐리 생성 항(EGGi+7)을 발생하는 회로이다. (c)는 32 비트 CLA의 전체 레이아웃을 나타낸다.

IV. 모의실험 및 측정 결과

그림 4는 그림 1의 ENMODL과 NMODL을 HSPICE로 모의실험을 하여 동작 속도를 비교한 것이다. NMODL은 약 0.65 ns의 지연시간을, ENMODL은 약 0.35 ns의 지연시간을 보여주고 있다. 약 두배 정도로 빠르게 동작하고 있음을 알 수 있다.

32 비트 CLA의 연산시간은 가장 긴 지연시간이 걸리는 C31의 발생시간을 측정하여 구할 수 있다. 그림 5는 한쪽 입력에 16 진수 A=FFFFFFF(16), 다

른 입력에 B=00000000, C0 = 1을 인가하여 측정된 것이다. Δ는 클럭을 나타내고, ○는 C31의 출력 파형을 나타낸 것으로 약 2.9ns의 지연시간이 발생한다.

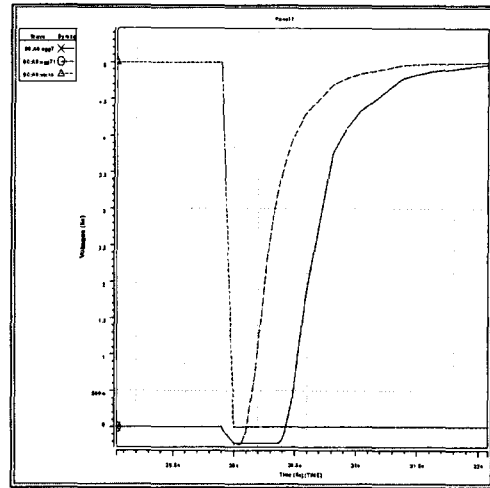


그림 4. NMODL과 ENMODL로 구성된 회로의 지연시간
Fig. 4. Propagation delays of NMODL and ENMODL circuits

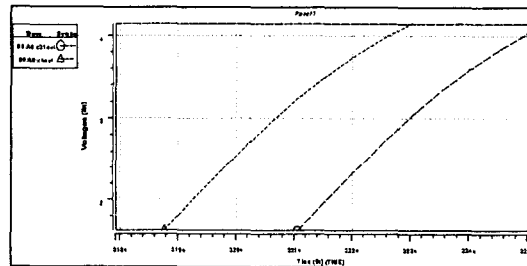
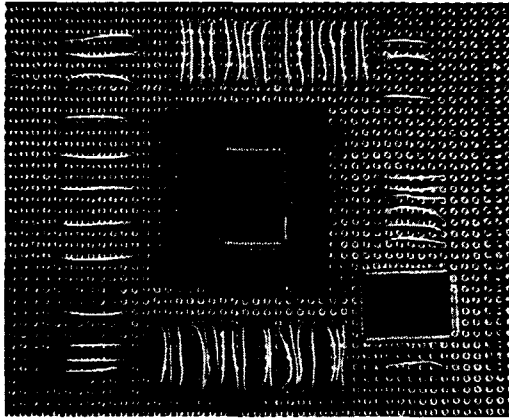


그림 5. C31의 지연시간 모의실험
Fig. 5. HSPICE simulation of delay time in C31

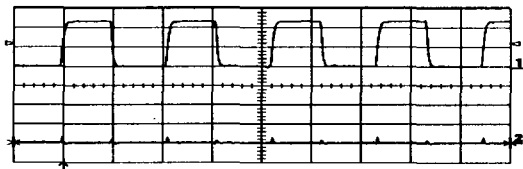
그림 6은 ENMODL 회로로 구현된 32 비트 가산기의 사진과 칩이 정상적으로 동작하는 것을 확인하기 위하여 오실로스코프를 사용하여 S31의 출력파형을 측정된 것이다. (b)는 A=FFFFFFF(16), B=00000000, C0 = 1을 인가하였을 때 32 비트의 합 S31을 측정된 것이다. 1은 클럭파형이고 2는 S31의 출력파형이다. C0의 캐리가 전파되어 S31의 출력과

형이 0이 됨을 알 수 있다. (c)는 $A=FFFFFFF_{(16)}$, $B=00000000$, $C_0 = 0$ 을 인가하였을 때 32 비트의 합 S_{31} 을 측정한 것이다. S_{31} 의 출력파형이 1로 나타난 곳을 확인할 수 있다.



(a) 제조된 칩 사진

(a) Photograph of the Chip



(b) $A=FFFFFFF_{(16)}$, $B=00000000_{(16)}$, $C_0=1$



(c) $A=FFFFFFF_{(16)}$, $B=00000000_{(16)}$, $C_0=0$

그림 6. CLA의 정상 동작 (1=CK, 2=Sum31)

Fig. 6. Normal operation of CLA

그림 7은 C_{31} 의 지연시간을 나타낸 것이다. 1은 클럭의 출력파형이고, 2는 C_{31} 의 출력파형을 나타내며, C_{31} 의 지연시간은 약 3.9nS이다. 2.3nS인 모의실험 결과와는 상당한 차이가 발생하였다. CLA를 모의실험할 때는 트랜지스터의 크기만 고려되었으나, 그림 3에 보여준 실제 레이아웃에서는 입

력 패드수가 많고, 셀을 반복사용하는 과정에서 도선의 길이가 길어졌고, 공정변수 등의 변화로 인하여 시뮬레이션과 측정 결과에서 차이가 발생한 것으로 고려된다.

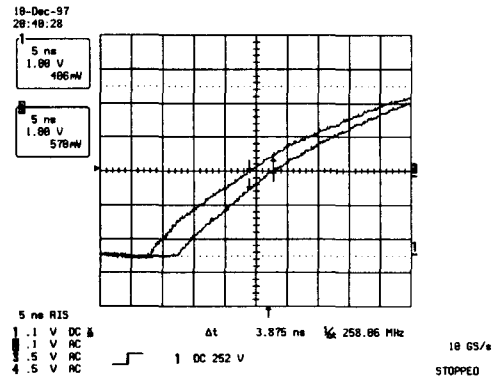


그림 7. C_{31} 의 지연시간

Fig. 7. Delay time of C_{31}

ENMODL로 구현된 32 비트 가산기에 여러 가지 입력값을 인가하여 정상적인 출력값이 나타남을 확인하였다. 그리고 동적 CMOS 로직에서 발생할 수 있는 시차문제와 전하 재결합에 의한 오동작도 발생하지 않았고, 전체회로는 안정된 상태에서 동작함을 확인하였다.

V. 결론

본 논문에서는 32 비트 CLA를 0.8 μm 이중 금속 CMOS 공정에 의해 칩으로 제작하여, 그 성능과 동작의 안정성을 측정하였다. ENMODL로 설계된 32 비트 CLA는 32 비트 연산을 약 3.9nS 이내에 수행할 수 있었다. 그리고 동적 CMOS 회로에서 발생할 수 있는 시차문제는 발생하지 않아 회로가 안정된 상태에서 동작함을 확인하였다. 본 연구에서 구현된 32 비트 CLA는 고속 연산이 가능하므로 빠른 연산시간을 요구하는 고성능 컴퓨터의 연산장치, 디지털 신호 처리 시스템과 반복성이 많은 회로에 사용될 수 있을 것이다.

감사의 글

본 연구는 과학기술부 한국과학재단지정 여수대학교 "설비자동화 및 정보시스템 연구개발센터"의 지원에 의한 것입니다.

참고문헌

[1] J. A. Pretorius, A. S. Shubat and C. A. T. Salama, "Analysis and design optimization of Domino CMOS logic with application to standard cells," IEEE Journal of Solid-State Circuits, vol. SC-20, pp. 523-530, April 1985.

[2] N. P. Goncalves and H. J. de Man, "NORA: a racefree dynamic CMOS technique for pipelined logic structures," IEEE Journal of Solid-State Circuits, vol. SC-18, pp.261-268, June 1983.

[3] C. M. Lee and E.W.Szeto, "Zipper CMOS," IEEE Circuits and Devices magazine, pp. 10-17, May 1986.

[4] R. H. Krambeck, C. M. Lee and H. F. S. Law, "High-speed compact circuits with CMOS," IEEE J. Solid-State Circuits. vol. SC-17. pp. 614-619. June 1982.

[5] J. A. Pretorius, A. S. Shubat and C. A. T. alama, "Charge redistribution and noise margins in Domino CMOS logic," IEEE Transactions on Circuits and Systems, vol. CAS-33, no.8, pp.768-793, Aug, 1986.

[6] David A. Hodges and Horace G. Jasckson, Analysis and Design of Digital Integrated Circuits, pp. 68-124, New York, 1983.

[7] I. S. Hwang and A. L. Fisher, "A 3.1 ns 32b CMOS Adder in multiple output domino logic." in ISSCC Dig. Tech. Papers. pp.140-141, 1988.

[8] I. S. Hwang and P. S. Magarshack, "A high-speed dynamically reconfigurable 32-bit CMOS adder." in Proc. CICC. pp.17.5.1-17.5.6. 1988.

[9] I. S. Hwang and A. L. Fisher, "Ultrafast Compact 32-bit CMOS Adders in Multiple-Output Domino Logic," IEEE Journal of Solid-State Circuits, vol. 24, No. 2, pp.358-368, April 1989.

[10] K. Hwang, Computer Arithmetic : Principles, Architecture, and Design, John Wiley & Sons, 1979.

[11] 김강철, 한석봉, "새로운 동적 CMOS 논리 설계방식을 이용한 고성능 32 비트 가산기 설계", 전자공학회논문지, 제 33 권, A편, 제 3 호, pp. 187-195, 1996년 3월.

[12] N. Weste and K. Eshraghian, Principles of CMOS VLSI Design, Addison-Wesley, 1988.

[13] J. J. F Cavanagh, Digital Computer Arithmetic : Design and Implementation, McGraw-Hill, 1985.



김 강 철 (Kang-Chul Kim)
 1981년 2월 서강대학교 전자공학과 졸업(공학사)
 1983년 2월 서강대학교 대학원 전자공학과 졸업(공학석사)
 1996년 8월 경상대학교 대학원 전자공학과 졸업(공학박사)
 1983년 3월 - 1989년 6월 한국전자통신연구소
 1989년 7월 - 1990년 2월 삼성종합기술원
 1990년 3월 - 1997년 8월 진주산업대학교 전자계산학과
 1997년 9월 - 현재 여수대학교 컴퓨터공학과
 *주관심분야 : VLSI 설계 및 테스트, 전력전자.

이 효 삼 (Hyo-Sang Lee)
 1993년 경상대학교 전자공학과 졸업
 1996년 경상대학교 대학원 전자공학과 졸업
 현재 경상대학교 전자공학과 박사과정
 *주관심분야 : VLSI 설계 및 테스트



송 근 호(Geun-Ho Song)

1995년 2월 동아대학교 전자공
학과 졸업(공학사)
1997년 2월 경상대학교 전자공
학과 졸업(공학석사)
1997년 3월 ~ 경상대학교 대학원
전자공학과 박사과정

*주관심 분야 : VLSI Design, VLSI Testing, Analog
Testing, Design for Testability, IDDQ
Testing, ATPG 등임

서 정 훈(Jeong-Hun Seo)

창원전문대학 전자통신과 전임강사
경상대학교 대학원 전자공학과 박사과정



한 석 봉(Seok-Bung Han)

1982년 2월 한양대학교 전자공
학과 졸업(공학사).
1984년 2월 한양대학교 대학원
전자공학과 졸업(공학
석사)
1988년 2월 한양대학교 대학원

전자공학과 졸업(공학박사)

1988년 3월 경상대학교 전자공학과 전임강사

1992년 1월 ~ 1993년 1월 Stanford University
department of Electrical Engineering and
Computer Science POST DOC.

1993년 4월 ~ 1999년 4월 현재 경상대학교 전자
공학과 교수

*주관심 분야 : VLSI Design(CMOS 및 BiCMOS),
VLSI Testing, Analog Testing, CMOS 칩의
Reliability Testing, Design for Testability, Fault
Tolerance Computing System, VLSI/CAD,
ASIC Design, ATPG 등임