

전류모드 CMOS 4치 논리회로를 이용한 64×64-비트 변형된 Booth 곱셈기 설계

김 정 범[†]

요 약

본 논문에서는 CMOS 다치 논리회로를 이용하여 64×64 비트 Modified Booth 곱셈기를 설계하였다. 설계한 곱셈기는 Radix-4 알고리즘을 이용하여, 전류모드 CMOS 4치 논리회로로 구현하였다. 이 곱셈기는 트랜지스터 수를 기존의 전압모드 2진 논리 곱셈기에 비해 64.4% 감소하였으며, 내부 구조를 규칙적으로 배열하여 확장성을 갖도록 설계하였다. 설계한 회로는 2.5V의 공급전압과 단위전류 5 μ A를 사용하여, 0.25 μ m CMOS 기술을 이용하여 구현하였으며 HSPICE를 사용하여 검증하였다. 시뮬레이션 결과, 2진 논리 곱셈기는 7.5×9.4 mm²의 점유면적에 9.8ns의 최대 전달지연시간과 45.2mW의 평균 전력소모 특성을 갖는 반면, 설계한 곱셈기는 5.2×7.8 mm²의 점유면적에 11.9ns의 최대 전달지연시간과 49.7mW의 평균 전력소모 특성으로 점유면적이 42.5% 감소하였다.

키워드 : 다치논리회로, 전류모드 논리회로, 연산회로, 곱셈기

Design of a 64×64-Bit Modified Booth Multiplier Using Current-Mode CMOS Quarternary Logic Circuits

Jeong Beom Kim[†]

ABSTRACT

This paper proposes a 64×64 Modified Booth multiplier using CMOS multi-valued logic circuits. The multiplier based on the radix-4 algorithm is designed with current mode CMOS quaternary logic circuits. Designed multiplier is reduced the transistor count by 64.4% compared with the voltage mode binary multiplier. The multiplier is designed with Samsung 0.35 μ m standard CMOS process at a 3.3V supply voltage and unit current 5 μ A. The validity and effectiveness are verified through the HSPICE simulation. The voltage mode binary multiplier is achieved the occupied area of 7.5×9.4 mm², the maximum propagation delay time of 9.8ns and the average power consumption of 45.2mW. This multiplier is achieved the maximum propagation delay time of 11.9ns and the average power consumption of 49.7mW. The designed multiplier is reduced the occupied area by 42.5% compared with the voltage mode binary multiplier.

Key Words : Multi-valued logic circuit, Current-Mode logic circuit, Arithmetic circuit, Multiplier

1. 서 론

오늘날 컴퓨터나 전자제품의 핵심은 마이크로프로세서라고 할 수 있다. 마이크로프로세서의 여러 기능 중에서 가장 기본이 되는 기능은 연산 기능이다. 연산기능 중에서 곱셈은 마이크로프로세서가 수행하는 핵심적인 산술연산으로서, 가감산에 비해 소요되는 연산시간이 길고 필요로 하는 트랜지스터의 개수가 많기 때문에 마이크로프로세서 성능에 많

은 영향을 미치게 된다. 또한 곱셈은 마이크로프로세서뿐 아니라 디지털 연산기능을 필요로 하는 다른 응용분야에 기본이 되기 때문에 고성능 곱셈기의 설계기술에 대한 연구가 필요하다. 이를 위하여, 많은 정보량의 처리 문제와 단자수 제한 문제, 단자간의 상호 연결 문제 등을 해결할 수 있는 설계기술이 뒷받침되어야 한다. 이러한 문제를 해결하기 위해 오랫동안 많은 연구가 진행되어 왔는데 여러 해결 방법들 중 하나로 제시된 것이 다치 논리(Multiple-Valued Logic)이다. 다치 논리는 단자간의 상호 연결 복잡도가 감소하고, 트랜지스터 수를 감소시킬 수 있기 때문이다. 또한 2진 논리와 아날로그 신호 처리의 혼합된 형식으로 보다 많은 정보를 처리할 수 있는 아날로그 신호의 장점을 가지고

* 본 연구에 사용한 CAD S/W는 반도체설계교육센터(IDEC)의 자원에 의한 것임.

[†]성희원 : 강원대학교 전기전자공학부 부교수

논문접수 : 2006년 12월 11일, 심사완료 : 2007년 5월 29일

있다. 그러나 다치 논리는 잡음 여유도의 감소와 회로구성의 복잡성 증가로 인해 발생하는 스위칭 속도의 감소 등의 문제를 가지고 있다. 이러한 문제점에도 불구하고 다치 논리에 많은 관심을 가지는 이유는 전체 시스템 측면에서 손실에 비해 성능 개선의 효과가 더욱 크기 때문이다[1-4]. 1983년에 전류모드 CMOS 다치 논리회로가 처음 소개되고 VLSI에 대한 가능성이 입증되면서 이에 대한 관심이 높아졌다. 전류모드 CMOS 다치 논리회로는 전압모드가 갖는 결점을 보완하였고, 낮은 공급 전압에서도 안정적으로 동작하였다. 또한 전류의 가·감산을 활용한 연산동작의 구성이 간편하다는 장점을 가지고 있다. 전류모드 CMOS 다치 논리회로는 디지털 신호처리와 연산회로 분야에서 많은 연구가 진행되어 왔다[1-4],[6-10]. Takahiro Hanyu 등은 이중(dual-rail) 입력과 Sign-Digit 연산을 이용하여 곱셈기를 설계하였다[4]. 현재 VLSI 시스템은 전압모드 2진 논리를 사용하고 있다. 이러한 시스템에 전류모드 다치 논리를 적용하기 위해서 신호간의 호환성이 필수적이다. 전류모드 다치 논리 중에서 4치 논리는 기존의 2진 논리와 변환 및 복원이 용이하여 현재 많은 시스템에 적용되고 있다. 최근에는 기본 CMOS 공정을 이용하여 전류모드 CMOS 다치 논리회로의 고성능화에 대한 연구가 일본을 중심으로 활발히 진행되고 있으며, 연산회로 뿐만 아니라 메모리 회로에 적용을 시도하고 있다.

본 논문에서는 전류모드 CMOS 4치 논리회로를 이용하여 64×64 비트 변형된 Booth 곱셈기를 설계하였다. 먼저 4치 논리에 적용할 수 있도록 변형된 Booth 알고리즘을 변형하였으며, 4치 논리에 적합한 부호확장 제거회로를 사용하여 추가적인 트랜지스터의 사용 없이 2의 보수 연산이 가능하도록 곱셈기를 설계하였다. 본 논문에서는 4치 논리신호의 표현을 위해 각각의 논리레벨을 나타내는 단위전류를 5 μ A로 하여 4치 논리신호의 '00'은 전류레벨로 0 μ A, '01'은 5 μ A, '10'은 10 μ A, '11'은 15 μ A로 정의하였다.

2. 기본회로

2.1 전류모드 CMOS 2진-4치 논리변환 부호기

현재 VLSI 시스템은 전압모드 2진 논리를 사용하고 있다. 전류모드 다치 논리 중에서 4치 논리는 기존의 2진 논리와 변환 및 복원이 용이하여 현재 많은 시스템에 적용되고 있다. 2진 논리 시스템에 다치 논리회로를 적용하기 위해서는 2진 논리를 4치 논리로 변환해 주는 부호기가 사용되는데 이것은 전압모드 2진 논리신호를 전류모드 4치 논리로 변환 해 주는 회로이다[1].

2.2 전류모드 CMOS 4치-2진 논리변환 복호기

전류모드 CMOS 4치-2진 논리변환 복호기는 2진-4치 논리변환 부호기와는 상반되는 개념으로 전류모드 4치 논리신호를 전압모드 2진 논리로 변환하여 주는 회로이다. 복호기는 전류모드 4치 논리회로의 출력 단에 사용되는데 이것은

전류모드의 출력신호를 2진 논리 시스템에서 사용 할 수 있도록 하기 위함이다[1].

2.3 전류모드 CMOS 4치 논리 전가산기

전류모드 CMOS 4치 논리회로는 디지털 신호처리와 연산을 보다 효율적으로 처리할 수 있다. 이것은 4치 논리회로가 2진 논리회로보다 연산회로가 간단하기 때문이다. 전류모드 CMOS 4치 논리 전가산기는 연산회로 중에서 가장 기본적인 회로이기 때문에 이 회로의 특성이 전체 연산에 큰 영향을 미치게 된다. 전류모드 CMOS 4치 논리 전가산기는 하나의 입력단자를 통해 세 개의 입력신호 A와 B 그리고 캐리 입력신호 Ci를 받고 입력신호에 대한 연산 결과를 sum과 carry의 두 단자를 통해 출력한다. 전류모드 CMOS 4치 논리 전가산기는 입력전류가 15 μ A이하까지는 출력 sum은 입력전류 값과 같은 값을 출력한다. 그러나 입력전류가 20 μ A이상이면 carry가 발생하게 된다[1].

3. 4치 논리를 이용한 변형된 Booth 곱셈기 설계

3.1 Booth 곱셈 알고리즘

모든 연산의 가장 기본은 가산과 곱셈이다. 따라서 최적의 가산기와 곱셈기를 사용하면 보다 효율적인 기능을 갖는 연산회로를 설계할 수 있다. 본 논문에서는 Radix-4 알고리즘을 이용한 변형된 Booth 곱셈기를 설계하였다. Booth 곱셈기는 2의 보수 곱셈 연산을 효율적으로 수행할 수 있으며, 일반적인 곱셈기와 비교했을 때 부분 곱의 수를 절반으로 감소시킴으로써 연산 속도와 트랜지스터 수를 감소시키는 장점을 가지고 있다. Booth 부호기와 Booth 선택기의 사용으로 면적이 증가하지만 가산기 부분의 면적이 절반으로 감소하여 칩 전체 면적을 감소시킬 수 있다. 또한 n비트 × n비트에서 n/2(Radix-4) 또는 n/3 (Radix-8) 번의 가산만을 요구하고, x × y에서 x, y 중 하나가 또는 둘 다 2의 보수일 때도 연산이 가능하다. 이 알고리즘은 승수의 인코딩 방법에 따라 Radix-2, Radix-4, Radix-8로 분리할 수 있다. 본 논문에서는 Radix-4 알고리즘을 사용하였다. 이것은 승수에서 나타나는 연속된 k개의 '1'을 변환을 통해서 연산 횟수를 감소시킨다. 이론적으로 이것은 비트의 반을 0으로 하는 방식으로 승수를 다시 쓰는 되는데 이는 특별한 수의 체계를 사용함으로써만이 가능하다. 이것은 부호화된 표준 2's 기수를 십진수들이 집합 { -2, -1, 0, 1, 2 } 안에 존재하는 수 체계로 변환이 된다. 이러한 시스템에서는 어떠한 수도 여러 가지 형태를 쓸 수 있다. 예를 들어

$B = bn-1bn-2\cdots b1b0$ 를 2의 보수 형태로 나타내면 식(1)과 같다.

$$B = -b_{n-1}2^{n-1} + \sum_{i=0}^{n-2} b_i 2^i \quad (1)$$

이 식(1)을 다시 쓰면 식(2)와 같다.

$$B = -b_{n-1} \cdot 2^{n-2} + b_{n-2} \cdot 2^{n-2} + b_{n-3} \cdot 2^{n-3} - b_{n-3} \cdot 2^{n-3} + \dots$$

$$= (-2b_{n-1} + b_{n-2} + b_{n-3})2^{n-2} + (-2b_{n-3} + b_{n-4} + b_{n-5})2^{n-4} + \dots \quad (2)$$

이 식은 $b_1 = 0$ 을 갖는 식으로 변경하면 식(3)으로 다시 쓸 수 있다.

$$B = \sum_{i=0}^{n/2-1} (-2b_{2i+1} + b_{2i} + b_{2i-1})2^{2i} \quad (3)$$

식(3)에서 괄호 속의 항은 $\{-2, -1, 0, 1, 2\}$ 의 집합이다.

〈표 1〉 Radix-4 알고리즘 값에 따른 피승수의 동작

$(-2b_{2i+1} + b_{2i} + b_{2i-1})$	operation
1	load
-1	invert & +1
2	shift left
-2	shift left & invert & +1
0	set to 0

〈표 1〉에 Radix-4 알고리즘의 값에 따른 피승수의 동작을 나타내었다. 표에서 볼 수 있듯이 '1'의 값을 가질 경우에 피승수는 그 값을 그대로 읽고 '-1'인 경우에는 값을 반전시키고 1값을 더하게 된다. '2'의 값을 가질 경우에는 왼쪽으로 이동하여 2배의 값을 갖도록 하며, '-2'인 경우에는 반전시키고 1을 더하여 왼쪽으로 이동한다. '0'인 경우에는 아무런 동작도 하지 않는다. 두 개의 수 A와 B를 연산한다고 가정하자. 우선 Radix-4 알고리즘을 통해 승수 B를 인코딩하면 -2, -1, 0, 1, 2의 숫자를 갖는 B'으로 변환된다. 이러한 형태에서 B의 숫자의 반은 B'에 존재하게 된다. B'의 숫자는 스캔되고 각 단계에서 피승수 A는 -2, -1, 0, 1, 2에 의해서 곱해진다.

3.2 부호확장 제거방법

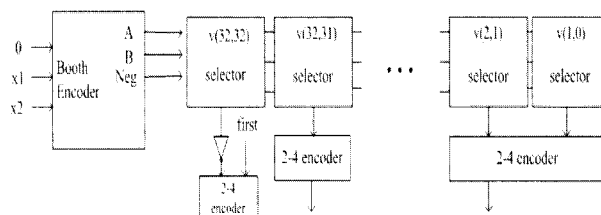
곱셈연산에서는 기본적으로 각 단에서 부분 곱을 생성한 뒤에 다음 단의 부분 곱은 자리를 쉬프트하고 나중에 그 값들을 더하게 된다. (그림 1)에 간단한 예를 제시하였다.

$$\begin{array}{r}
 A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0 \\
 B_7 B_6 B_5 B_4 B_3 B_2 B_1 B_0 \\
 \hline
 C_7 C_6 C_5 C_4 C_3 C_2 C_1 C_0 \\
 D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0 \\
 \hline
 P_{13} P_{12} P_{11} P_{10} P_9 P_8 P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0
 \end{array}$$

(그림 1) 부호 확장 제거 방법의 예

$$\begin{array}{r}
 I \\
 \hline
 I \overline{A_7} A_6 A_5 A_4 A_3 A_2 A_1 A_0 \\
 I \overline{B_7} B_6 B_5 B_4 B_3 B_2 B_1 B_0 \\
 \hline
 I \overline{C_7} C_6 C_5 C_4 C_3 C_2 C_1 C_0 \\
 \hline
 \overline{D_7} D_6 D_5 D_4 D_3 D_2 D_1 D_0 \\
 \hline
 P_{13} P_{12} P_{11} P_{10} P_9 P_8 P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0
 \end{array}$$

(그림 2) 부호 확장 제거 방법



(그림 3) 부호 확장 제거 방법을 이용한 곱셈기의 첫째 단

음영으로 외각선 처리된 부분이 부호 확장된 부분이다. 이처럼 곱셈연산에서는 각 단마다 부호확장된 부분이 생기는데 간단한 연산에서는 문제가 되지 않지만 비트 수가 많아질수록 이 값을 연산하기 위해서는 부가적인 시간과 회로들이 필요하다. 그래서 부호확장된 부분을 제거하여 연산시간을 감소시키고 부가적인 회로를 사용하지 않음으로써 인한 면적을 감소시킬 수 있는 것이 부호확장 제거방법이다.

부호 비트 보수화 방법은 첫 번째 부호비트에 반전된 값을 취하고 확장 비트에 '1'을 더해준다. 이것을 식(4)에 표현하였다. 그리고 첫 번째 단의 부분 곱에서만 최상위 비트에 '1'값을 더해준다.

$$\begin{array}{r}
 s s s s s s s x x x x x x x x x x \\
 + \\
 i i i i i s x x x x x x x x x x \\
 \hline
 i
 \end{array} \quad (4)$$

위의 관계를 정리하여 일반화시키면 (그림 2)와 같다.

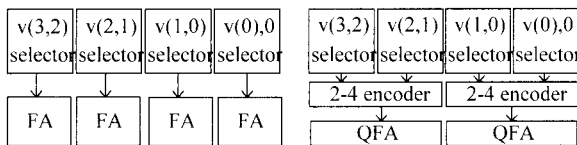
지금까지 설명한 부호확장 제거방법을 전류모드 CMOS 4차 논리회로에 적용한 것을 (그림 3)에 나타내었다. 부호확장 제거기법을 사용한 변형된 Booth 곱셈기의 첫 번째 단은 그림에서 보이는 것처럼 상단에서 내려오는 중간 결과가 없기 때문에 위와 같은 구조로 중간 결과를 만들어 낸다. 최상위 비트의 입력 값을 보면 같은 값이 입력으로 들어가는 데 이것은 부호확장을 위한 값이다. 그 값은 Booth 선택기로 들어가서 어떤 연산을 하는지 결과 값을 출력하게 되고 그 값을 인버터를 통해 반전시킨다. 이것은 앞에서 설명했듯이 부호비트를 반전시키는 역할을 한다. 그리고 첫 번째 단의 최상의 비트에 '1'값을 더해주는 역할은 First라고 표시된 회로가 하게 된다. 그러나 이 값들은 전압모드에서 동작하는 것이 아니라 전류모드에서 동작해야 하기 때문에 전압 값을 전류 값으로 변환시켜주는 회로를 사용하여야 한다.

3.3 변형된 Booth 곱셈기 구조

전류모드 4치 논리 변형된 Booth 곱셈기는 Booth 부호기와 선택기에 의해 만들어진 부분 곱이 전류모드 CMOS 2진-4치 논리변환 부호기에 입력되어 전류 값으로 변하게 된다. 그 값이 실제 연산하는 전류모드 전가산기로 입력되어 sum과 carry를 출력하게 된다. 이것은 기존의 전압모드 2진 논리회로와 비교했을 때 전류 값을 전압 값으로 또는 전압 값을 전류 값으로 변환해 주기 위한 회로가 부가적으로 들어가는 단점이 있다.

그러나 곱셈기를 구현하기 위해 사용된 전가산기의 수가 50% 감소하고, 연산하는 전류모드 전가산기 또한 기존의 전압모드 전가산기에 비해 훨씬 적은 트랜지스터를 사용하기 때문에 전체적으로 사용한 트랜지스터가 감소하는 장점을 가지고 있다. 또한 부분 곱을 한 비트씩 실행하는 2진 논리와는 달리 부분 곱을 두 비트씩 묶어서 연산을 실행하기 때문에 전체 시스템의 상호연결 단자수가 감소하는 장점을 가지고 있다. (그림 4)는 전압모드 2진 논리로 구현할 때의 Booth 곱셈기와 전류모드 4치 논리로 구현할 때의 Booth 곱셈기의 구조를 개략적으로 나타내었다.

본 논문에서 구현한 전류모드 CMOS 4치 논리 곱셈기는 셀 단위로 구성되어 확장성이 용이하도록 설계하였다. 곱셈기의 구조는 우선 승수는 세 비트씩 묶어서 Booth 부호기에 입력한다. 그러나 상위 비트부터 묶으면 마지막 하위 비트는 두 비트밖에 남지 않기 때문에 그 빈자리에는 '0'값을 입력한다. 또한 세 비트씩 묶을 때는 반드시 한 비트씩 겹쳐서 묶어야 한다. 이렇게 Radix-4 알고리즘을 사용함으로써 일반적인 곱셈기와 비교했을 때 전체 열수가 N/2로 감소하기 때문에 면적 및 연산시간이 감소한다. Booth 부호기를 통해 나온 값은 피승수 두 비트와 함께 Booth 선택기의 입력으로 들어가서 부분 곱을 생성한다. Booth 선택기의 출력 값은 전류모드 CMOS 2진-4치 논리변환 부호기의 입력 값으로 들어가서 전류 값으로 변환되며 각 단의 최하위 비트에 사용된 마지막 2진-4치 논리변환 부호기는 전류모드 전가산기의 입력 값으로 들어가서 출력 값으로 나오게 된다. 이때 나오는 값은 전류 값이므로 다시 전압 값으로 변환해주는 전류모드 CMOS 4치-2진 논리변환 복호기를 거치게 된다. 각 단의 최하위 비트에서 나온 2진-4치 논리변환 부호기의 값을 제외한 나머지 값들은 다음 단의 전가산기의 입력 값으로 들어가서 연산하게 된다. 각 단의 최상위 비트는 부호확장 제거방법을 사용해서 한 비트씩 연장한 값이



(그림 4) 변형된 Booth 곱셈기의 배열 구조

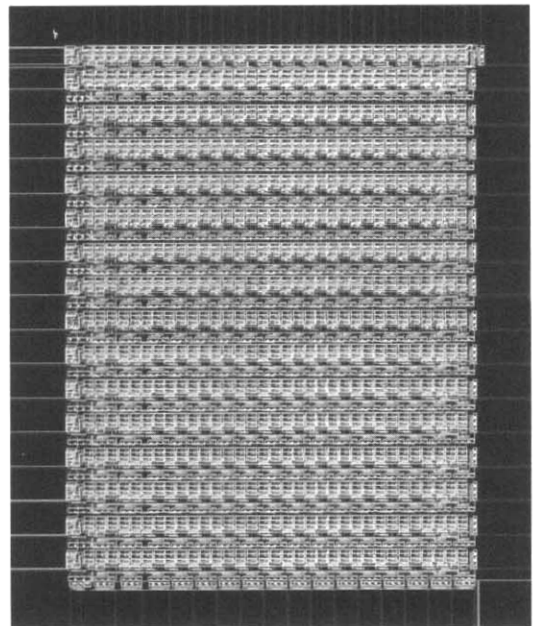
(a)전압모드 2진 논리 (b)전류모드 4치 논리

다. 이 비트 값으로 Booth 선택기에서 나온 값은 인버터를 거쳐 반전되고 그 값이 전류모드 CMOS 2진-4치 논리변환 부호기에 입력되어 연산하게 된다. 마지막 단에서는 전가산기의 sum값이 4치-2진 논리변환 복호기로 입력되어 전압 값으로 출력된다.

본 논문에서 설계한 변형된 Booth 곱셈기는 Radix-4 알고리즘의 장점과 다치 논리의 장점을 이용하였다. 즉 Radix-4 알고리즘의 사용으로 승수를 세 비트씩 묶어서 연산하기 때문에 일반적인 곱셈기보다 N/2의 열로 단수가 감소함으로써 면적에 대한 오버헤드를 감소시켰고 다치 논리의 장점을 이용하여 연결 단자수 문제를 해결하였으며 곱셈기를 구현하기 위해 사용된 트랜지스터의 수도 감소시켰다. 그리고 셀 단위로 회로를 구성하여 확장성이 용이하도록 설계하였고 전류모드와 전압모드의 형태를 변화 시켜주는 부호기와 복호기의 사용으로 인해서 기존의 2진 논리회로와 호환성을 갖도록 하였다.

4. 시뮬레이션 결과 및 비교

본 논문에서는 0.25 μ m CMOS 기술을 이용하여 HSPICE로 시뮬레이션 하였다. 모델 파라미터는 Level 28을 사용하였고 공급전원은 2.5V를 사용하였다. 4치 논리신호의 단위 전류는 5 μ A를 사용하여 각각의 신호 값을 0 μ A, 5 μ A, 10 μ A, 15 μ A로 정의하였다. 출력노드에 흐르는 전류를 측정하기 위해서 출력 단에는 5k Ω 의 저항을 연결하였고 전압 출력노드에는 100fF의 캐패시터를 연결하였다. 설계한 회로는 0.25 μ m 공정기술의 설계규칙에 따라 레이아웃하고 검증을 하였다. (그림 5)는 본 논문에서 설계한 곱셈기의 레이아웃이다.



(그림 5) 곱셈기의 레이아웃

<표 2>는 본 논문에서 설계한 회로와 이전 연구에서 설계한 곱셈기를 비교한 표이다. 전압모드 2진 논리 곱셈기와 비교했을 때 트랜지스터 수는 64.4% 감소하였다. 이로 인해 전체 면적도 2진 논리회로에 비하여 42.5%감소하였다. 설계한 곱셈기 회로의 최대 전달지연은 11.9ns, 평균 전력소모는 49.7mW이다. 반면에, 2진 논리 곱셈기는 최대 전달지연 9.8ns, 평균 전력소모 45.2mW으로 본 논문에서 설계한 곱셈기 보다 우수한 성능을 보이고 있다. 따라서 지금까지의 연구결과를 종합해 보면 트랜지스터 수와 점유면적은 기존 2진 논리 곱셈기에 비해 월등히 우수하지만, 최대 전달지연과 평균 전력소모는 아직 개선해야 할 특징임을 알 수 있다. 또한 4차 논리를 적용함으로써 발생하는 잡음 여유의 감소 문제가 발생한다. 이러한 특성은 앞으로 전류모드 CMOS 다치 논리회로가 반드시 해결해야 할 문제이다.

<표 2> 비교표

	2진 논리 곱셈기[3]	본 논문의 곱셈기
트랜지스터 수	164,506	58,618
점유면적(mm ²)	7.5×9.4	5.2×7.8
평균 전력소모(mW)	45.2	49.7
최대 전달지연(ns)	9.8	11.9

5. 결론

본 논문에서는 전류모드 CMOS 4차 논리회로를 이용하여 64×64 비트 변형된 Booth 곱셈기를 설계하였다. 이 곱셈기는 Radix-4 알고리즘을 이용함으로써 승수를 세 비트씩 묶어서 연산하게 됨으로 N/2열로 단수 및 연산시간을 단축시켰으며 연산과정에서 발생하는 부호비트를 제거함으로써 회로의 면적을 감소시켰다. 본 논문에서 설계된 곱셈기는 기존의 2진 논리 시스템과 호환성을 갖고 있으며 규칙적인 셀 구조를 가짐으로써 확장성을 갖는다. 또한 전류모드 CMOS 4차 논리를 적용하여 상호 연결 노드를 감소시켰으며 트랜지스터 수를 감소시켜 전체적인 면적 감소 효과를 얻을 수 있었다.

참고 문헌

- [1] 이용섭, 광철호, 김정범, “전류모드 다치논리 CMOS 회로를 이용한 전가산기 설계” 전자공학회 논문지, 제39권, SD편, 제1호, pp.76-82, 2002년 1월.
- [2] K. Wayne Current, “Current-Mode CMOS Multiple-Valued Logic Circuits” IEEE J. Solid-State Circuits, vol.29, No.2, pp.95-107, Feb. 1994.
- [3] Takahiro Hanyu and Michitaka Kameyama, “A 200 MHz pipelined multiplier using 1.5 V-supply multiple-valued MOS current-mode circuits with dual rail source-coupled logic” IEEE J. Solid-State Circuits, vol.30, No.11, pp.1239-1245, Nov. 1995.
- [4] Shoji Kawahito, Michitaka Kameyama, Tatsuo Higuchi, and Haruyasu 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.
- [5] Gensuke Goto, Atsuki Inoue, Ryoichi Ohe, Shoichiro Kashiwakura, Shin Mitarai Tsuru, and Tetsuo Izawa, “A 4.1-ns Compact 54 × 54-b Multiplier Utilizing Sign-Select Booth Encoders” IEEE J. Solid-State Circuits, vol.32, No.11, pp.1676-1682, Nov. 1997.
- [6] Chung-Hsun Huang, Jimn-Shyan Wang, and Yan-Chao Huang, “Design of high-performance CMOS priority encoders and incrementor/decrementors using multilevel lookahead and multilevel folding techniques” IEEE J. Solid-State Circuits, vol. 37, No.1, pp.63-76, Jan. 2002.
- [7] A. Morgul and T. Temel, “Current-model level restoration circuit for multi-valued logic”, IEE Electron. Lett., vol. 41, No.5, pp.230-231, Mar. 2005.
- [8] A. Morgul and T. Temel, “Implementation of multi-valued logic gates using full current-mode CMOS circuits” Analog Integr. Circuits Signal Process, vol. 39, No.1, pp.191-204, 2004.

- [9] K. Wayne Current, "Application fo quaternary logic to the design of a proposed discrete cosine transform chip" Int. J. Electronics, vol. 67, No.5, pp.678-701, 1989.
- [10] Masayuki Mizuno, Masakazu Yamashina, Koichiro Furuta, Hiroyuki Igura, Hitoshi Abiko, Kazuhiro Okabe, Atsuki Ono, and Hachiro Yamada "A Ghz MOS Adaptive pipeline Technique Using MOS Current-Mode Logic" IEEE J. Solid-State Circuits, vol.31, No.6, pp.784-791, June 1996.



김 정 범

email : kimjb@kangwon.ac.kr

1985년 인하대학교 전자공학과 학사

1987년 인하대학교 대학원 전자공학과 석사

1997년 포항공과대학교 대학원 전자전기
공학과 박사

1987년~1992년 금성반도체 중앙연구소선임연구원

1994년~1997년 현대전자 시스템IC연구소 책임연구원

1997년~1998년 충북대학교 전기전자공학부

1999년~현 재 강원대학교 전기전자공학부 부교수

관심분야 : VLSI 설계