

논문 2014-51-9-8

# 멤리스터-CMOS 기반의 잉여 이진 가산기 설계

(Design of Redundant Binary Adder based on Memristor-CMOS)

안 연 규\*, 이 상 진\*\*, 김 석 만\*, 캄란 에쉬라기안\*\*, 조 경 록\*\*

(Yeongyu Ahn, Sang-Jin Lee, Seokman Kim, Kamran Eshraghian, and Kyoungrok Cho<sup>©</sup>)

## 요 약

본 논문은 멤리스터-CMOS 기반의 잉여 이진 부호화 자리수 (RBSD) 가산기를 제안한다. 기존의 RBSD 가산기는 리플 캐리 가산기에 비해 큰 면적을 차지한다. 또한 처리하는 비트 수가 적을 때 연산 속도가 느린 단점이 있다. 제안된 RBSD 가산기는 기존 RBSD 가산기의 단점을 보완하기 위해 멤리스터-CMOS 회로를 사용한다. 제안된 멤리스터-CMOS 기반의 RBSD 가산기는 기존 RBSD 가산기에 비해 단위 셀 면적이 45% 감소하였고, 지연시간이 24% 감소하였다. 제안된 멤리스터-CMOS 기반의 RBSD 가산기의 구현으로 인해 RBSD 가산기의 장점이 더욱 부각되고, 대용량 회로에서 더 큰 이득을 얻는다.

## Abstract

This paper presents a memristor-CMOS based RBSD adder. Conventional RBSD adders suffer bigger hardware due to the extra logic handling larger number of bits. The purpose of this paper is to improve the silicon surface area and the computation delay of conventional RBSD adders. The proposed method employs memristor-CMOS based circuit. The implementation results shows that the proposed memristor-CMOS based RBSD adder saves the cell area by 45%, and reduces time delay 24% compared to conventional RBSD adders. The proposed RBSD adder design can bring further area saving for large scale designs.

**Keywords** : 멤리스터, 멤리스터-CMOS, Redundant, Adder, Signed-Digit

## I. 서 론

최근 집적회로와 다양한 고속회로의 급속한 발전으로 인해 고집적 대용량 회로의 연산 처리 속도의 향상과 단위 회로 면적의 감소가 중요한 문제가 되고 있다. 리플 캐리 가산기 (ripple carry adder, RCA)와 같이 일반적인 가산 연산은 캐리 전달 지연 문제 때문에 비트

수가 늘어날수록 지연 시간이 증가하게 된다. 이러한 캐리 전달 지연을 줄이고 빠른 가산 연산을 하기 위해 carry-look-ahead 가산기, carry-skip 가산기, carry-select 가산기 등 다양한 방법들이 연구되고 있다<sup>[1~3]</sup>. 이러한 방식들은 캐리 전달 지연 문제를 해결 하기 위해 추가적인 하드웨어를 사용하여, 처리하는 비트 수에 따른 지연 시간이 리플 캐리 방식 보다 감소하였다. 또 다른 방법으로 캐리 전달 지연 문제를 근본적으로 해결하기 위해 잉여 수 체계가 개발 되었다<sup>[4~6]</sup>.

잉여 이진 수 체계 (redundant binary number system)는 디지털 연산 회로의 캐리 전달 지연 문제를 근본적으로 해결한다<sup>[4]</sup>. 잉여 이진 수 체계는 기존의 이진 수와 달리 가산 연산 시 캐리를 생성하지 않는다. 또한 2의 보수를 쓰지 않고 음수를 표현하기 때문에 고속

\* 학생회원, \*\* 정회원, 충북대학교 전자정보대학 (College of Electrical and Computer Engineering, Chungbuk National University)

<sup>©</sup> Corresponding Author(E-mail: krcho@cbnu.ac.kr)

※ 이 논문은 2013년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었음.

접수일자: 2014년3월24일, 수정일자: 2014년7월30일

수정완료: 2014년9월11일

연산에 유리하다. 잉여 이진 수 체계는 하나의 수를 여러 가지 방법으로 표현할 수 있다. 하나의 이진 수를 표현하기 위해 기존의 수 체계보다 더 많은 비트를 사용한다. 잉여 수 체계 가산기는 처리 하는 비트 수가 적을 때 기존의 가산기에 비해 연산 속도가 느리다. 하지만 잉여 수 체계는 캐리 전달 지연 문제가 발생하지 않기 때문에 연산 속도가 처리하는 비트 수와 관계 없이 일정하다. 잉여 수 체계의 하나인 잉여 부호화 자리수 (redundant binary signed digit, RBSD) 체계는 1961년에 Avizienis에 의해 제안되었다<sup>[7]</sup>. 잉여 이진 부호화 자리수 가산기는 캐리의 전달을 1-비트로 제한한다. 그림 1은 처리하는 비트 수에 따른 리플 캐리 가산기, carry-look-ahead 가산기(CLA), 그리고 잉여 이진 부호화 자리수 가산기의 지연 시간을 비교한 것이다<sup>[8]</sup>. 잉여 수 체계의 가산기는 처리하는 비트 수가 늘어날수록 기존의 다른 방식의 가산기에 비해 큰 이득을 가진다.

RBSD 가산기는 하나의 수를 여러 가지 방법으로 표현하기 위해 추가적인 하드웨어를 가지기 때문에 기존의 방식에 비해 큰 면적을 필요로 하는 단점을 가지고 있다. RBSD 가산기는 비트 수가 늘어날수록 면적 또한 기존의 방식에 비해 증가하는 문제를 가지고 있다. 잉여 수 체계 방식의 가산기는 보다 적은 면적으로 구현하는 것이 중요하다. 본 논문에서는 기존의 RBSD 가산기를 보다 적은 칩 면적으로 구현하기 위해 멤리스터-CMOS 설계 기법을 이용하여 새로운 RBSD 가산기를 제안한다.

본 논문의 구성은 다음과 같다. II장에서는 멤리스터와 멤리스터-CMOS 회로에 대해 설명한다. III장에서는 기존 RBSD 가산기와 제안하는 멤리스터-CMOS 기반의 RBSD 가산기에 대해 설명한다. IVIV장에서 실험결과를 제시하고, 기존 RBSD 가산기와 제안하는 멤리스터-CMOS 기반의 RBSD 가산기를 비교한다. 마지막으로 V장에서 결론을 맺는다.

## II. 멤리스터

멤리스터는 메모리 (memory)와 레지스터 (resistor)의 합성어로 두 개의 터미널을 가진 나노 스케일의 메모리 소자이다<sup>[9-11]</sup>. 멤리스터의 구조는 두 개의 금속 사이에 절연체 역할을 하는 금속 산화물이 접합되어 있는 MIM (metal-insulator-metal) 구조이다. 멤리스터는

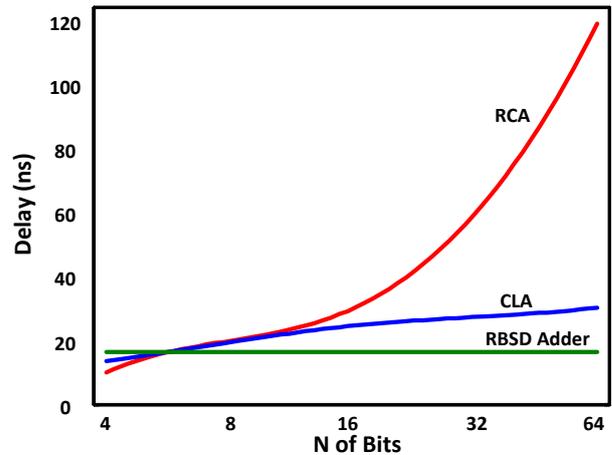


그림 1. 비트수에 따른 지연시간 비교<sup>[8]</sup>  
Fig. 1. Comparison of delay for the input bits.

다양한 금속과 금속 산화물로 구성이 가능하다. 그림 2와 같이 금속 와이어 사이에  $TiO_2$ 와  $TiO_{2-x}$ 의 접합으로 구성된다<sup>[10]</sup>.

멤리스터는 바이폴라 스위칭 특성을 갖는 소자이다. 멤리스터는 바이어스 방향에 따라 산소이온이 이동하여 저항상태가 변하는 특성을 가지고 있다. 이러한 특성을 가진 두 개 이상의 멤리스터를 이용하여 논리 게이트를 구성 할 수 있다<sup>[12]</sup>. 멤리스터가 연결된 중간 노드를 출력으로 사용한다. 출력 노드의 전압은 각 멤리스터의 멤리스턴스 (memristance) 에 따라 결정된다.

멤리스터 AND의 경우 높은 전압이 인가된 쪽의 멤리스터는 높은 저항 상태로 변하게 되고 낮은 전압이 인가된 쪽의 멤리스터는 낮은 저항 상태로 변화한다. 멤리스터 OR의 경우, 높은 전압이 인가된 쪽의 멤리스터는 낮은 저항 상태로 변하게 되고 낮은 전압이 인가된 쪽 멤리스터는 높은 저항 상태로 변화한다. AND와 OR 동작을 하는 멤리스터 게이트를 CMOS와 결합하여 멤리스터-CMOS 회로를 구성할 수 있다<sup>[12]</sup>.

멤리스터-CMOS 기반의 2 입력 NAND 게이트는 두

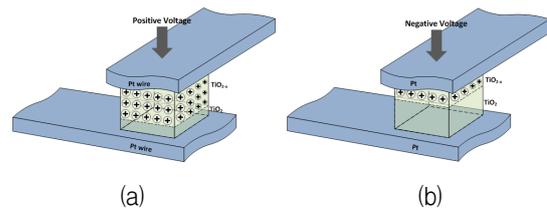


그림 2. 멤리스터의 구조와 특성  
(a) 순방향바이어스 (b) 역방향바이어스  
Fig. 2. Structure and character of memristor.  
(a) Forward bias (b) Reverse bias.

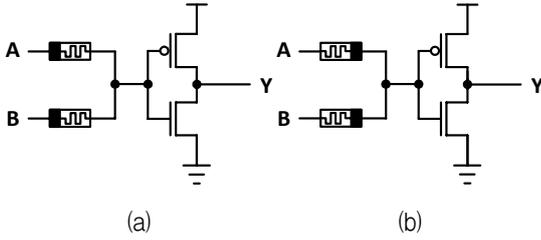


그림 3. 멤리스터-CMOS 논리 게이트<sup>[12]</sup>  
(a) NAND 게이트 (b) NOR 게이트  
Fig. 3. Memristor-CMOS based logic gate.  
(a) NAND gate (b) NOR gate.

입력 모두 '1'인 경우에만 출력이 '0' 된다. 멤리스터-CMOS 기반의 2 입력 NOR 게이트는 두 입력이 모두 '0'인 경우에만 출력이 '1'이 된다. 그림 3은 멤리스터-CMOS NAND 게이트와 멤리스터-CMOS NOR의 구조를 보여준다<sup>[12]</sup>.

Xia<sup>[15]</sup>가 적용한 멤리스터-CMOS 제작 방식을 따르면, 멤리스터-CMOS 회로는 점점을 갖는 CMOS 칩 위에 멤리스터 구성을 위한 메탈 및 TiO<sub>2</sub> 층을 적층하여 만들 수 있다. 이는 기존의 CMOS 단독 칩에 비해 추가되는 공정으로 인해 비용의 상승이 발생한다. 하지만, 멤리스터와 CMOS 칩을 별도 제작한 후 와이어 본딩을 통해 회로를 구성하는 방법에 비해 수율 및 신뢰도가 우수하다. 또한, 이렇게 생성된 멤리스터는 CMOS 회로의 면적을 차지하지 않기 때문에 2 입력 NAND/NOR 게이트의 경우 27%의 면적을 절약할 수 있다<sup>[12]</sup>.

### III. 멤리스터-CMOS 기반의 RBSD 가산기

#### 1. 기존의 RBSD 가산기

RBSD 가산기는 캐리의 전달을 1-비트로 제한한다. RBSD 가산기는 (-1, 0, 1)을 사용하고, Rajshekhar에 의해 제안된 3가지 단계를 거쳐 연산된다<sup>[13]</sup>. RBSD 가산기의 부울식은 다음과 같고, 그림 4는 RBSD 가산연산의 3단계를 보여준다.

$$d_i = m_i \oplus \overline{x_i^+} \overline{x_i^-} \oplus \overline{y_i^+} \overline{y_i^-} \quad (1)$$

$$m_{i+1} = \overline{x_i^+} \overline{y_i^+} \quad (2)$$

$$b_{i+1} = \overline{m_i} \overline{x_i^+} \overline{x_i^-} + \overline{m_i} \overline{y_i^+} \overline{y_i^-} + \overline{m_i} \overline{y_i^+} \overline{y_i^-} + \overline{m_i} \overline{x_i^+} \overline{x_i^-} \overline{y_i^+} \overline{y_i^-} \quad (3)$$

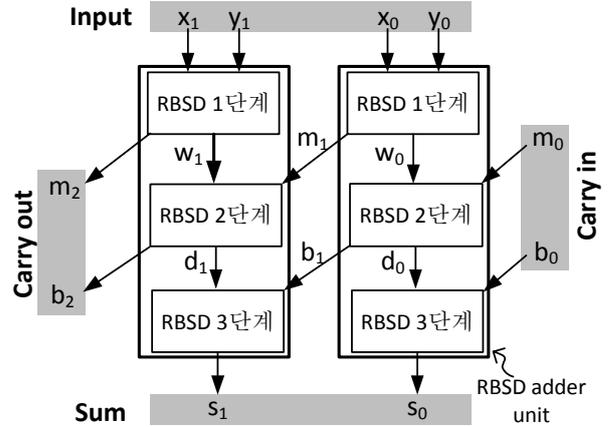


그림 4. RBSD 가산 연산의 3단계  
Fig. 4. 3-steps RBSD addition.

$$s_i^+ = \overline{d_i} b_i \quad (4)$$

$$s_i^- = d_i \overline{b_i} \quad (5)$$

RBSD 가산기의 입력은  $x_i$ 와  $y_i$ 이고,  $s_i$ 는 RBSD 가산기의 최종 합이다. 첫 번째 중간 캐리 값은  $m_i$ 이고, 두 번째 중간 캐리 값은  $b_i$ 이다. 첫 번째 중간 합 값은  $w_i$ 이고, 두 번째 중간 합 값은  $d_i$ 이다. 일반적으로 논리 게이트는 음수(-1)를 사용할 수 없다. 따라서 RBSD 가산기에서 -1, 0, 1을 사용 하기 위해 각 각 (1, 0), (0, 0), (0, 1)로 인코딩하여 사용한다. 회로에서  $x_i$ ,  $y_i$ ,  $s_i$ 는 각각  $(x_i^+, x_i^-)$ ,  $(y_i^+, y_i^-)$ ,  $(s_i^+, s_i^-)$ 로 표현된다. 입력  $x_i$ 에 대해  $(x_i^+, x_i^-)$ 의 값이 (1, 0), (0, 0), (0, 1) 일 때,  $x_i$ 는 각각 -1, 0, 1이 된다. RBSD 가산기는 그림 5와 같이 구성된다<sup>[14]</sup>.

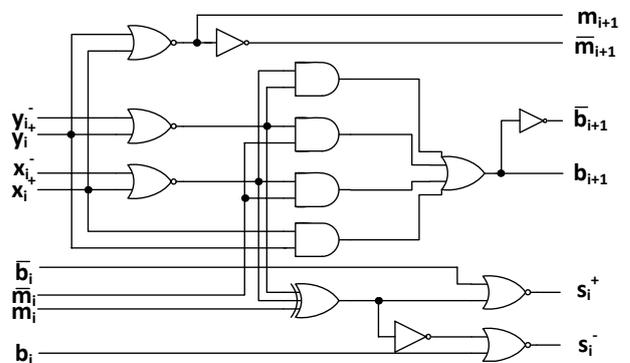


그림 5. RBSD 가산기의 구조  
Fig. 5. Structure of RBSD adder.

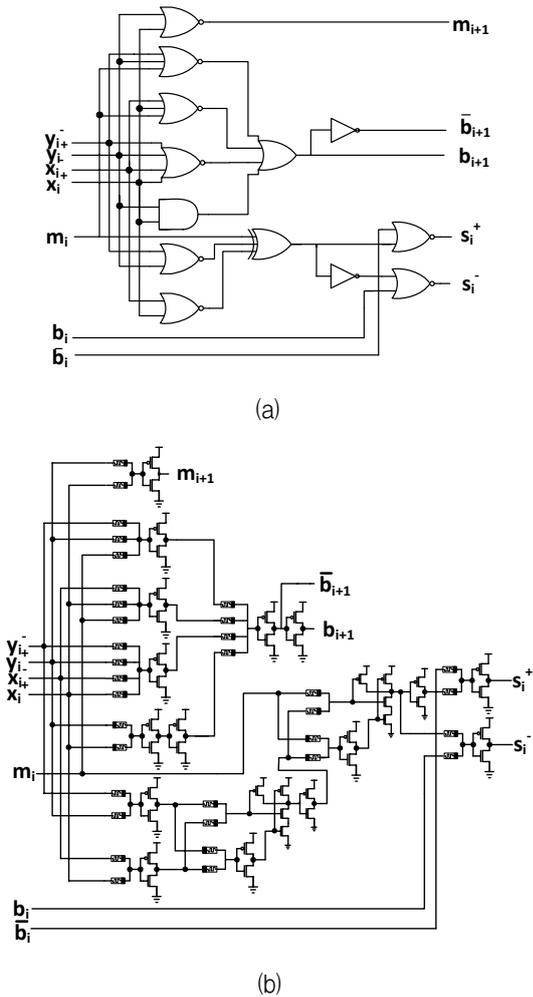


그림 6. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기의 구조 (a) 게이트-레벨 다이어그램 (b) 트랜지스터-레벨 다이어그램

Fig. 6. Structure of proposed memristor-CMOS based RBSD adder (a) Gate-level circuit diagram (b) Transistor-level circuit diagram.

2. 제안하는 RBSD 가산기

기존 RBSD 가산기는 캐리 전달 지연 문제가 발생하지 않기 때문에 처리하는 비트 수가 증가해도 지연시간이 증가하지 않는 장점이 있다. 하지만 기존 RBSD 가산기는 캐리 전달 지연을 갖는 가산기에 비해 면적이 큰 단점이 있다. 따라서 비트 수가 증가하는 만큼 면적이 커지게 된다. 이러한 단점을 보완하기 위해 멤리스터와 CMOS를 이용한 논리 회로로 구성된 멤리스터-CMOS 기반의 RBSD 가산기를 제안한다. 그림 6은 제안하는 멤리스터-CMOS 기반의 RBSD 가산기를 보여준다.

제안하는 멤리스터-CMOS 기반의 RBSD 가산기는

기존의 RBSD 가산기와 같이 3단계 연산을 거쳐 연산된다. 첫 번째 단계에서  $x_i^+$ 와  $y_i^+$ 에 의해 첫 번째 중간 캐리 값인  $m_{i+1}$ 이 연산되고 나머지 입력에 의해 첫 번째 중간 합이 연산된다. 두 번째 단계에서 이전 비트에서 넘어온 첫 번째 중간 캐리 값인  $m_i$ 와 첫 번째 중간 합에 의해 두 번째 중간 캐리 값인  $b_{i+1}$ 과 두 번째 중간 합이 연산된다. 세 번째 단계에서 이전 비트에서 넘어온 두 번째 중간 캐리 값인  $b_i$ 와 두 번째 중간 합에 의해 최종 합이 연산된다. 따라서 병렬연산이 가능하며 캐리 전달 지연 문제가 발생하지 않는다.

제안하는 멤리스터-CMOS 기반의 RBSD 가산기는 기존 RBSD 가산기와 비교하였을 때 게이트의 입력 개수는 증가하였다. 하지만 입력에서 출력까지 신호가 전달되는 구간을 줄였기 때문에 지연시간을 감소시킬 수 있다. 또한 제안하는 RBSD 가산기의 구조를 사용하였을 때 기존 RBSD 가산기에서 입력으로 사용되는  $\bar{m}_i$ 를 사용하지 않고  $m_i$ 만 사용하여 구현할 수 있다. 따라서 기존 RBSD 가산기에 비해 인버터의 수와 신호선이 감소하여 면적 이득을 얻는다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기는 40개의 트랜지스터와 34개의 멤리스터로 구성된다. 92개의 트랜지스터를 사용하는 기존의 RBSD 가산기에 비해 트랜지스터의 수를 52개 줄였기 때문에 면적 면에서 큰 이득을 얻는다.

IV. 실험 결과 및 비교

1. 실험 결과

제안하는 멤리스터-CMOS 기반의 RBSD 가산기의 동작을 Mentor사의 Eldo를 사용하여 시뮬레이션 하였다. 본 논문에서는 멤리스터 해석에 Eshraghian의 SPICE 모델을 사용하여 동작을 검증하였다<sup>[10]</sup>. 시뮬레이션을 통해  $x_i^+, x_i^-, y_i^+, y_i^-, b_i, m_i$  이 입력 되었을 때,  $s_i^+, s_i^-, b_{i+1}, m_{i+1}$ 이 출력 되는 것을 확인 할 수 있다. 그림 7은 제안하는 멤리스터-CMOS 기반의 RBSD 가산기의 시뮬레이션 결과를 보여준다.

제안하는 가산기에 입력  $x_i^+, x_i^-, y_i^+, y_i^-, b_i$ , 그리고  $m_i$ 의 입력이  $\{0, 1, 0, 1, 0, 1\}$  이 인가되었을 때, 출력  $s_i^+, s_i^-, b_{i+1}, m_{i+1}$ 이 각각  $\{0, 1, 0, 1\}$ 이 출력되었다. 이때  $x_i = 1, y_i = 1, carry_i = 1, s_i = 1, carry_{i+1} = 1$  이다. 따라서 제안하는 멤리스터-CMOS 기반의 RBSD 가산기가 기존의 RBSD 가산기와 같이 가산 연산 동작을

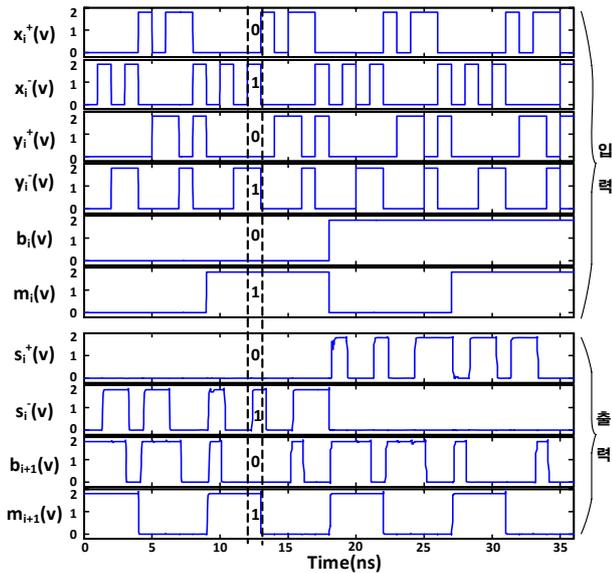


그림 7. 멤리스터-CMOS 기반의 RBSD 가산기 시뮬레이션 결과

Fig. 7. Simulation result of memristor-CMOS based RBSD adder.

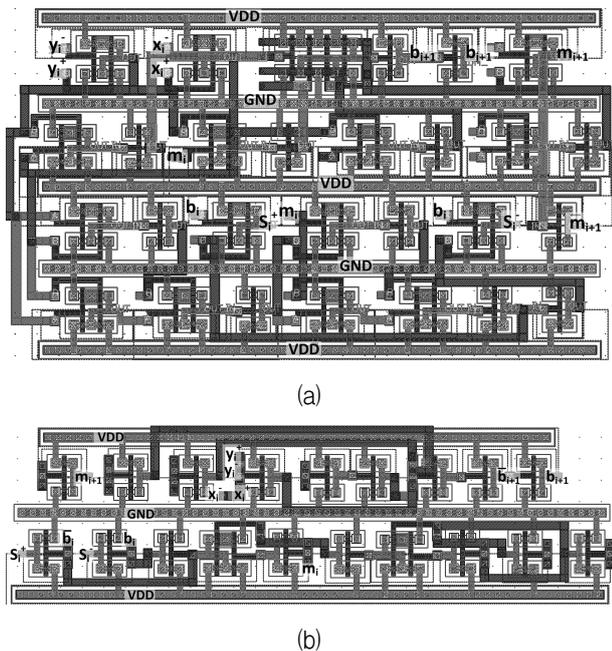


그림 8. 0.18  $\mu\text{m}$  공정의 1-비트 RBSD 가산기 레이아웃 (a) 기존 RBSD 가산기 (b) 멤리스터-CMOS 기반의 RBSD 가산기

Fig. 8. Layout of 1-bit RBSD adder for 0.18  $\mu\text{m}$  CMOS technology (a) Conventional RBSD adder (b) Memristor-CMOS based RBSD adder.

하는 것을 확인 할 수 있다.

기존의 RBSD 가산기는 92개의 트랜지스터를 사용하여 구현된다. 제안하는 멤리스터-CMOS 기반의 RBSD

가산기는 40개의 트랜지스터와 34개의 멤리스터로 구현된다. CMOS 회로를 먼저 제작한 후 CMOS 웨이퍼 위에 멤리스터 소자를 만든다. 멤리스터는 메탈-3 레이어를 통하여 CMOS 회로와 연결된다. 그림 8은 0.18  $\mu\text{m}$  CMOS 공정을 이용한 기존의 1-비트 RBSD 가산기와 제안하는 멤리스터-CMOS 기반의 1-비트 RBSD 가산기의 레이아웃이다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기의 단위 셀의 면적은  $184\mu\text{m}^2$ 이다. 기존 CMOS 기반의 RBSD 가산기의 단위 셀의 면적은  $335\mu\text{m}^2$ 이다. 따라서 기존의 RBSD 가산기에 비해 제안하는 RBSD 가산기의 면적이 약 45% 감소하였다.

## 2. 성능 비교

기존의 1-비트 RBSD 가산기와 제안하는 멤리스터-CMOS 기반의 1-비트 RBSD 가산기를 구조와 면적, 지연시간, 전력소모에 관하여 비교하였다. 구조와 면적은 앞에서 언급하였으며, 0.18  $\mu\text{m}$  CMOS 공정을 사용하여 레이아웃하였다. 지연시간과 전력소모는 SPICE 틀인 Eldo를 이용하여 동일한 공정의 모델로 시뮬레이션 하였다. 표 1은 기존 RBSD 가산기와 제안하는 RBSD 가산기의 성능을 비교한 결과이다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기는 기존 RBSD 가산기에 비해 45%의 면적과 24%의 지연시간이 크게 감소하였다. 전력소모는 약 6% 증가하였지만 면적과 지연시간의 이득에 비해 크게 증가하지 않았다. 따라서 RBSD 가산기를 구성하였을 때 제안하는 멤리스터-CMOS RBSD 가산기가 더 효율적이다.

리플 캐리 가산기와 기존의 RBSD 가산기 그리고 제

표 1. 0.18  $\mu\text{m}$  공정의 RBSD 가산기와 제안하는 RBSD 가산기 비교

Table 1. Comparison of RBSD adder and proposed RBSD adder for 0.18  $\mu\text{m}$  CMOS technology.

	[13]	This work	개선율 (%)
구조	92 Tr.	40 Tr. 34 mem.	-
면적 ( $\mu\text{m}^2$ )	335	184	45%
지연시간 (ps)	326	248	24%
전력소모 ( $\mu\text{W}$ )	32.1	34.2	-6%

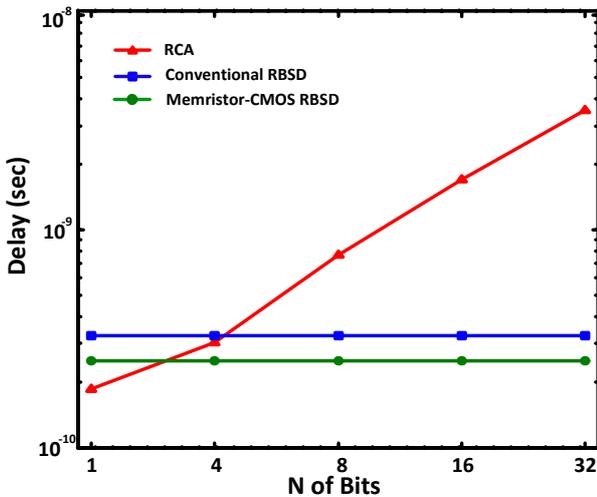


그림 9. 비트 수에 따른 지연 시간 비교

Fig. 9. Comparison of delay for the input bits.

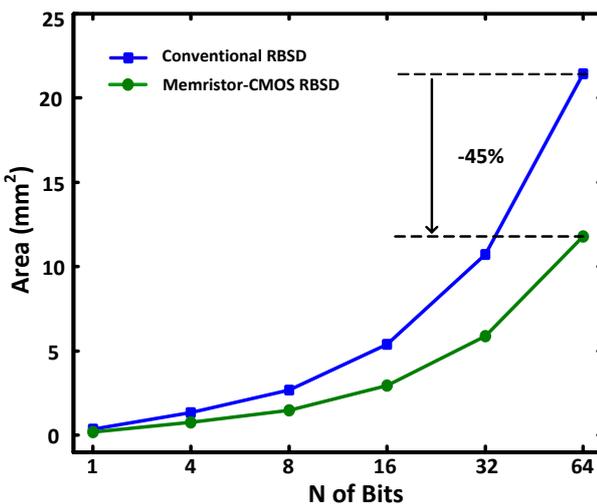


그림 10. 비트 수에 따른 면적 비교

Fig. 10. Comparison of area occupation for the input bits.

안하는 멤리스터-CMOS 기반의 RBSD 가산기의 지연 시간을 처리하는 비트 수에 따라 비교한 결과를 그림 9에 나타내었다. 리플 캐리 가산기의 경우 비트 수가 적을 때는 RBSD 가산기에 비해 지연시간이 짧지만, 비트 수가 증가할수록 캐리 전달 지연 문제가 발생하게 되어 지연시간이 길어지게 된다. RBSD 가산기는 캐리가 전달되지 않기 때문에 처리하는 비트 수가 증가하여도 지연시간이 증가하지 않는 장점을 갖는다. 멤리스터-CMOS 기반의 RBSD 가산기 또한 비트 수가 증가해도 지연시간이 늘어나지 않는 장점을 가지는 것을 확인할 수 있다. 그림 10은 처리하는 비트 수에 따른 기존의

RBSD 가산기와 제안하는 RBSD 가산기의 면적을 비교한 것이다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기는 기존 RBSD 가산기에 비해 비트 수가 증가할수록 더 큰 면적 이득을 가지게 된다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기를 0.18  $\mu\text{m}$  공정에서 64-비트로 구현하였을 때 기존의 RBSD 가산기에 비해서 면적이 약 45% 개선되었다.

## V. 결 론

본 논문에서는 멤리스터-CMOS 회로 설계 기법을 이용하여 RBSD 가산기를 설계하였다. 제안된 멤리스터-CMOS 기반의 RBSD 가산기는 캐리 전달 지연 문제가 발생하지 않기 때문에 비트 수가 증가하여도 지연시간이 증가하지 않는다. 이러한 기존 RBSD 가산기의 장점을 유지하면서, 기존 RBSD 가산기의 단점인 면적과 지연시간을 보완하였다. 제안된 멤리스터-CMOS 기반의 RBSD 가산기는 기존 RBSD 가산기에 비해 게이트의 입력 개수는 증가하였지만 입력에서 출력까지 신호가 전달되는 구간을 줄였다. 그 결과 기존의 가산기에 비해 제안하는 가산기의 지연시간이 24% 감소하였다. 기존의 RBSD 가산기는 92개의 트랜지스터로 구성된다. 제안하는 멤리스터-CMOS 기반의 RBSD 가산기는 40개의 트랜지스터와 34개의 멤리스터로 구현 가능하여 52개의 트랜지스터가 감소하였다. 레이아웃 결과 기존 RBSD 가산기에 비해 제안하는 RBSD 가산기의 면적이 45% 감소하였다. 멤리스터-CMOS 설계 기법을 이용하여 지연 시간과 면적을 줄여 고집적 회로에서의 RBSD 가산기의 장점이 더욱 부각될 것이다.

## REFERENCES

- [1] H. Ling, "High-speed binary adder," *IBM Journal of Research and Development*, Vol. 25, no. 3, pp. 156-166, March 1981.
- [2] O. J. Bedriji, "Carry-select adder," *IRE Transactions on Electronic Computers*, Vol. 11, no. 3, pp. 340-346, June 1962.
- [3] V. Kantabutra, "Recursive carry-lookahead / carry-select hybrid adder," *IEEE Transactions on Computers*, Vol. 42, no. 12, pp. 1495-1499, December 1993.
- [4] G. Metzke, and J. E. Robertson, "Elimination of

- carry propagation in digital computers,” *IFIP Congress*, pp. 389–395, Paris, France, June 1959.
- [5] A. F. Gonzalez, and P. Mazumder, “Redundant arithmetic, algorithms and implementations,” *Integration, the VLSI Journal*, Vol. 30, no. 1, pp. 13–53, November 2000.
- [6] H. R. Srinivas, and K. K. Parhi, “Computer arithmetic architectures with redundant number systems,” *1994 Conference Record of the Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*, Vol. 1, pp. 182–186, November 1994.
- [7] A. Avizienis, “Signed-digit number representations for fast parallel arithmetic,” *IRE Transactions on Electronic Computers*, Vol. EC-10, no. 3, pp. 389–400, September 1961.
- [8] A. Kumar, N. Sharma, and A. K. Wadhvani, “Fast adder design using redundant binary numbers with reduced chip complexity,” *IACSIT International Journal of Engineering and Technology*, Vol. 3, no. 3, pp. 274–278, June 2011.
- [9] L. O. Chua, “Memristor—the missing circuit element,” *IEEE Transactions on Circuit Theory*, Vol. 18, no. 5, pp. 507–519, September 1971.
- [10] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, “The missing memristor found,” *Nature*, Vol. 453, pp. 80–83, May 2008.
- [11] K. Eshraghian, O. Kavehei, K. R. Cho, J. M. Chappell, A. Iqbal, S. F. Al-Sarawi, and D. Abbott “Memristive device fundamentals and modeling: applications to circuits and systems simulation,” *Proceedings of the IEEE*, Vol. 100, no. 6, pp. 1991–2007, June 2012.
- [12] Ca-Ram Han, Sang-Jin Lee, K. Eshraghan, Kyoungrok Cho, “Primitive IPs Design Based on a Memristor-CMOS Circuit Technology,” *Journal of the Institute of Electronics Engineers of Korea*, Vol. 50, No. 4, pp. 65–72, April 2013.
- [13] T. N. Rajashekhar, and O. Kal, “Fast multiplier design using redundant signed-digit numbers,” *International Journal of Electronics Theoretical and Experimental*, Vol. 69, no. 3, pp. 359–368, 1990.
- [14] N. Sharma, B. S. Raj, and A. Kumar, “Design of RBSD adder and multiplier circuits for high speed arithmetic operations and their timing analysis,” *Special Russian Issue: Advances in Computer Science and Engineering, Research in Computing Science23*, pp. 243–254, 2006.
- [15] Q. Xia, et al., “Memristor—CMOS hybrid integrated circuits for reconfigurable logic,” *Nano letters*, Vol. 9, No. 10, pp. 3640–3645, 2009.

— 저 자 소 개 —



안 연 규(학생회원)  
 2012년 충북대학교 정보통신공학과 학사 졸업.  
 2014년 충북대학교 정보통신공학과 석사 졸업.  
 <주관심분야:멤리스터-CMOS >



이 상 진(정회원)  
 2008년 충북대학교 화학공학과 학사 졸업.  
 2010년 충북대학교 정보통신공학과 석사 졸업.  
 2014년 충북대학교 정보통신공학과 박사 졸업.

2014년~현재 (재)스마트 IT 융합 시스템 연구단 연구조교수  
 <주관심분야 : CMOS image sensor, 멤리스터-CMOS 회로설계>



김 석 만(학생회원)  
 2005년 충북대학교 전기전자컴퓨터공학부 학사 졸업.  
 2008년 충북대학교 정보통신공학과 석사 졸업.  
 2008년~현재 충북대학교 정보통신공학과 박사과정.

<주관심분야 : SoC 설계, 저전력 회로 설계>



Kamran Eshraghian(정회원)  
 1968년 B.Tech, Electrical & Electronic Eng., University of Adelaide.  
 1978년 MEng. Sc., Electrical & Electronic Eng., University of Adelaide.

1980년 Ph.D., Electrical & Electronic Eng., University of Adelaide.  
 1969년~1977년 Philips 선임연구원.  
 1979년~1994년 University of Adelaide 교수.  
 1994년~2004년 Edith Cowan University 교수.  
 2007년~2009년 University of California Merced 객원교수.  
 2009년~현재 충북대학교 전자정보대학 석좌교수  
 <주관심분야 : CMOS VLSI 설계, system on system (SoS) multi-technology 회로설계>



조 경 록(정회원)  
 1977년 경북대학교 전자공학과 학사 졸업.  
 1989년 일본 동경대학교 전자공학과 석사 졸업.  
 1992년 일본 동경대학교 전자공학과 박사 졸업.

1979년~1986년 (주)금성사TV연구소 선임연구원.  
 1999년, 2005년 Oregon State University 객원교수.  
 1992년~현재 충북대학교 전자정보대학 교수.  
 2008년~2011년 World Class University program (충북대학교) 책임.  
 2010년~현재 IDEC 충북대지역센터장  
 <주관심분야 : 통신시스템 LSI 설계, 저전력 고속 회로설계, Platform 기반의 SoC 설계>