

論文94-31B-5-9

## 배열 승산기의 속도 향상을 위한 CSA 배열 구조에 관한 연구

## (A Study on the Design Scheme of CSA Array for the Speed up of Array Multiplier)

李庸奭\*, 洪謹善\*, 金容得\*

(Yong Seok Lee, Geun Seon Hong and Yong Deak Kim)

## 要約

본 논문에서는 기존의 배열 연산기의 CSA(carry save adder) 배열 구조를 수정하여 그 연산시간을 단축할 수 있는 새로운 구조를 제안하며 이러한 구조에 의한 배열 승산기를 설계하고 설계된 회로의 동작을 시뮬레이션 하였다. 제안된 방식은 기존의 CSA의 구조를 가수 및 피가수가 인가되는 논리배타합의 입력을 자리올림 입력보다 먼저 인가되도록 수정하여 CSA 배열의 연산시간을 약 50% 단축할 수 있음을 확인하였다.

## Abstract

This paper deals with a new design scheme to reduce the array multiplication time by modifying the structure of the CSA array in the conventional array multiplier. A circuit with the suggested scheme is designed and simulated. The suggested scheme is to assign the inputs of addend and augend faster than carry input to the CSA. It is shown that the operation time of the CSA array is reduced to 50%.

## 1. 서론

고속 승산기에 대한 연구는 1950년대 컴퓨터가 사용되기 시작한 이래 꾸준히 진행되어 왔다. 그중 대표적인 승산방식으로는 1951년 Booth가 제안한 Booth algorithm<sup>[2]</sup>, 1964년 Wallace가 제안한 Wallace tree<sup>[3]</sup>, Dadda가 제안한 카운터를 사용한 승산항의 압축기법<sup>[4]</sup> 및 배열 승산기<sup>[5][6]</sup>를 들 수 있다. 배열 승산 방식은 승산기의 구성에 소요되는 논리 소자의 양은 많으나 그 구조가 단순하고 규칙적 이므로 VLSI 구현에 적합한 특징을 가지고 있어서

디지털 프로세서에 사용되는 고속 승산방식으로 많이 사용되고 있으며, 최근에는 승산시 발생하는 승산항의 수를 줄일 수 있는 수정된 부스 알고리즘<sup>[7]</sup>을 기반으로 하는 복수 비트 검색기법을 적용한 고속 배열 승산기에 대한 연구<sup>[8][9]</sup>가 활발히 진행되고 있다.

복수비트 검색 방식은 승산항의 수를 25% 이상 줄일 수 있으나 추가되는 회로의 양이 많다는 단점을 갖는다.

본 논문에서는 배열 승산기의 기본 소자로 사용되는 1비트 전 가산기인 CSA(carry save adder)의 연산특성을 이용하여 CSA의 구조 및 CSA 배열의 연결을 수정하여 CSA 배열의 연산시간을 약 50% 줄일 수 있는 배열 승산기의 구조를 제안하고 성능을 평가한다. 제안된 방식에 의하여 설계된 배열 승산기의 동작은 시뮬레이션 소프트웨어인 WORKVIEW

\* 正會員, 亞洲大學校 電子工學科  
(Dept. of Elec. Eng., Ajou Univ.)  
接受日字 : 1993年 7月 23日

를 사용하여 확인 하였다.

II . 배열 승산기의 구조

배열 승산기는 1비트 전 가산기인 CSA를 기본 소자로 하여 승산시 발생하는 승산항을 더한 후 최종단에서 발생하는 자리올림 및 가산합을 자리올림 전파형 가산기(carry propagation adder)로 더하는 구조를 갖는다. 5비트 승산시 발생하는 승산항은 그림 1과 같고 이를 연산하기 위한 승산기의 구조는 그림 2와 같다.

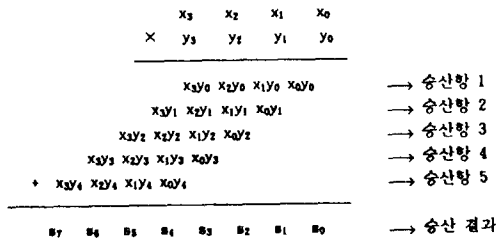


그림 1. 4 비트 승산에서의 승산항의 구성  
Fig. 1. Product terms of 4 bit multiplication.

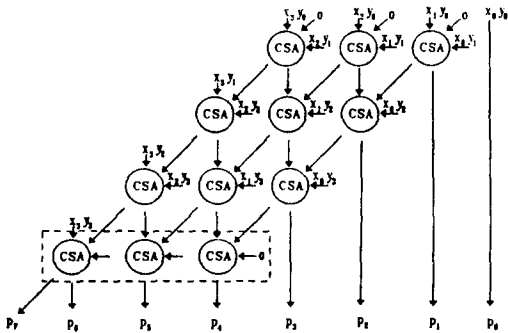


그림 2. 부호 없는 4 비트 수의 승산을 위한 배열승산기의 구성도  
Fig 2. The schematic circuit diagram of 4 bit unsigned array multiplier.

그림 2에서 점선 부분의 CSA들은 상위 CSA열에서 연산된 자리올림 및 가산합을 더하여 최종 승산합을 발생시키기 위한 자리올림 전파형 가산기로 고속 승산기를 구성하는 경우에는 이 부분에 자리올림 예측형 가산기나 자리올림 선택형 가산기등의 고속 가산기를 사용하여 승산시간을 단축한다.

III . 속도 향상을 위한 CSA 구조의 제안

배열 승산기의 기본 소자로 사용되는 1비트 전 가산기의 자리올림에 대한 출력특성은 표 1과 같이 가수 및 피가수의 조합이 1,0 또는 0,1인 경우에는 자리올림 입력 값을 그대로 자리올림 출력 값으로 하고 가수 및 피가수의 조합이 0,0인 경우에는 자리올림 입력값에 관계없이 자리올림 출력값을 0으로 출력하며 가수 및 피가수의 조합이 1,1인 경우에는 자리올림 입력값에 관계없이 자리올림 출력값을 1로 출력한다. 가산합에 대한 출력 특성은 가수 및 피가수의 논리 배타합을 자리올림 입력이 0인 경우에는 그대로 출력하고 자리올림 입력이 1인 경우에는 반전시켜 출력하며 이러한 특성은 표 2와 같다.

표 1. 가수 및 피가수의 조합에 의한 자리올림의 출력특성

Table 1. Carry propagation characteristics of a full adder.

| 대용 비트의 조합 |     | 자리올림에 대한 특성          |
|-----------|-----|----------------------|
| 가수        | 피가수 |                      |
| 0         | 0   | 자리올림이 발생된다(자리올림 = 0) |
| 0         | 1   | 앞자리에서의 자리올림을 전파시킴    |
| 1         | 0   | 앞자리에서의 자리올림을 전파시킴    |
| 1         | 1   | 자리올림이 발생된다(자리올림 = 1) |

표 2. 자리올림 입력에 의한 가산합의 출력 특성

Table 2. Sum generation characteristics of a full adder.

| 자리올림 입력값 | 임시 가산합            | 가산합 출력값             |
|----------|-------------------|---------------------|
| 0        | $ST = A \oplus B$ | $S = ST$            |
| 1        |                   | $S = \overline{ST}$ |

표 1 및 표 2의 가산 특성을 이용한 1비트 전 가산기는 가수 및 피가수 비트의 임시 가산합 발생을 위한 논리 배타합 소자와 자리올림 입력에 의한 가산합 출력 선택을 위한 2입력 멀티플렉서 그리고 가수 및 피가수 비트의 조합에 의하여 결정되는 자리올림의 전파 또는 발생에 의한 자리올림 출력 값 선택을 위한 2입력 멀티플렉서로 구성할 수 있으며 그 구조는 그림 3과 같다.

그림 3의 1비트 전 가산기는 그림 2의 배열 승산기에서 승산 합 가산을 위한 CSA로 사용되며 3개의 입력중 A는 가수, B는 피가수 그리고 CIN은 자리올림

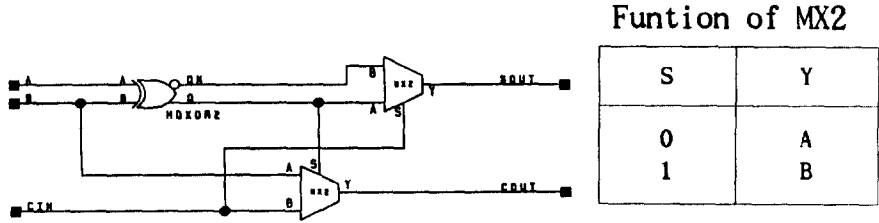


그림 3. 멀티플렉서를 이용한 전가산기 회로도  
Fig. 3. A full adder using a multiplexer.

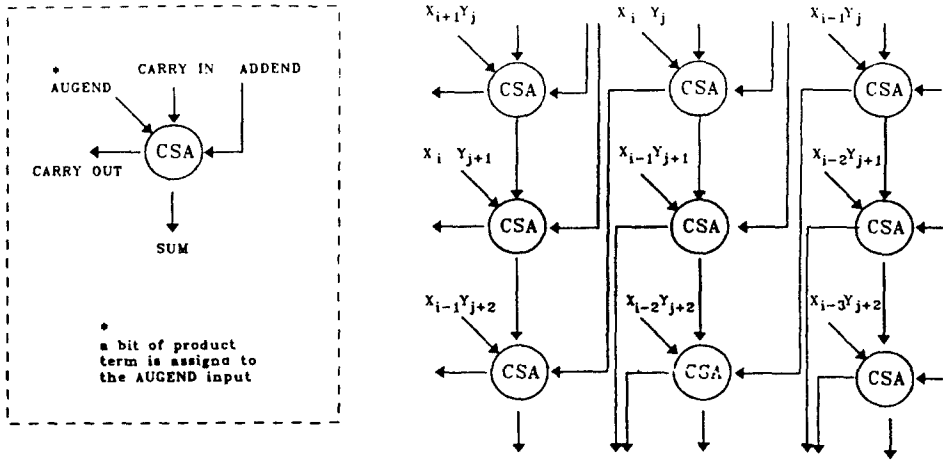


그림 4. 배열 승산기의 CSA열의 연산시간 단축을 위한 구조  
Fig. 4. Time reduction schemes of CSA array.

입력이다. 가산 시간을 논리적으로 설명하기 위하여 1비트 전 가산기에 사용되는 멀티플렉서 및 논리 배타합 소자의 동작 지연을 각각  $14t$ 라고 하자. 이 경우 그림 3의 1비트 전 가산기의 동작지연은  $24t$ 이다. 1비트 전 가산기를 배열 승산기의 CSA 배열에 사용하는 경우 그림 3의 B 단자에는 그림 1에 나타난 것과 같은 승수 및 피승수 비트의 논리 곱으로 발생하는 승산항이 인가되어 있으며 나머지 2개의 입력인 A와 CIN은 전단의 CSA 열에서 발생한 자리올림 출력 및 가산합이 인가된다.

따라서 그림 3의 1비트 전 가산기를 그림 2의 CSA로 사용하는 경우에는 각 CSA의 연산시간은  $24t$ 이다. 그러나 만약 CSA의 3개의 입력중 A의 입력값이 CIN의 입력값보다  $14t$  만큼 먼저 인가된다면 A, B를 입력으로 하는 논리 배타합의 출력값이 자리올림 입력인 CIN이 인가되는 순간에 출력되므로 각 CSA 열의 연산시간은  $14t$ 가 된다. 이러한 원리에 의한 CSA 배열의 구성은 그림 4와 같다.

그림 4에서  $i$ 번째 CSA 열의 가산합 SOUT은  $i+1$ 번째 CSA 열의 자리올림 입력인 CIN으로 인가되고

$i$ 번째 CSA 열의 자리올림 출력인 COUT는  $i+2$ 번째 CSA 열의 가수 입력인 A로 인가된다.

따라서 각 CSA는 자리올림 입력보다 먼저 B와 A 입력에 값이 인가되는 구조를 갖는다.  $n$ 비트 승산시 이러한 원리로 연결되는 CSA 배열의  $n-1$ 번째 CSA 열 다음에는  $n-1$ 번째 CSA 열에서 발생하는 자리올림 출력과 가산합 그리고  $n-2$ 번째 CSA 열에서 발생한 자리올림 출력이 발생되므로 이의 가산을 위하여 1개의 CSA열이 추가 되므로 기존 방식보다 CSA열의 수가 1개 많게 된다.

#### IV. 고속 배열 승산기의 설계

본 논문에서 제안하는 고속 배열 승산방식은 배열 승산기의 기본 구성 요소인 CSA의 구조를 변형하여 각 CSA의 동작속도를 향상 시키고 이를 사용한 CSA 배열의 배선을 수정하여 CSA 배열의 연산시간을 단축하는 것이다.

CSA 자체의 동작속도를 향상 시키는 방법은 그림 3에서 사용된 비반전형 멀티플렉서를 반전형 멀티플

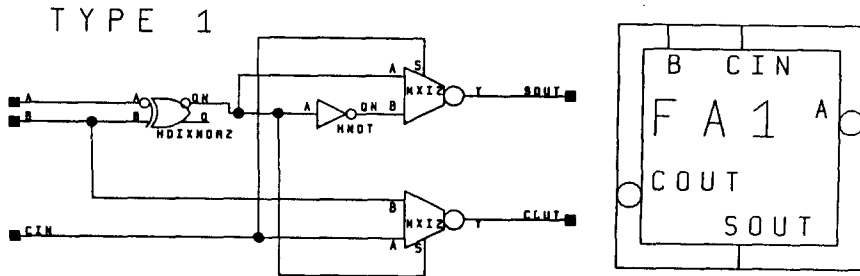


그림 5. 1개의 반전 입력 및 반전 자리올림 출력을 발생시키는 CSA(FA1)  
 Fig. 5. A CSA having one inverted input and an inverted carry out(FA1).

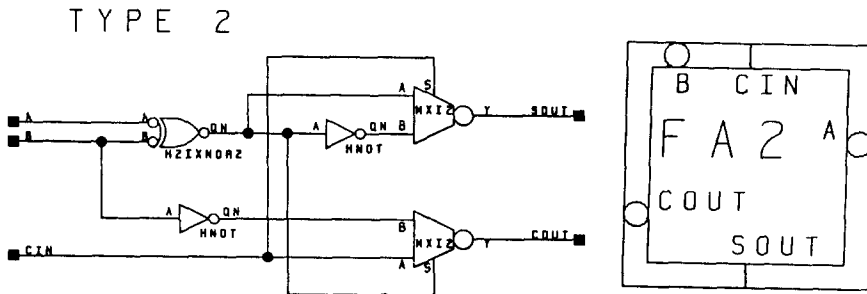


그림 6. 2개의 반전 입력 및 1개의 반전 출력을 갖는 CSA(FA2)  
 Fig. 6. A CSA having two inverted input and an inverted output(FA2).

렉서로 대체하는 것이다. 현재 대부분의 표준 CMOS 논리소자로 구성되는 2입력 비 반전형 멀티플렉서의 동작 지연이 1.0ns인데 비하여 2입력 반전형 멀티플렉서의 동작 지연은 0.4ns<sup>[10]</sup>이다. 그러나 비반전형 멀티플렉서를 반전형 멀티플렉서로 대체하는 경우, 자리올림 출력이 반전되게 되므로 이와 연결되어 있는 CSA의 입력은 이를 반전시켜 처리하여야 한다. 본 논문에서는 CSA 소자의 성능 향상을 위하여 2입력 비 반전형 멀티플렉서를 반전형으로 대체한 구조의 2종류의 CSA인 FA1 및 FA2를 사용한다. FA1은 각 승산항을 가산하기 위한 1개의 반전된 입력 및 1개의 반전된 출력을 갖는 CSA이며 FA2는 승산항 가산을 위한 CSA 배열의 최종열에서 발생하는 2개의 반전된 출력을 가산하기 위한 CSA이며 그 구조는 각각 그림 5 및 그림 6과 같다.

이러한 2종류의 CSA 소자를 사용하여 본 논문에서 제안한 방식에 의한 4비트 배열 승산기를 구성한 예가 그림 7 및 그림 8이다. 4비트 승산시 인가되는 피승수 X라 하고 그 비트 요소를 X4, X3, X2, X1, X0라 하고 승수를 Y라 하고 그 비트 요소를 Y4, Y3, Y2, Y1, Y0라 하자. 승산시 발생하는 승산항은 그림 7과 같이 승수 및 피승수 각 비트의 논

리곱으로 얻을 수 있다. 그림 8은 발생한 승산항을 더하기 위한 수정된 CSA 배열의 구조이다. 승산항을 더하는 CSA인 FA1은 반전된 승수 입력과 반전된 자리올림 출력을 갖는다. 이는 CSA의 동작시간을 줄이기 위하여 자리올림 출력을 위한 2입력 멀티플렉서를 반전형으로 사용하였기 때문이다. 각 CSA의 가산합 출력은 다음 CSA 열의 자리올림 입력으로 인가되고 자리올림 출력은 2번째 아래의 가수 입력인 A로 인가된다. 이러한 구성에서는 각 CSA의 가수 및 피가수 입력이 자리올림 입력보다 먼저 인가되어 가수 및 피가수의 논리 배타합을 미리 연산하고 이전 CSA열에서 출력되는 가산합이 자리올림 입력으로 인가되었을 때 2입력 반전형 멀티플렉서의 동작 지연 후 가산합 및 자리올림 출력이 발생되도록 하기 위한 것이다. 따라서 각 CSA열의 동작시간은 논리 배타합 소자와 반전형 멀티플렉서 중에서 동작 지연시간이 큰 소자의 동작 시간과 같다.

그림 8의 FA1 CSA 열의 최종단에서는 제안된 구조의 특성으로 발생하는 2개의 반전된 자리올림 출력 및 1개의 가산합이 발생되게 된다. 이 3개의 값을 CSA를 이용하여 2개의 값으로 줄여 이를 자리올림 전과정 가산기로 가산하기 위하여 1개의 CSA 열이

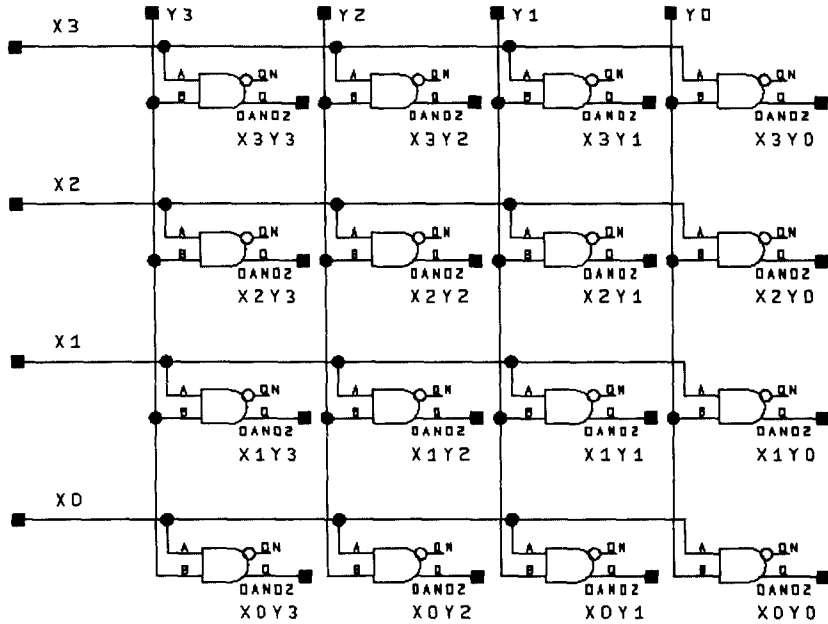


그림 7. 4비트 배열 승산기의 승산항 발생을 위한 논리곱 소자의 구성  
 Fig. 7. The configuration of an AND array for a 4 bit array multiplier.

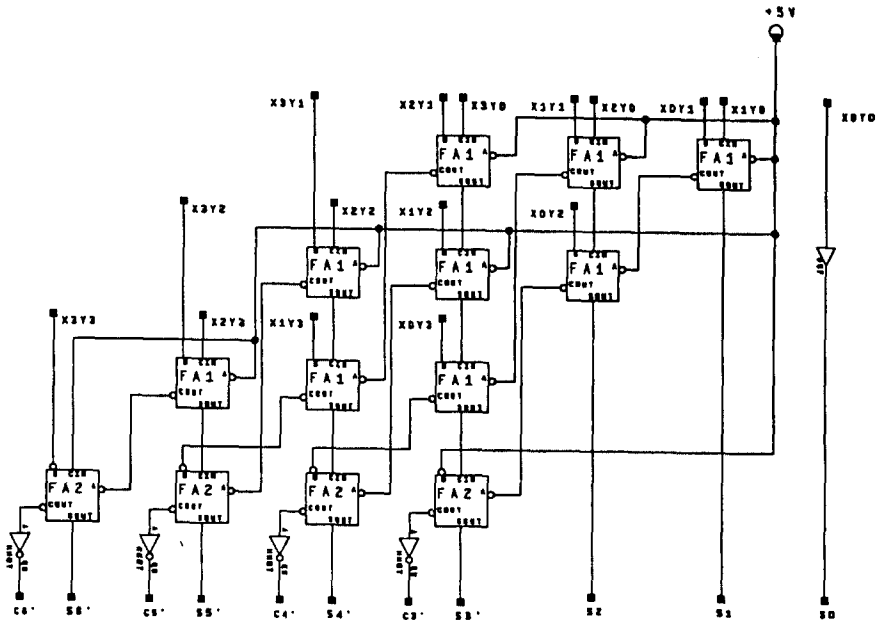


그림 8. 제안된 구조의 4비트 배열 연산기의 승산항 가산을 위한 CSA배열의 구성  
 Fig. 8. The configuration of a CSA array with the suggested scheme for a 4 bit array multiplier.

최종단에 구성된다. 이 CSA 열은 2개의 반전된 자리 올림 입력을 가산해야 하므로 2개의 반전된 입력을

받아 가산하는 FA2 소자를 사용한다. 그림 8에서 승산결과 하위 출력인 S0, S1, S2는

CSA 배열의 연산이 종료되면 바로 얻어지고 최종단에서 발생된 C7, C6, C5, C4 및 S7, S6, S5, S4는 자리올림 전파형 가산기를 이용하여 가산하여 승산결과의 상위 출력을 얻는다.

V. 실험 및 고찰

본 논문에서는 제안된 방식의 성능 평가를 위하여 기존의 방식에 의한 배열 승산기와 제안된 방식에 의한 배열 승산기를 구성하고 그 회로의 동작을 시뮬레이션 하였다. 배열 승산기의 동작은 다음의 3 단계로 구분하여 설명할 수 있다.

- 단계 1. 승산항의 발생을 위한 승수 및 피승수의 논리곱 발생 시간
- 단계 2. 발생된 승산항의 가산을 위한 CSA 배열의 동작시간
- 단계 3. CSA 배열의 최종단에서 발생된 자리올림 및 가산합의 가산을 위한 자리올림 전달형 가산기의 가산시간

단계 1의 소요시간은 약 0.6ns이며 단계 3의 소요시간은 일반적으로 사용되는 고속 가산기인 자리올림 예측형 가산기(carry lookahead adder)를 사용하는 경우 표 3과 같다.

표 3. 자리올림 예측형 가산기의 가산시간 (단위 ns)

Table 3. The operation time of a carry look-ahead adder (unit: ns).

| 가산 비트 수 | 8비트 | 16비트 | 32비트 |
|---------|-----|------|------|
| 가산 시간   | 6.2 | 8.6  | 11.1 |

제안된 구조에 의한 CSA 배열의 수는 기존 방식에 비하여 1개가 많다. CSA 구성에 소요되는 각 논리

표 4. 승산 비트 수에 따른 CSA 배열의 논리적 동작 지연

Table 4. The logical operation delay of CSA array with several size of multiplication bit.

| 승산 비트 수        | 16    | 32    | 64     |
|----------------|-------|-------|--------|
| CSA의 구조        |       |       |        |
| 기존 방식의 CSA 배열  | 30 Δt | 62 Δt | 126 Δt |
| 제안된 방식의 CSA 배열 | 16 Δt | 32 Δt | 64 Δt  |

소자의 지연을 1Δt라 가정하는 경우 기존 방식의 CSA 소자의 동작지연은 2Δt이고 제안된 방식의 CSA 소자의 동작지연은 1Δt 이다.

따라서 두 가지 방식에 의한 배열 승산기의 논리적 CSA 배열의 동작 지연 표 4와 같이 표현할 수 있다.

현재의 CMOS 표준 셀로 제공되는 1비트 전 가산기인 CSA의 가산 시간은 1.8ns<sup>[10]</sup> 이다. 제안된 방식으로 CSA 배열을 구성하는 그림 5의 경우 1개의 반전된 입력을 갖는 반전형 논리 배타합 소자(exclusive NOR)의 동작 지연은 최대 0.9ns 이고 이 소자의 입력중 하나가 반전된 값이므로 CMOS로 이러한 회로를 구현하면 0.2ns 이상 동작 지연을 감소시킬 수 있으며 fan out이 2개 이므로 이에 의한 동작지연 0.1nsec를 감안 한다면 이 소자의 최대 동작 지연은 최대 0.8ns로 볼 수 있다. 반전형 멀티플렉서의 동작 지연은 최대 0.7ns 이므로 제안된 방식으로 CSA 배열을 구성하는 경우 각 CSA의 가산 시간은 0.8ns 이하 이다. 이러한 시간 특성을 사용하여 CSA 배열의 연산 시간을 시뮬레이션 한 결과는 표 5와 같다.

표 5. 승산 비트 수에 따른 CSA 배열의 연산 시간 (단위: ns)

Table 5. The operation delay of CSA array with multiplication bit size (unit: ns).

| 승산 비트 수        | 16 비트 | 32 비트 | 64 비트 |
|----------------|-------|-------|-------|
| CSA의 구조        |       |       |       |
| 기존 방식의 CSA 배열  | 27    | 55.8  | 113.4 |
| 제안된 방식의 CSA 배열 | 12.8  | 25.6  | 51.2  |

표 3 및 표 5의 자료를 바탕으로 기존 방식과 제안된 방식의 배열 승산기의 승산시간을 비교하면 표 6과 같다.

표 6. 기존 방식과 제안된 방식에 의한 배열 승산기의 승산시간 비교 (단위: ns)

Table 6. The comparison of operation time between conventional and suggested array multiplier scheme (unit: ns).

| 승산 비트 수        | 16 비트                            | 32 비트                            | 64 비트                               |
|----------------|----------------------------------|----------------------------------|-------------------------------------|
| CSA의 구조        |                                  |                                  |                                     |
| 기존 방식의 CSA 배열  | 0.6<br>+ 27<br>+ 6.2<br>= 33.8 * | 0.6<br>+ 55.8<br>+ 8.6<br>= 55.8 | 0.6<br>+ 113.4<br>+ 11.1<br>= 125.1 |
| 제안된 방식의 CSA 배열 | 0.6<br>+ 12.8<br>+ 6.2<br>= 19.6 | 0.6<br>+ 25.6<br>+ 8.6<br>= 34.8 | 0.6<br>+ 51.2<br>+ 11.1<br>= 62.9   |

\* : 승산항 발생시간 + CSA 배열 연산시간 + 최종 가산시간 = 총 승산시간

표 6에서 보였듯이 승산 비트 수가 커질수록 총 승산 시간에서 CSA 배열의 연산 시간이 차지하는 비중이 커지므로 CSA 배열의 연산시간을 줄이면 배열 승산기의 고속화가 가능하게 된다. 제안된 구조의 CSA 배열은 기존 구조에 비하여 배열의 수가 하나 많고 사용되는 CSA 의 종류가 2가지이며 각 CSA 의 자리올림 출력 및 가수를 연결하는 배선이 1개의 CSA 열을 건너 뛰는 구조로 기존 방식에 비하여 구성에 소요되는 소자의 수가 많고 배선이 복잡하다는 단점을 가지나 기존의 배열 승산기의 구조를 일부 수정하는 것만으로 표 6에서 보였듯이 기존 방식에 비해 약 2배의 속도 향상을 얻을 수 있으며 승산 비트수가 커질수록 승산항의 가산을 위한 CSA 배열의 연산 시간이 차지하는 비중이 크게 되므로 고속 승산이 가능하게 된다.

## Ⅶ. 결 론

기존의 배열 승산기를 이용한 고속 승산 알고리즘에서는 배열 승산기 자체의 성능 향상을 위한 방식에 대한 연구보다는 배열의 수를 줄이는 방식에 대한 연구가 많이 수행되었다. 그러나 배열의 수를 줄이기 위하여는 추가되는 회로 및 배선 량이 증가하고 배선이 복잡하고 많아진다는 문제점을 안고 있다. 본 논문에서 제안된 승산 방식은 기존의 고속 승산방식 중 VLSI 구현에 가장 적합한 방식인 기존의 배열 승산기의 구조에서 각 CSA소자를 1개의 반전된 입력과 1개의 반전된 출력을 가진 형태로 구성하고 자리올림 출력을 2개의 CSA열씩 건너 뛰면서 전달 되도록 구성하여 약 2배의 연산속도를 갖는다. 제안된 구조의 배열 승산기는 기존 방식에 비하여 CSA 열의 수가 1개 많고 각 CSA에서 출력되는 자리올림 출력이 2개의 CSA 열을 건너 뛰어 연결되므로 기존 방식에 비하여 배선이 차지하는 면적이 늘어난다는 단점이 있다.

본 논문에서는 제안된 방식에 의한 16비트 및 32비트 배열 승산기의 회로를 구성하고 이를 시뮬레이션 하였다. 시뮬레이션은 모든 경우의 승수 및 피승수를 입력으로 인가하여 그 출력을 확인 하였고 승산에 소요되는 시간을 확인 하였다. 그러나 제안된 구조의 승산기가 차지하는 면적 및 회로 구성의 난이도는 반도체 실장을 통하여 확인할 수 있으며 이러한 부분에

대한 연구가 추진되어 제안된 배열 승산 방식에 대한 완전한 평가가 이루어져야 되리라 생각된다.

## 參 考 文 獻

- [1] 김용득, 정기현, 이용석, "자리올림의 전파특성을 이용한 가산기의 설계에 관한 연구", 전자공학회 논문지 제30권 5호, 1993.
- [2] A. D. Booth, "A signed binary multiplication technique," *Quarterly J. Mechan. Appl. Math.*, vol. IV, part2, 1951.
- [3] C. S. Wallace, "A suggestion for a fast multiplier", *IEEE trans. Electron. Comput.*, Feb. 1964.
- [4] L. Dadda, "Some scheme for parallel multiplier," *Alta Frequenza*, vol.34, pp.349-356, May. 1965.
- [5] Pezaris, S. D., "A 40ns 17bit-by-17bit array multiplier," *IEEE Trans. Comput.*, Apr. 1971.
- [6] C. R. Baugh, B. A. Wooley, "A two's complement parallel array multiplication algorithm," *IEEE Trans. Comput.*, C-22, pp.1045-1047, 1973.
- [7] O. L. MacSorley, "High speed arithmetic in binary computers," *Proc. IRE*, Jan. 1961.
- [8] S. Vassiliadis, E. M. Schwarz, and D. J. Hanrahan, "A general proof for overlapped multiple-bit scanning multiplication," *IEEE Trans. Comput.*, Feb. 1989.
- [9] Homayoon Sam and Arupratan Gupta, "A generalized multibit recording of two's complement binary numbers and its proof with application in multiplier implementation," *IEEE Trans. Comput.*, Aug. 1990.
- [10] Data book, "HSG1K/10K channelless gate array", Hyundai Electronics industries Co.,Ltd. Korea, 1991.

---

 著 者 紹 介
 

---

李 庸 奭(正會員) 第 30卷 B編 第 5號 參照

•

金 容 得(正會員) 第 30卷 B編 第 5號 參照  
현재 아주대학교 전자공학과 교수

洪 謹 善(正會員)

1964年 11月 15日生. 1987年 2月 아주대학교 전자공학과(학사). 1990年 2月 아주대학교 전자공학과(석사). 현재 아주대학교 전자공학과 연구원.