

## 계층적인 구조를 갖는 고속 병렬 곱셈기

## (A High Speed Parallel Multiplier with Hierarchical Architecture)

陳庸先\*, 鄭正和\*\*

(Yong-Sun Jin and Jong-Wha Chong)

## 요약

본 논문에서는 고속 4-2 compressor와 6-2 compressor를 사용한 계층적인 구조를 갖는 병렬 곱셈기를 제안한다. 병렬곱셈기는 일반적으로 CSA 덧셈기를 사용한 부분곱 덧셈 트리 블록의 처리속도에 영향을 받는다. 따라서, 본 논문에서는 일반적인 CSA 덧셈기 회로보다 전달 지연시간을 감소시킨 고속 4-2 compressor와 6-2 compressor 회로를 제안한다. 또한, 제안하는 compressor를 사용하여 16×16 병렬곱셈기의 처리속도를 향상시키며 규칙적인 레이아웃을 할 수 있는 계층적 곱셈기 구조를 제안한다. 제안하는 4-2 compressor 회로를 SPICE 시뮬레이션한 결과 기존의 4-2 compressor 회로에 비하여 전달지연 시간을 14% 감소시킬 수 있었다. 한편 제안하는 4-2 compressor와 6-2 compressor를 사용하여 16×16 비트 병렬곱셈기를 설계한 결과 일반 병렬곱셈기에 비하여 총 전달지연시간이 12% 이상 감소되었다.

## Abstract

In this paper, we propose a high speed parallel multiplier with a hierarchical architecture using a fast 4-2 compressor and 6-2 compressor. Generally, the performance of parallel multiplier depends on the processing speed of partial products summation tree with CSA adder. In this paper we propose a new circuit of 4-2 compressor and 6-2 compressor which reduces the propagation delay time, compared with conventional one. We propose a hierarchical multiplier architecture in order to improve the execution speed of 16×16 parallel multiplier using proposed compressors in this paper and make layout design easily by regular structure. The propagation delay time of the proposed 4-2 compressor circuit was 14% reduced as a result of SPICE simulation, compared with the conventional 4-2 compressor. The total propagation delay time of proposed 16×16 parallel multiplier was 12% reduced using proposed 4-2 compressor and 6-2 compressor.

## I. 서론

최근 정보통신의 발달로 멀티미디어 정보 전송이 요

\* 正會員, 京文大學 情報通信科

(Dept. of Information & Communication, Kyung Moon College)

\*\* 正會員, 漢陽大學校 電子工學科

(Dept. of Electronics Eng., Hanyang University)

接受日字:2000年7月26日, 수정완료일:2000年9월16일

구되어짐에 따라 고품질 정보의 실시간 처리가 필요하게 되었다. 멀티미디어 정보 처리는 데이터가 디지털로 변환되며 이에 따라 실시간에 많은 양의 데이터의 처리를 위해 고속 곱셈기가 필요하게 되었다. 고속 곱셈을 위한 곱셈기 구조는 어레이 곱셈기와 병렬 곱셈기로 나눌 수 있다.<sup>[4]</sup> 어레이 곱셈기는 단위 셀의 규칙성 및 VLSI 구현이 쉽다는 장점을 가지고 있으나 계산 속도가 곱해지는 비트 수에 비례한다는 단점을 가지고 있으며, 병렬 곱셈기의 경우 곱셈 비트 수가 증가함에 따라 곱셈 시간이 어레이 곱셈기 보다 적다는 장점을

가지고 있다.<sup>[5]</sup> 따라서 특정 목적에 따라 다양한 형태의 하드웨어 구조가 제안되고 있으며 응용 목적에 따라 적절한 선택이 요구된다. 즉, 파이프라인 방식의 병렬 곱셈기의 경우 특정 DSP 알고리즘 구현에 적합하고 트리 구조의 병렬 곱셈기의 경우 마이크로 프로세서나 범용 DSP 프로세서에 적합하다.<sup>[10]</sup> 곱셈 과정은 일련의 부분곱 덧셈 과정을 반복적으로 수행하는 것이기 때문에 고속 승산을 위해 부분 곱의 수를 줄이므로서 계산 단계를 줄이는 수정형 booth 알고리즘이 적용되고 있으며 또한 Wallace tree를 적용하여 부분 곱의 빠른 덧셈을 수행하고 있다.<sup>[11][12]</sup> 그러나 [1]에서는 booth 알고리즘의 하드웨어 구현이 부분곱 수의 감소를 상쇄시킬 만큼 크기 때문에 booth 알고리즘을 적용하지 않고 부분 곱을 생성하는 곱셈기를 제안하고 있다.

본 논문에서는 [1]에서 제안한 것처럼 수정형 booth 알고리즘을 적용하지 않고 부분 곱을 생성하며 빠른 부분 곱 덧셈을 위하여 일반적인 4-2 compressor 회로보다 전달지연시간을 감소시킨 새로운 4-2 compressor와 6-2 compressor 회로를 설계하여 병렬 곱셈기의 처리 속도를 향상시켰으며 곱셈기의 새로운 구조를 설계하여 전체 곱셈기의 처리 속도를 향상 시켰다. 또한, 본 논문에서 제안하는 병렬 곱셈기는 고속처리속도를 가지고 있고 규칙적인 구조를 가지기 때문에 입력 비트수 증가에 따른 확장성이 좋아 모듈화된 소프트웨어 곱셈기 컴파일러에 적용 가능하다.

II. 제안하는 곱셈 처리 방식

부호화된 정수에 대한 곱셈 수행시 승수가 음수인 경우 승수의 MSB에 의해 발생하는 부분 곱은 2의 보수를 취하여 계산이 되어야 한다. 또한 n×n 비트 곱셈기를 설계하고자 할 때 발생하는 부분 곱의 덧셈 시, 자리 수를 맞추기 위해 부호 비트가 확장되어 덧셈이 이루어진다. 본 장에서는 이러한 문제를 별도의 하드웨어의 부가 없이 부분 곱의 덧셈 과정에서 규칙적인 구조로 처리하는 방법을 제안한다.

1. signed와 unsigned 비트의 곱셈계산

본 논문에서는 입력이 n비트인 곱셈을 계산하기 위하여 승수와 피승수의 비트를 n비트와 n/2 비트로 분할하여 곱셈처리 하는데 이때 signed 비트와 unsigned 비트의 여러 가지 조합된 계산이 이루어진다. 16×16

곱셈의 계산시 발생하는 signed 비트와 unsigned 비트의 곱셈종류를 그림 1에 표시하였다.

그림 1과 같이 signed 비트와 unsigned 비트의 곱셈이 혼합 분할되어 처리되어도 그 계산이 정확하게 계산되기 위해서는 그림 2와 같이 부분 곱이 작성되어야 한다.

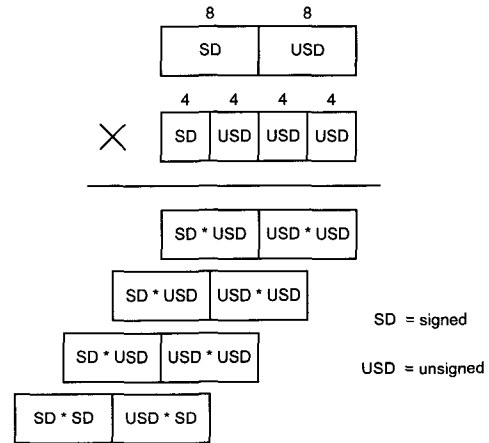


그림 1. signed/unsigned 계산의 부분 곱  
Fig. 1. Partial products of signed/unsigned multiplication.

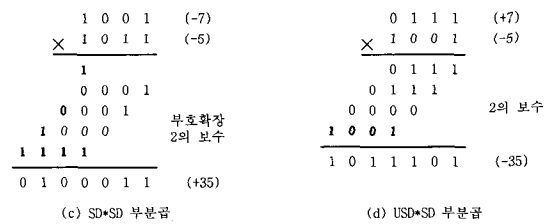
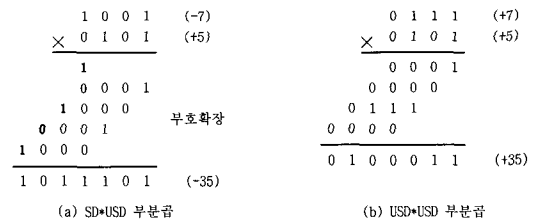


그림 2. 부분 곱의 생성  
Fig. 2. Generation of partial products.

2. 16×16 곱셈기의 부분 곱 생성

16×16 곱셈을 처리 시 그림 1과 그림 2에서 제시한 방법에 따라 그림 3과 같이 부분 곱을 생성하여 부분 곱 덧셈을 수행하면 정확하게 올바른 계산을 수행할

수 있다. 그러나, 부호확장과 2의 보수를 만들기 위하여 추가되는 2개의 "1" 비트 때문에 규칙적인 하드웨어를 구현하기가 어렵다. 이를 해결하기 위하여 부호확장을 위한 비트 "1"과 2의 보수를 위한 비트 "1"이 부분 곱 덧셈시 동일한 비트의 위치에서 더해지고 있다는 사실에 착안하여 이를 각각 더해주는 대신에 바로 차상위 비트에 "1"을 삽입하여 더해주는 것으로 대체 함으로써 그림 4와 같이 부분 곱을 구성하면 4-2 compressor에 의하여 부분 곱의 덧셈이 이루어 질 수 있으며 이를 통하여 보다 규칙적인 구조의 곱셈기를 구현할 수 있다.

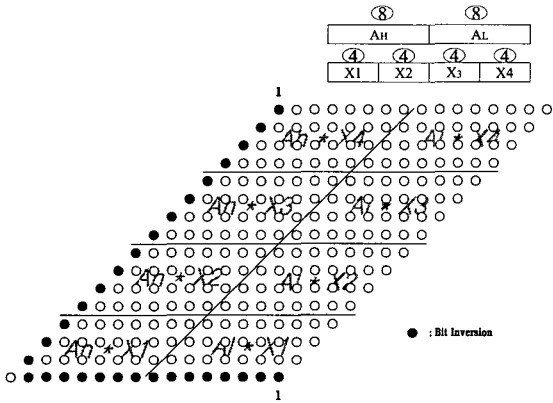


그림 3. 16×16 곱셈기의 부분 곱 생성  
Fig. 3. Partial product generation of 16×16 multiplier.

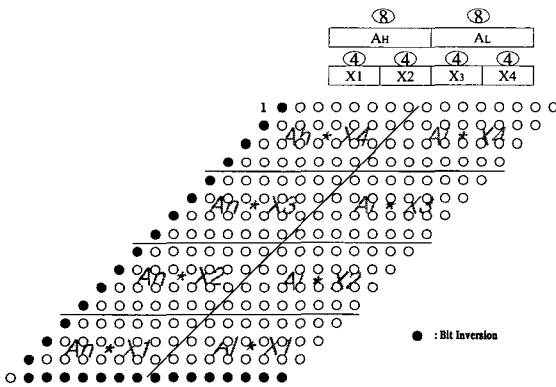


그림 4. 재구성된 16×16 곱셈기 부분 곱  
Fig. 4. Reorganized partial products of 16×16 multiplier.

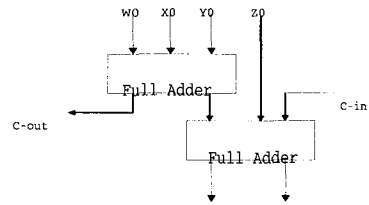
3. 병렬곱셈 처리방법

곱셈과정은 부분 곱의 생성과 생성된 부분 곱을 더하는 두 가지의 기본동작을 통하여 처리된다. 따라서

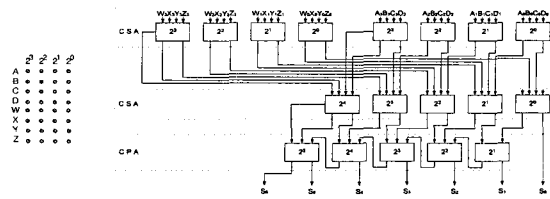
고속의 곱셈처리를 얻기 위하여는 생성되는 부분 곱의 수를 줄이는 방법과 부분 곱의 덧셈 처리속도를 증가시키는 것이다. 본 논문에서는 모든 부분 곱을 병렬로 동시에 생성하고 4-2 compressor와 6-2 compressor를 사용하여 multi-operand를 동시에 더하는 병렬곱셈기를 제안한다. 곱셈계산에서 two-operand 덧셈 기를 사용하여 multi-operand의 덧셈을 수행 할 때에는 carry 전달지연시간이 operand 수에 비례하여 증가한다. 그러나 아래 예제에서 보인 것처럼 8개의 operand를 더하는 경우를 예들들어 설명하면 그림 5에서 제안한 바와 같이 4-2 compressor를 사용하여 CSA(Carry Save Adder)를 구성 8개의 operand 덧셈을 동시에 병렬 처리 되도록 회로를 구성할 수 있다. 이때 덧셈시간은 다음 식과 같이 나타낼 수 있다.

$$T = 2 \cdot T_{CSA} + T_{CPA} \quad (T_{CSA} = 2 \cdot \Delta_{FA})$$

여기서  $T_{CSA}$ 는 CSA 한 단계를 병렬처리 하는데 걸리는 지연시간이고 8개의 4-2 compressor가 동시에 병렬로 처리된다. 그림 5에 CSA에 사용된 4-2 compressor의 등가회로를 나타내었다.  $\Delta_{FA}$ 는 1개의 Full Adder 처리지연시간이다.  $T_{CPA}$ 는 최종 부분 곱을 더하는 CPA(Carry Propagation Adder)를 처리하는데 걸리는 시간이다.



(a) CSA에 사용한 4-2 compressor 등가회로



(b) 8 비트 병렬덧셈 처리회로

그림 5. 8개 operand의 병렬 덧셈처리  
Fig. 5. Parallel addition of 8 operands.

그림 5에서 상위 첫 번째, 두 번째 단계는 4 비트

CSA이고 세 번째 단계는 4비트 CPA(Carry Propagation Adder)이다. 본 논문에서 제안하는 16×16 곱셈기에서는 빠른 처리속도를 얻기 위하여 최종단계의 CPA가 CLA(Carry Look Ahead Adder)로 대체되어 설계되었다.

### III. 제안하는 병렬 곱셈기 구조

#### 1. 계층적인 곱셈기 구조

본 논문에서는 입력이  $n$ 비트인 곱셈기를 설계하기 위한 계층적인 곱셈기 구조를 제안한다. 제안하는 기본적인 곱셈기 형태는 그림 6과 같이 승수와 피승수의 곱셈을  $n$ 비트와  $n/2$  비트로 나누어 곱하는 것이다. 적은 비트의 곱셈기에 의해  $n \times n$ 비트 곱셈기를 설계하고자 할 경우 승수와 피승수를 2의 배수에 의해 분할하고 곱해지는 비트의 크기는 항상 2:1을 유지시킨다. 이것을 그림 6에 표시하였다.

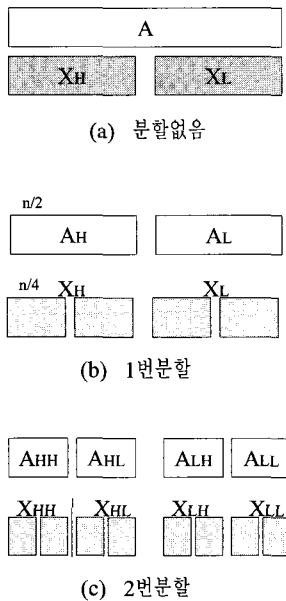


그림 6.  $n \times n$  곱셈기의 계층적 분할  
Fig. 6. Hierarchical partition of  $n \times n$  multiplier.

그림 6에서 제안한 계층적 곱셈 과정에 대한 부분 곱 생성 및 덧셈과정을 16×16 비트 곱셈기를 예를 들어 그림 7에 표시하였다.

피승수의 16비트 입력 데이터는 2로 나누어지고 승수의 16비트 입력 데이터는 4로 나누어져 8×4 부분

곱 생성에 의한 16×16 곱셈이 처리되도록 하였다, 이 때 피승수와 승수의 곱해지는 비트 수는 본 논문에서 제안하는 대로 2 : 1을 유지한다. 8×4 부분 곱 생성을 선택한 이유는 그림 10과 그림 11에서 제안하는 고속 4-2 compressor와 6-2 compressor 를 사용하고 부분 곱 덧셈을 위한 규칙적인 구조의 덧셈 트리를 취할 수 있기 때문이다.

한편, 부분 곱 덧셈시 두 번째 단계에서 마지막 부분 곱의 위치를 부분 곱 재 정렬하여 덧셈을 수행하면 세 번째 단계에서 6-2 compressor를 사용할 수 있게되어 그림 8에 표시한 바와 같이 부분 곱 덧셈의 처리시간을 감소 시킬 수 있다. 본 논문에서는 계층적 분할을 사용하는 곱셈기 구조와 4-2 compressor 그리고 6-2 compressor를 사용하여 빠른 처리속도를 갖는 고속병렬곱셈기 구조를 그림 9와 같이 제안한다.

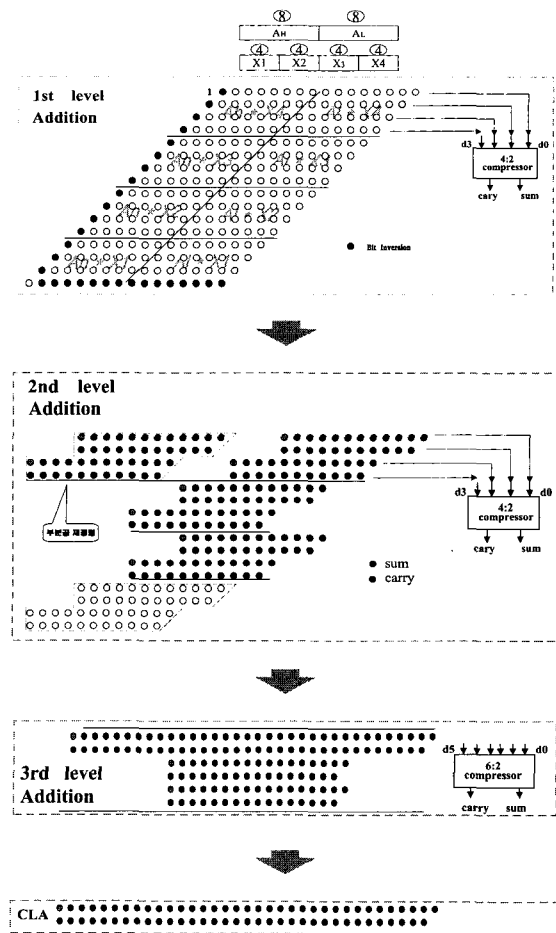


그림 7. 부분 곱의 덧셈 트리  
Fig. 7. Addition Tree of partial products.

Summation Level	Conventional	Proposed
1st	4:2 Comp. 2	4:2 Comp. 2
2nd	4:2 Comp. 2	4:2 Comp. 2
3rd	4:2 Comp. 2	6:2 Comp. 3
4th	4:2 Comp. 2 CLA	
Equivalent # of Full Adder	8	7
Equivalent Delay time	$T = 8 \times \Delta FA$	$T = 7 \times \Delta FA$

그림 8. 16×16 곱셈기의 구조변경으로 인한 처리속도 개선

Fig. 8. Speed improvement by modifying structure of 16×16 multiplier.

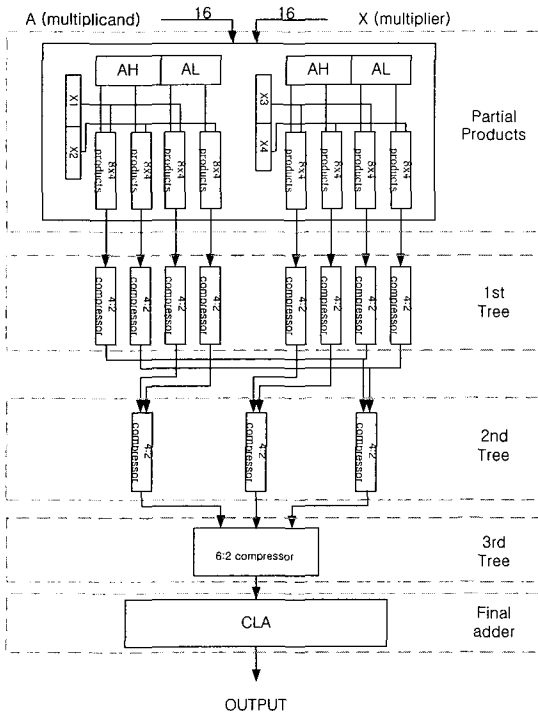


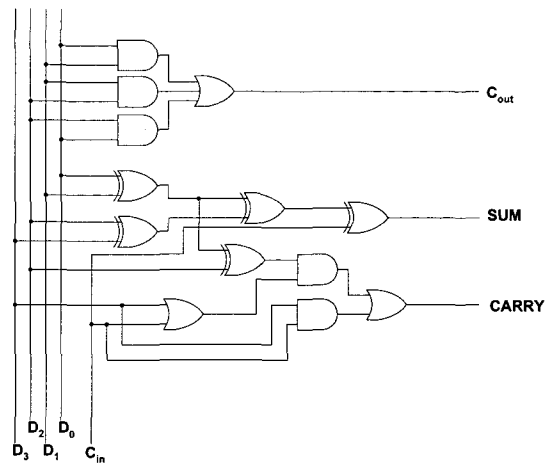
그림 9. 제안하는 16×16 병렬곱셈기 구조  
Fig. 9. Proposed architecture of 16×16 parallel multiplier.

2. 고속 4-2 compressor 설계  
고속 병렬 곱셈기는 3가지의 주요 부분으로 구성된다

다. 부분 곱의 수를 줄이기 위한 수정형 booth 엔코더, 만들어진 부분 곱을 더하는 CSA 구조를 통한 부분 곱 덧셈 트리, 그리고 최종결과를 더하는 CLA(Carry Look-Ahead Adder) 덧셈기로 구성된다.

그중 부분 곱 덧셈 트리 가 전체 고속 병렬 곱셈기의 처리 속도를 좌우하는 커다란 비중을 차지한다. 부분 곱 덧셈 기는 4-2 compressor를 사용한 CSA 구조를 통하여 일반적으로 구현된다. 따라서 빠른 처리속도를 갖는 4-2 compressor의 설계는 고속 병렬 곱셈기의 처리 속도를 개선하는데 중요한 역할을 하게된다.

본 논문에서는 기존에 제안되고 있는 일반적인 4-2 compressor에<sup>[2][3][13]</sup> 비교하여 빠른 처리 속도를 가진 새로운 고속 4-2 compressor를 제안한다.



(a) 논리 회로

Logic	TR 회로	Logic	TR 회로
AND		OR	
NAND		NOR	
XOR		MUX	
XNOR			

(b) 논리회로 구성에 사용된 DPL 트랜지스터 회로

그림 10. 제안하는 4-2 compressor  
Fig. 10. Proposed 4-2 compressor.

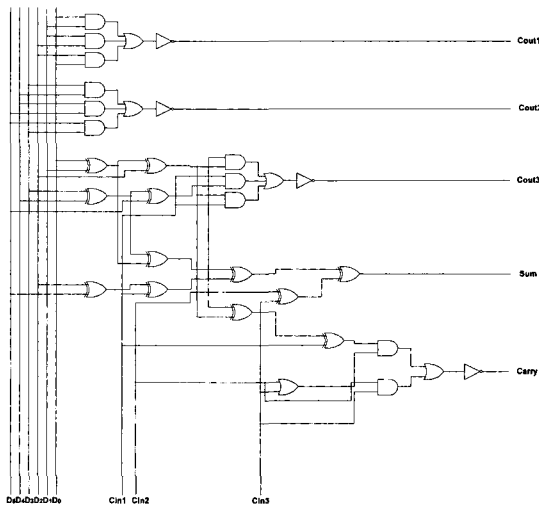
그림 10에 설계된 바와 같이 최장지연경로(critical path)인 입력신호 대 carry 출력의 회로를 기존논문<sup>[2]</sup>에 나타난 단점을 개선하여 carry 출력을 얻기 위한 입력

신호의 처리를 최 앞단에서부터 처리하도록 논리회로를 설계함으로써 최장지연경로의 입력대 출력의 전달 지연 시간을 감소시켰다. 이를 확인하고자 본 논문에서 제안하는 4-2 compressor와 기존 제안되고 있는 4-2 compressor<sup>[2][3][13]</sup>를 SPICE를 통한 회로 시뮬레이션을 하여 입력 대 최장지연경로의 전달지연(propagation delay) 시간을 비교하였다.

이때 사용된 트랜지스터 회로는 기존에 발표된 DPL(Double Pass-Transistor Logic)<sup>[6]</sup> 트랜지스터 회로를 사용하여 구성하였고 SPICE의 모델 파라미터는 0.8um CMOS 공정, 공급전압 3V, BSIM1 Hspice 모델을 사용하였다.

3. 6-2 compressor 설계

본 논문에서는 6-2 compressor 회로를 제안한다. 그림 11에서 표시한 바와 같이 6-2 compressor는 9개의



(a) 논리 회로

Logic	TR 회로	Logic	TR 회로
AND		OR	
NAND		NOR	
XOR		MUX	
XNOR			

(b) 논리회로 구성에 사용된 DPL 트랜지스터 회로

그림 11. 제안하는 6-2 compressor  
Fig. 11. Proposed 6-2 compressor.

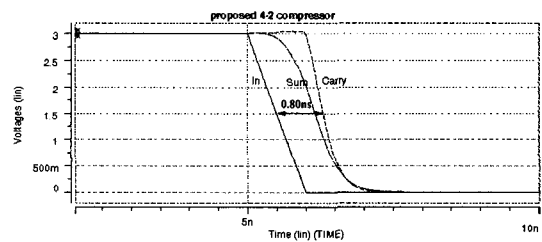
입력과 5개의 출력을 가지고 있다. Sum 출력은 9개 입력의 XOR 게이트 조합으로부터 얻어지고 carry 출력은 carry 발생회로와 XOR 게이트 조합으로 만들어진다. 그림 11은 설계된 6-2 compressor 의 논리회로도이다

IV. 실험 및 고찰

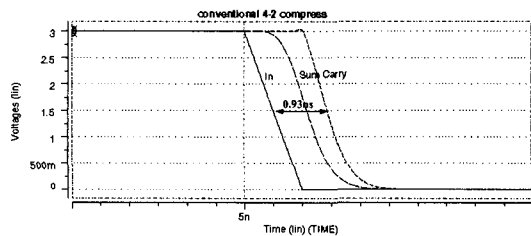
1. 고속 4-2 compressor

본 논문에서 제안한 고속 4-2 compressor의 성능 평가를 위하여 기존의 방식에 의한 4-2 compressor와 제안된 방식의 4-2 compressor의 회로를 DPL 트랜지스터 회로를 사용하여 구성하고 SPICE의 모델 파라미터는 0.8um CMOS 공정, 전원전압 3V, BSIM1 Hspice 모델을 사용하여 회로 시뮬레이션 하였다. SPICE를 사용한 회로 시뮬레이션 결과는 그림 12에 나타내었다.

입력펄스 신호의 하강시간(falling time)이 1[ns] 일 때 본 논문에서 제안한 4-2 compressor의 전달 지연시간은 0.8[ns] 이고, 일반적인 4-2 compressor는<sup>[2][3][13]</sup> 각각 0.93[ns], 0.94[ns], 0.94[ns] 이었다. 비교결과 본 논문에서 제안한 4-2 compressor의 전달지연시간이 일반적인 4-2 compressor에 비하여 14% 감소되어 동작 처리 속도가 개선되었다.



(a) 제안하는 4-2 compressor



(b) 일반적인 4-2 compressor

그림 12. 4-2 compressor의 SPICE simulation 결과  
Fig. 12. SPICE Simulation result of 4-2 compressor.

본 논문에서 제안하는 4-2 compressor와 기존 발표된 여러 가지 4-2 compressor<sup>[2][3][13]</sup>의 전달 지연시간의 비교를 그림 13에 정리하였다. 그림 13에 나타난 바와 같이 본 논문에서 제안한대로 설계된 4-2 compressor가 가장 빠른 처리속도를 가지고 있다.

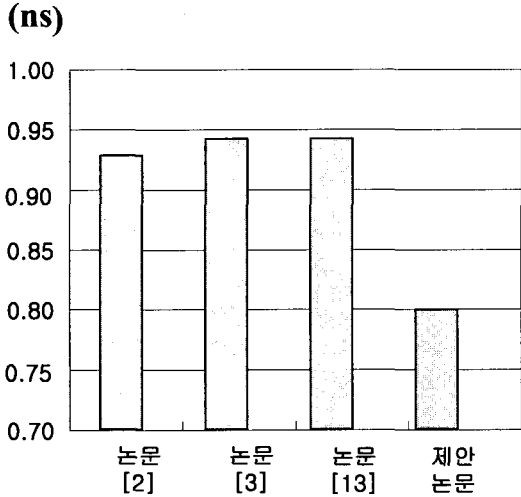


그림 13. 여러 가지 4-2 compressor 의 처리속도 비교  
Fig. 13. Speed comparisons of different 4-2 compressors.

2. 6-2 compressor

본 논문에서 제안한 6-2 compressor를 DPL 트랜지스터 회로를 사용하여 회로 구성하고 SPICE 모델 파라미터는 0.8um CMOS공정, 전원전압 3V, BSIM1 Hspice 모델을 사용하여 회로 시뮬레이션 하였고 그 결과는 그림 14에 나타내었다. 입력 펄스신호의 하강시간 1[ns] 일 때 본 논문에서 제안한 6-2 compressor의 전달지연시간은 1.10[ns] 이었다.

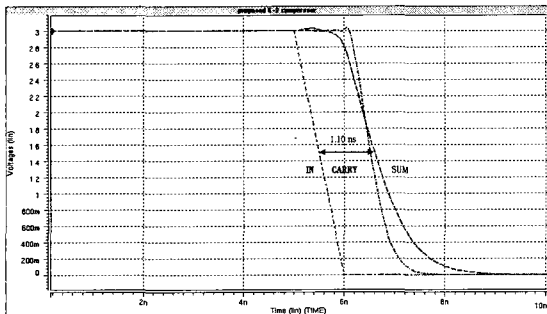


그림 14. 6-2 compressor 의 SPICE 회로 시뮬레이션 결과  
Fig. 14. SPICE simulation result of 6-2 compressor.

3. 16×16 병렬 곱셈기

본 논문에서 제안한 계층적 구조의 병렬 곱셈기의 회로 동작을 검증하기 위하여 본 논문에서 제안한 고속 4-2 compressor와 6-2 compressor 그리고 계층적인 곱셈기 구조를 적용하여 16×16 비트의 병렬곱셈기를 그림 15와 같이 구현하였다. 이때 사용한 부분 곱셈기는 booth 알고리즘을 사용하지 않고 AND 게이트의 배열로 구성하였으며 부분 곱 덧셈 트리는 본 논문에서 제안한 4-2 compressor와 6-2 compressor를 사용하였고 최종결과 덧셈은 CLA(Carry Look-Ahead Adder) 덧셈 기를 사용하였다.

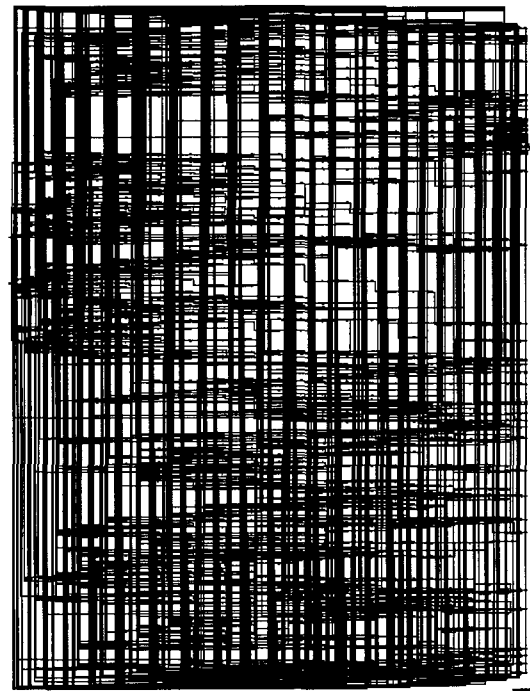


그림 15. 16×16 bit 곱셈기 회로  
Fig. 15. 16×16 bit multiplier circuit.

그림 15는 LSILOGIC사의 0.8um 스탠더드 셀 라이브러리를 사용하여 SYNOPSIS 사의 CAD 툴로 설계한 16×16 곱셈기 회로도이다. 전체 동작속도를 시뮬레이션 하여 기존의 병렬곱셈기와 속도를 비교한 결과 4-2 compressor 만으로 구성된 기존 병렬곱셈기 보다 본 논문에서 제안한 4-2 compressor와 6-2 compressor를 사용하여 계층적 구조로 설계된 병렬곱셈기가 총 전달 지연시간이 Spice 시뮬레이션 처리속도 결과로는 27%, 스탠더드셀로 구현된 시뮬레이션 처리속도 결과로는

표 1. 정규화된 16×16 병렬곱셈기의 전체 속도 비교

Table 1. Normalized speed comparison of 16×16 parallel multiplier.

구분	제안논문	논문[2]
스탠더드셀로 구현된 처리속도	0.88	1
Spice 시뮬레이션 처리속도	0.73	1

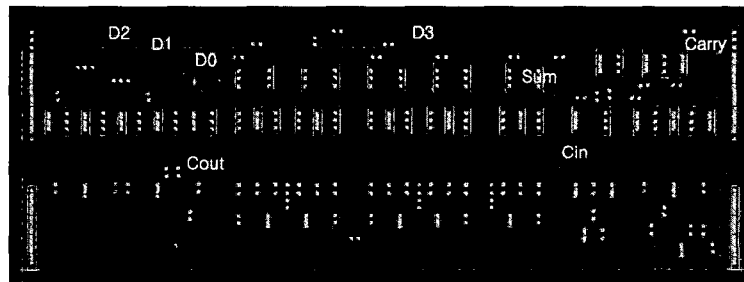
12% 감소되었다. 그 결과를 정규화 하여 표1에 나타내었다.

그림 16은 그림 15의 논리 시뮬레이션 결과로서 설계된 회로의 논리가 정확하게 동작하고 있음을 보여주고 있다.

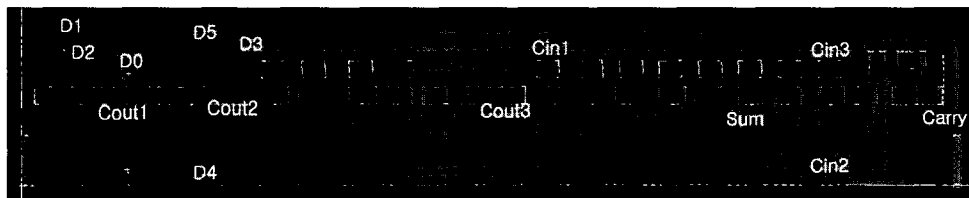
실제 검증된 병렬곱셈기 회로를 가지고 레이아웃의 Routability를 검증 하기 위하여 레이아웃을 시행하였다. 그림 17은 16×16 병렬곱셈기의 레이아웃을 보여주



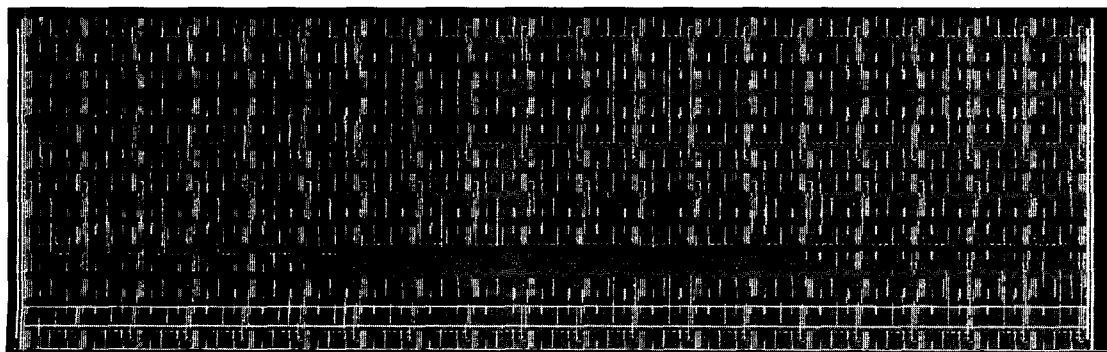
그림 16. 회로 그림 15 의 논리 시뮬레이션 결과  
Fig. 16. The simulation result of fig. 15.



(a) 설계된 4-2 compressor 레이아웃



(b) 설계된 6-2 compressor 레이아웃



(c) 설계된 16×16 병렬곱셈기 레이아웃

그림 17. 16×16 병렬곱셈기의 레이아웃 설계  
Fig. 17. Layout design of 16×16 parallel multiplier.



고 있다. 본 논문에서 제안한 병렬곱셈기는 규칙적인 구조를 가지고 있기 때문에 레이아웃 설계가 단순하고 간단하여 배선이 편리하며 곱셈기의 입력 비트 수가 증가하여도 확장성이 좋은 것으로 나타났다. 이때 분할에 따른 sign 비트와 unsigned 비트의 혼합곱셈의 처리는 본논문의 II장에서 제안하는 곱셈처리방식에 따라 그림 1, 그림 2, 그림 3, 그림 4에서 제시한 방법대로 부분곱을 작성한 후 덧셈을 실행하면 간단히 처리할 수 있다.

## V. 결 론

본 논문에서는 부분 곱 덧셈 트리의 처리속도를 향상시키기 위하여 일반적인 4-2 compressor 회로보다 처리속도가 향상된 고속 4-2 compressor와 6-2 compressor를 제안하였다. 제안한 고속 4-2 compressor와 6-2 compressor 회로를 사용하여  $16 \times 16$ 비트 고속 병렬 곱셈기를 설계한 결과 일반적인 4-2 compressor 회로를 사용한 병렬곱셈기에 비하여 전달지연시간이 12% 이상 감소되었다. 또한, 본 논문에서 제안하는 병렬곱셈기는 규칙적인 구조를 가지기 때문에 입력 비트수 증가에 따른 확장성이 좋아 병렬곱셈기의 확장설계에 편리하다. 제안하는 곱셈기는 고속처리속도를 가지고 있어 멀티미디어 데이터의 실시간 신호 처리에 응용될 수 있으며 확장성이 좋아 모듈화된 컴파일러에 적용 가능하다. 앞으로는 부분 곱 덧셈의 처리속도를 보다 빠르게 향상시키기 위하여 부분 곱의 재배열 방법과 여러 가지 종류의 다입력 compressor 설계에 대한 연구가 뒤따라야겠다.

## 참 고 문 헌

- [1] Mark R. Santoro and Mark A. Horowitz, "SPIM : A Pipelined  $64 \times 64$  bit Iterative Multiplier," IEEE J. of Solid-State Circuits, Vol. 24, No. 2, pp.487-493, Apr 1989.
- [2] Masato Nagamatsu, et al., "A 15-ns  $32 \times 32$ -b CMOS Multiplier with An Improved Parallel Structure," IEEE J. of Solide-State Circuits, Vol. 25, pp.494-497, Apr 1990.
- [3] Shen-Fu Hsaio, Ming-Roun Jiang and Jia-Sien Yeh, "Design of high speed low-power 3-2 counter and 4-2 compressor for fast multiplier," ELECTRONICS LETTERS, Vol. 34, No. 4, pp.341-343, Feb 1998.
- [4] Israel Koren, "Computer Arithmetic Algorithms", John Wiley Inc., pp.71-123, 1993.
- [5] Yoshita Harata, et al., "A High-Speed Multiplier Using a Redundant Binary Adder Tree," IEEE J. of Solide-State Circuits, Vol. sc-22, No. 1, pp.28-33, Feb 1987.
- [6] Makoto Suzuki, et al., "A 1.5ns 32-b CMOS ALU in Double Pass-Transistor Logic" IEEE J. of Solide-State Circuits, Vol. 28, No. 11, pp.1145-1151, Nov 1993.
- [7] Norio Ohokubo, et al., "A 4.4 ns CMOS  $54 \times 54$ -b Multiplier Using Pass-Transistor Multiplexer," IEEE J. of Solide-State Circuits, Vol. 30, No. 3, pp.251-257, Mar 1995.
- [8] Kazuo Yano, et al., "A 3.8-ns CMOS  $16 \times 16$ -b Multiplier Using Complementary Pass-Transistor Logic," IEEE J. of Solide-State Circuits, Vol. 25, No. 2, pp.388-395, Apr 1990.
- [9] C.F. Law and S.S. Rofail, "A Low-Power  $16 \times 16$ -b Parallel Multiplier Utilizing Pass-Transistor Logic," IEEE J. of Solide-State Circuits, Vol. 34, No. 10, pp.1395-1399, Oct 1999.
- [10] 김득경, 신경욱, 이용석, 이문기, "4-비트 리코딩 기법을 이용한  $64 \times 67$  병렬 승산기 설계," 대한 전자공학회 추계종합학술대회 논문집 17권 2호, pp.1289-1292, 1994
- [11] 강정엽, 이원형, 권오영, 한탁돈, "4-2 콤퍼레서를 이용한 승산기 모듈생성기의 설계," 대한 전자공학회 추계종합학술대회 논문집 16권 2호, pp.388-392, 1993
- [12] 김영민, 조진호, "32×32 비트 고속 병렬 곱셈기 구조," 전자공학회논문지 31권 B편 10호 pp.1505-1510, 1994
- [13] Masanori Isumikawa, et al., "A 0.25um CMOS 0.9V 100Mhz DSP Core," IEEE J. of Solide-State Circuits, Vol. 32, No.1, pp.52-61, Jan 1997.

저 자 소 개



陳庸先(正會員)

1955년 2월 29일생. 1978년 한양대학교 전자공학과 졸업(학사). 1991년 한양대학교 대학원 전자공학과 졸업(석사). 1991년~현재 한양대학교 대학원 전자공학과. 박사과정. 1978년~1979년 전자통신연구원 연구원. 1980년~1987년 LG전자연구소 선임연구원. 1988년~1996년 Texas Instrument 연구소 연구원. 1997년~현재 경문대학 정보통신과 교수. <관심분야> 신호처리, VLSI설계, 디지털통신 시스템

鄭正和(正會員) 第37卷 SP編 第2號 參照