
저전력 디지털 신호처리 응용을 위한 작은 오차를 갖는 절사형 Booth 승산기 설계

정해현* · 박종화** · 신경욱***

A Design of Low-Error Truncated Booth Multiplier for Low-Power DSP
Applications

Hae-Hyun Jung* · Jong-Wha Park** · Kyung-Wook Shin***

이 논문은 2001년도 금오공과대학교 학술연구비를 지원에 의한 연구결과의 일부임

요 약

N-비트×N-비트 2의 보수 승산에서 승산결과 2N-비트 중 상위 N-비트만을 출력하는 절사형 Booth 승산기의 절사오차 최소화를 위한 효율적인 오차보상 방법을 제안하였다. 제안된 방법을 적용하여 작은 칩 면적과 저전력 특성을 갖는 절사형 승산기를 설계하고 면적, 절사오차 등을 기존의 방식과 비교하였다. 제안된 절사형 Booth 승산기는 승산결과의 하위 N-비트를 계산하는 회로를 생략하므로, 절사되지 않은 일반 승산기에 비해 게이트 수가 약 35% 정도 감소한다. 본 논문에서 설계된 절사형 Booth 승산기는 기존의 고정 오차보상 방법을 적용한 경우에 비해 평균오차를 약 60% 정도 줄일 수 있다. 제안된 방법을 적용하여 16-비트×16-비트 절사형 승산기를 0.35- μm CMOS 공정을 이용하여 full custom 방식으로 설계하였다. 약 3,000개의 트랜지스터로 구성되는 승산기 코어는 330- μm ×262- μm 의 면적을 가지며, 3.3-V 전원전압에서 200-MHz로 동작 가능하며 약 20-mW의 전력소모 특성을 갖는다.

ABSTRACT

This paper describes an efficient error-compensation technique for designing a low-error truncated Booth multiplier which produces an N-bit output from a two's complement multiplication of two N bit inputs by eliminating the N least-significant bits. Applying the proposed method, a truncated Booth multiplier for area efficient and low-power applications has been designed, and its performance(truncation error, area) was analyzed. Since the truncated Booth multiplier does not have about half the partial product generators and adders, it results an area reduction of about 35%, compared with non-truncated parallel multipliers. Error analysis shows that the proposed approach reduces the average truncation error by approximately 60%, compared with conventional methods. A 16-b×16-b truncated Booth multiplier core is designed in full-custom style using 0.35- μm CMOS technology. It has 3,000 transistors on an area of 330- μm ×262- μm and 20-mW power dissipation at 3.3-V supply with 200-MHz operating frequency.

키워드

Multiplier, Booth multiplier, Fixed-width multiplier, Truncated multiplier

*금오공과대학교 전자공학부 석사과정
***금오공과대학교 전자공학부 부교수

**시누로직(주) EDA 사업부
접수일자 : 2002. 4. 11

1. 서론

승산기는 통신 및 디지털 신호처리 프로세서, 디지털 필터, 마이크로 프로세서 등의 핵심 구성 요소이며, 칩의 면적과 동작 속도에 큰 영향을 미치는 연산장치이다. 두 이진수에 대한 승산은 부분곱 생성과 생성된 부분곱의 가산으로 이루어지며, 부분곱 생성과 가산 방식에 따라 매우 다양한 승산기 구현 방법들이 제안되고 있다. 널리 사용되고 있는 승산기 구조는 전가산기 배열을 이용하여 부분곱을 가산하는 배열 승산기와 수정형 Booth 알고리즘^[1,2]을 이용하여 부분곱의 수를 줄이는 Booth 승산기 등이 있다. 일반적으로, 승수와 피승수가 각각 N-비트인 승산의 결과는 2N-비트가 되며, 승산기의 비트 수가 커질수록 부분곱의 수와 부분곱의 비트 수가 증가하므로 승산기 회로의 복잡도, 승산시간, 전력소모 등이 크게 증가한다^[3].

FIR(finite-impulse response) 필터, IIR(infinite-impulse response) 필터, 등화기 등 대부분의 디지털 신호처리 응용분야에서는 고정된 비트 수를 갖는 데이터 패스가 사용되므로, 내부 연산과정의 승산결과를 고정된 비트 수로 잘라내어 사용한다. 데이터 패스의 비트 수가 N-비트로 고정된 디지털 신호처리 분야에서 N-비트×N-비트 승산기가 사용되는 경우, 승산결과와 하위 N-비트를 절사(truncation)하고 상위 N-비트만을 사용해야 한다.

N-비트×N-비트 승산결과를 N-비트로 절사하는 방법은 크게 두 가지로 나눌 수 있다. 첫째, 2N-비트의 승산결과를 정확하게 계산한 후, 하위 N-비트를 잘라버리는 방법은 절사오차는 작지만 승산결과와 하위 N-비트를 계산하는 회로가 전부 사용되므로 일반 승산기와 동일한 면적과 전력소모를 갖는 단점이 있다. 둘째, 승산결과와 하위 N-비트를 계산하는 회로를 생략하여 칩 면적과 전력소모를 일반 승산기에 비해 약 절반으로 줄이는 방법을 생각할 수 있으며, 이를 절사형 승산기(truncated multiplier)라고 한다. 그러나, 절사형 승산기는 절사된 하위 N-비트에 대한 연산이 생략되므로 승산결과와 절사오차가 커지게 되며, 따라서 절사오차를 보상해 주기 위한 적절한 방법이 필요하다.

절사형 승산기의 오차보상 방법은 확률적 접근방법에 기초한 고정 오차보상 방법과 근사화된 캐리 입력 신호(바이어스 신호)를 만들어 이용하는 가변 오차보

상 방법의 두 가지로 구분된다. Kidambi에 의해 처음으로 제안된 고정 오차보상 방법^[4]은 배열형 승산기의 절사된 하위 N-비트의 부분곱에서 발생할 수 있는 캐리신호를 확률적으로 분석하여 고정된 보상값을 구하는 방법이다. 이 방법은 오차보상을 위해 추가적인 회로가 필요 없다는 장점은 있으나, 오차보상 후의 절사오차가 비교적 크다는 단점을 갖는다. 최근, Kidambi의 고정 오차보상 방법을 Booth 승산기에 적용한 결과도 발표되었다^[5]. 한편, Jou는 절사된 부분곱의 하위 비트 중 일부를 이용하여 근사화된 캐리 신호(바이어스)를 생성하고, 이를 가산하는 진보된 형태의 오차보상 방법을 배열형 승산기 구조에 대해 제안하였다^[6]. 이 방법은 Kidambi가 제안한 고정 오차보상 방법에 비해 절사오차를 크게 줄일 수 있다는 장점을 가지며, 바이어스 생성 회로도 매우 단순하다. 그러나, 배열구조의 절사형 승산기에만 적용될 수 있으며, 절사형 Booth 승산기의 오차보상에는 직접 적용될 수 없다.

본 논문에서는 2의 보수 승산을 위한 절사형 Booth 승산기의 오차보상 방법을 제안하였으며, 이를 적용한 승산기의 면적, 절사오차 등을 기존의 방식과 비교하였다. 제안된 방법을 적용한 16-비트×16-비트 절사형 Booth 승산기를 full-custom 방식으로 설계하고, 전력소모 특성을 분석하였다.

II. 절사형 Booth 승산기의 오차 보상

일반적으로, 승산기는 승수의 비트 수 만큼 부분곱을 생성하여 가산하는 배열형 승산기와 승수에 대한 리코딩을 통해 부분곱의 수를 줄이는 Booth 승산기로 구분되며, 따라서 절사형 승산기도 위의 두 가지 방법으로 구현할 수 있다. 본 장에서는 N-비트×N-비트 2의 보수 승산의 결과 2N-비트 중, 하위 N-비트를 잘라버리고 상위 N-비트만을 취하는 절사형 승산기의 오차보상 방법에 대해 기술한다. 편의상, N=8인 경우를 예로 들어 설명한다.

절사형 배열 승산은 그림 1에 표시된 cut-line 우측의 하위 N-비트 부분을 잘라 버리고 부분곱의 나머지 상위 N-비트와 오차보상을 위한 바이어스 신호를 가산하여 N-비트의 승산결과를 얻는다. 이에 대한 오차보상은 아래의 식(1)과 같이 표현된다^[6].

$$\alpha_{n-1} \cong \bar{\alpha}' = \begin{cases} \sum_{\substack{i+j=n-1 \\ i,j \neq n-1}} x_i y_j + 1, & \text{if } \beta = 0 \\ \sum_{\substack{i+j=n-1 \\ i,j \neq n-1}} x_i y_j, & \text{if } \beta > 0 \end{cases} \quad (1)$$

단, $\beta = \sum_{i=0}^{N-1} x_{N-1-i} y_i$ 이며, 절사된 부분의 최상위 비트에서 발생하는 캐리의 합을 나타낸다. $N=8$ 인 경우, 오차보상에 사용될 바이어스 신호는 식(2)에 의해 6개의 OR 게이트와 1개의 NOR 게이트로 생성된다.

i) $\beta > 0$ 인 경우 ;

$$\alpha_{n-1} = (x_7 y_0 + x_6 y_1 + x_5 y_2 + x_4 y_3 + x_3 y_4 + x_2 y_5 + x_1 y_6 + x_0 y_7) \cdot 2^8 \quad (2-a)$$

ii) $\beta = 0$ 인 경우 ;

$$\alpha_{n-1} = (x_7 y_0 + x_6 y_1 + x_5 y_2 + x_4 y_3 + x_3 y_4 + x_2 y_5 + x_1 y_6 + x_0 y_7 + 1) \cdot 2^8 \quad (2-b)$$

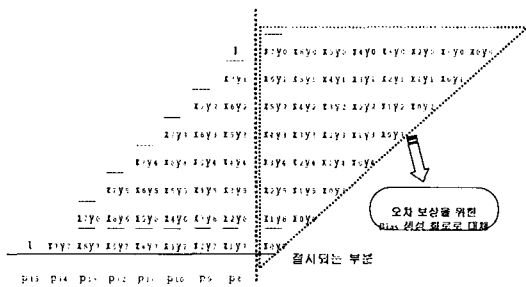
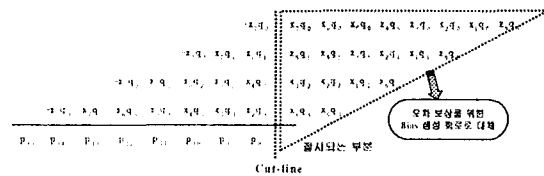


그림 1. 절사형 배열 승산(8-비트×8-비트의 경우)
Fig. 1 Truncated array multiplication (8-b×8-b)

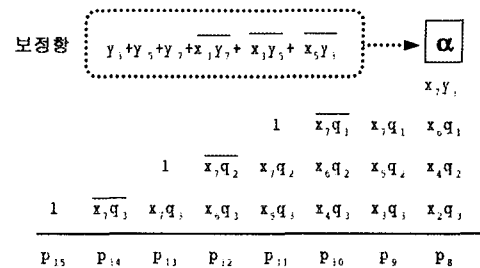
한편, 절사형 Booth 승산도 동일한 방식으로 그림 2-(a)에 표시된 cut-line 우측의 하위 N-비트 부분을 잘라버리고 부분곱의 나머지 상위 비트만을 가산하여 N-비트의 승산결과를 얻는다. 그러나 Booth 승산의 경우, 승수 3-비트 단위로 Booth 인코딩 되어 부분곱이 생성되므로 인접한 부분곱들은 서로 독립적이지 않다. 따라서, 그림 2 (a)와 같은 cut line으로 절사된 Booth 승산에 식(1)과 식(2)에 정의된 오차 보상방법

을 적용하면 절사오차가 매우 커지게 된다.

본 논문에서는 이와 같은 문제점을 해결하기 위해 Booth 승산의 부분곱과 배열형 승산의 부분곱 사이의 관계에 대한 분석을 통해 절사형 Booth 승산기의 효율적인 오차보상 방법을 고안하였다. 이를 위해, 그림 2-(a)의 cut-line으로 절사된 부분곱을 Booth 인코딩의 역 과정으로 분해한 후, 이로부터 보정항 α 를 유도함으로써 그림 2-(b)와 같은 절사형 Booth 승산의 변형된 부분곱을 얻었으며, 따라서 식(1)과 식(2)에 정의된 오차보상 방법을 적용할 수 있도록 하였다.



(a) partial products and cut-line for truncation



(b) proposed error compensation method

그림 2. 절사형 Booth 승산(8-비트×8-비트)

Fig. 2 Truncated Booth multiplication(8-b×8-b)

보정항 α 는 다음과 같은 두 가지를 고려하여 생성된다. 첫째, Booth 인코딩에 의해 생성되는 음의 부분곱 ($-X$, $-2X$)을 처리하기 위해, 절사 승산기의 최하위 비트 위치에 Booth 인코더에서 생성되는 부호백터 (y_3, y_5, y_7)를 가산해야 한다. 둘째, 그림 2-(a)에 표시된 cut-line으로 절사된 부분곱의 최하위비트 위치에 $-(x_1 y_7 + x_3 y_5 + x_5 y_3) \cdot 2^8$ 항을 추가해야 한다. 한편, 그림 2 (a)의 cut line으로 절사된 부분곱에는 $(x_6 y_1 + x_4 y_3 + x_2 y_5) \cdot 2^8$ 가 포함되어 있으므로, 오차보상을 위한 바이어스 회로에서 제거되며 이에 의해

절사형 배열 승산기에 비해 오차보상회로가 단순해진 다. 따라서, 보정항 α 를 포함하는 절사형 Booth 승산은 그림 2-(b)와 같이 된다.

III. 절사형 Booth 승산기 설계

3.1 회로 설계

본 장에서는 2장에서 설명된 오차보상 방법을 적용한 절사형 Booth 승산기 설계에 대해 기술한다. 그림 3은 오차보상 회로를 갖는 8-비트×8-비트 절사형 Booth 승산기의 구조를 나타낸 것이며, 부분곱 생성블록, 부분곱 가산회로, 오차보상 회로 등으로 구성된다. 부분곱 생성블록은 승수에 대한 Booth 인코딩을 수행하는 인코더 블록과 여기서 생성된 3-비트 신호와 피승수를 받아 부분곱을 생성하는 디코더 블록으로 구성된다. 디코더 블록에서는 절사된 부분곱의 최하위 비트 위치(즉, 2^8 위치)에 가산될 보정항 α 가 생성된다. 한편, 첫 번째 부분곱 x_7y_1 은 Booth 인코더/디코더 블록을 거치지 않고 AND 게이트를 이용하여 직접 생성된다. 생성된 부분곱은 캐리 보존 가산기(carry-save adder)에 의해 가산되어 최종 승산결과가 얻어진다.

그림 4는 오차 보상 회로와 보정항 처리 회로가 포함된 부분곱 가산 블록의 내부 구성도이며, 부호확장 제거 기법^[7]을 적용하였다. 오차보상 회로는 4-입력 OR 게이트 1개, 2-입력 OR 게이트 3개, 그리고 2-입력 NOR 게이트 1개로 구성되며, 부분곱 가산 블록의 최하위 비트(즉, 2^8 위치)로 입력되는 4개의 오차보상 신호를 생성한다. 오차보상 회로는 매우 단순하므로, 승산기 전체 면적에서 차지하는 비율은 5% 이하로 거의 무시할 수 있다. 한편, 보정항 처리 회로는 그림 2-(b)에 표시된 보정항 α 와 첫 번째 부분곱 x_7y_1 을 가산하기 위해 사용된다.

설계된 절사형 Booth 승산기는 VHDL을 이용하여 모델링되었으며, Max+Plus II와 C언어를 사용하여 검증하고, Synopsys를 사용하여 합성하였다. 그림 5는 논리검증 결과를 보인 것이다. 그림 5-(a)는 절사되지 않은 일반 승산기의 승산결과 16-비트를 보인 것이며, 그림 5-(b)는 본 논문에서 설계된 절사형 승산기의 출력을 보인 것이다.

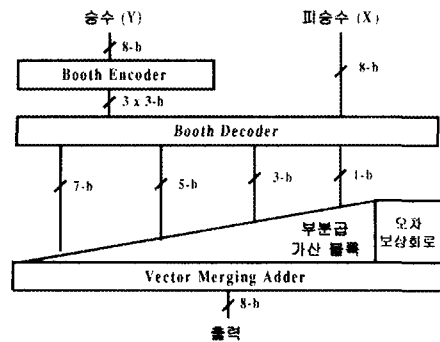


그림 3. 오차보상을 갖는 8-비트×8-비트 절사형 Booth 승산기의 구조
Fig. 3 8-b×8-b truncated Booth multiplier with error compensation

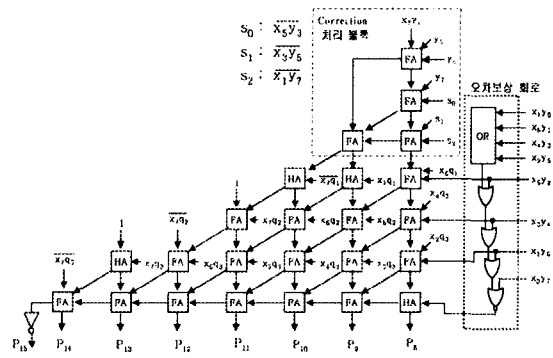
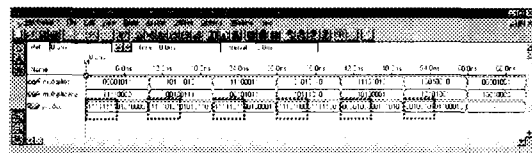
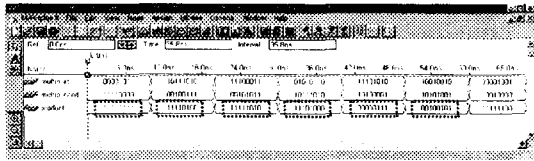


그림 4. 제안된 방법에 의한 오차보상 및 보정항 처리를 갖는 부분곱 가산블록 (8-비트×8-비트)
Fig. 4. Partial product addition block with the proposed error compensation and correction method (8-b×8-b)

일반 승산기의 출력 중 상위 8-비트(그림 5-(a)에서 점선으로 표시된 부분)와 그림 5-(b)의 8-비트 승산결과를 비교하면, 일부 승산 결과의 최하위 비트 값에만 차이가 있어 절사 오차가 매우 작음을 알 수 있다.



(a) standard multiplier without truncation



(b) truncated Booth multiplier with error compensation
 그림 5. 8-비트×8-비트 승산기의 시뮬레이션 결과
 Fig. 5 Simulation results of 8-b×8-b multipliers

3.2 성능 분석

절사형 승산기의 성능은 승산결과에 대한 평균 오차와 분산 그리고 면적을 통해 평가될 수 있다. M 개의 데이터에 대한 절사 승산기의 평균오차 E 는 식(3)과 같이 정의된다.

$$E = \frac{1}{M} \sum_{i=1}^M (\Gamma_i - \Gamma'_i) \quad (3)$$

식(3)에서 Γ_i 는 절사되지 않은 승산결과 상위 비트의 값을 나타내며, Γ'_i 는 절사 승산기의 출력 N 비트의 값을 나타낸다.

표 1은 10,000개의 데이터에 대해 구해진 평균 오차를 비교한 것이다. 본 논문의 오차보상 방법을 적용한 절사형 Booth 승산기는 오차보상이 없는 절사 승산기에 비해 평균오차가 약 50%에서 70%까지 감소하며, 또한 문헌 [4]의 고정 오차보상 방법을 적용한 승산기에 비해서도 약 60% 정도 감소한다. 본 논문의 방법은 승산기의 비트 수가 증가할수록 기존의 방법에 비해 평균 오차가 더욱 감소함을 알 수 있다.

표 2는 제안된 방법으로 설계된 절사형 Booth 승산기의 게이트 수(Synopsys 합성결과)를 일반 Booth 승산기 및 문헌 [6]의 절사형 배열 승산기와 비교한 것이다. 절사되지 않은 일반 승산기에 비해 약 35%의 면적 감소가 얻어지며, 문헌 [6]의 승산기와 비슷한 면적으로 구현된다. 따라서, 본 논문의 방법은 면적과 평균 오차에서 우수한 성능을 가지며, 승산기의 비트 수가 클수록 평균 오차 및 면적이 감소하므로 비트 수가 큰 경우에도 적합하다.

표 1. 평균오차 비교
 Table 1. comparison of average error

비트 수	오차보상 없는 절사 승산기	문헌 [4]의 승산기	본 논문의 승산기
8	130.5 (1.0)	187.9 (1.44)	65.0 (0.50)
12	4,099.5 (1.0)	3,927.9 (0.96)	1,570.3 (0.38)
16	98,308.5 (1.0)	74,497.4 (0.76)	30,403.7 (0.31)

표 2. 게이트 수 비교
 Table 2. comparison of gate count

비트 수	일반 승산기	문헌 [6]의 승산기	본 논문의 승산기
8	568 (1.0)	370 (0.65)	374 (0.66)
12	1,134 (1.0)	780 (0.69)	734 (0.65)
16	1,865 (1.0)	1,299 (0.70)	1,158 (0.62)

3.3 Full-custom 설계 및 시뮬레이션 결과

설계된 16 비트×16 비트 절사형 Booth 승산기는 4층 금속배선을 갖는 0.35- μm N-Well CMOS 공정을 사용하여 full-custom 방식으로 레이아웃 설계를 하였다. 배선에 의한 면적소모를 최소화함과 동시에 설계의 효율성을 높이기 위해 각 단위 셀들의 외부 연결단자를 표준화하였으며, 이를 통하여 셀들의 규칙적인 연결에 의해 레이아웃이 완성되도록 하였다. 설계된 승산기의 레이아웃은 그림 6과 같으며, 약 3,000개의 트랜지스터로 구성되며, 코어의 면적은 약 330- μm ×262- μm 이다.

설계된 회로의 지연시간 특성을 분석하기 위해 0.35- μm 공정 파라미터를 사용하여 HSPICE 시뮬레이션을 수행하였다. 시뮬레이션의 정확도를 높이기 위해 기능블록 상호간의 부하효과를 고려하여 버퍼 크기를 결정한 후 시뮬레이션을 수행하였다. 설계된 절사형 승산기에 대한 HSPICE 시뮬레이션 결과, 전원전압 3.3 V에서 약 4.5 ns의 최대 지연시간을 가져 200-MHz의 클럭 주파수로 안전하게 동작할 것으로 예상된다.

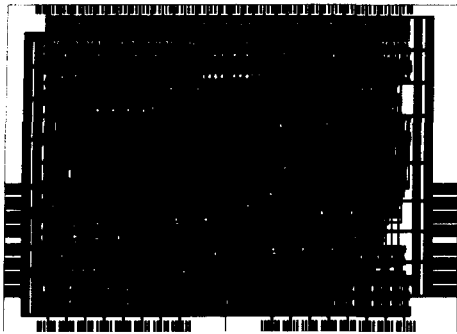


그림 6. 16-비트×16-비트 절사형 Booth 승산기의 레이아웃
Fig. 6 Layout of 16-b×16-b truncated Booth multiplier

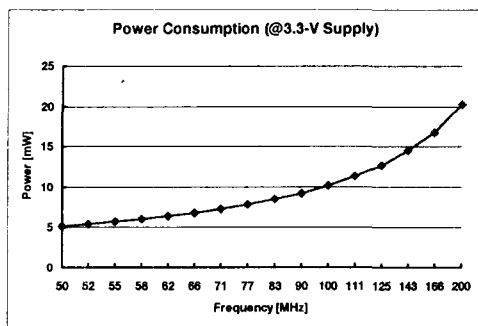


그림 7. 설계된 16-비트×16-비트 절사형 Booth 승산기의 전력소모
Fig. 7 Power dissipation of the designed 16-b×16-b truncated Booth multiplier

한편, 설계된 절사형 승산기의 동작 주파수에 따른 전력소모 특성을 분석하기 위해 랜덤 입력벡터를 인가한 상태에서 HSPICE 시뮬레이션을 수행하였으며, 그림 7은 시뮬레이션으로 얻어진 평균 전력소모 특성을 나타낸 것이다. 설계된 승산기는 3.3-V 전원전압으로 동작하는 경우, 200-MHz의 동작속도에서 약 20-mW의 전력을 소모하며, 100-MHz의 동작속도에서는 약 10-mW의 전력을 소모하는 것으로 예측되었다. 표 3은 설계된 16-비트×16-비트 절사형 Booth 승산기의 특성을 요약한 것이다.

표 3. 16-비트×16-비트 절사형 Booth 승산기의 설계 결과
Table 3. Summary of the 16-b×16-b truncated Booth multiplier

Technology	0.35- μm QLM CMOS
Operating frequency	200-MHz @3.3-V
Power dissipation	20.26-mW (@200-MHz, 3.3-V)
Transistor count	2,990
Area	330- μm ×262- μm
Layout density	34.58k transistors/ mm^2

IV. 결론

일반적으로, 고정 비트 크기를 갖는 디지털 신호처리 응용에서는 승산결과 상위 비트 절반만 연산에 사용된다. 따라서, 승산기 설계 시에 부분곱의 하위 비트 절반에 대한 연산을 생략함으로써 승산기가 차지하는 면적과 전력소모를 줄이기 위한 절사형 승산기에 대한 관심이 높아지고 있다.

2의 보수 승산을 위한 절사형 Booth 승산기의 절사 오차를 최소화하기 위한 오차보상 방법을 제안하고, 이를 적용하여 설계된 승산기의 절사오차와 면적을 분석하였다. 문헌 [4]의 절사 승산기와 비교하여 평균 오차가 약 60% 정도 감소하며, 절사되지 않은 일반 승산기에 비해 약 35%의 면적 감소가 얻어진다. 제안된 방법을 적용하여 16-비트×16-비트 절사형 승산기를 0.35- μm CMOS 공정을 이용하여 full-custom 방식으로 설계하였으며, 동작속도에 따른 전력소모 특성을 분석하였다. 설계된 절사형 Booth 승산기는 3.3-V 전원전압에서 200-MHz로 동작 가능하며, 약 20-mW의 전력소모 특성을 갖는다. 제안된 방법은 승산기의 비트 수가 클수록 기존의 방법에 비해 평균 오차 및 면적이 감소하므로 비트 수가 큰 절사형 승산기의 설계에도 적합할 것으로 평가되며, FIR 필터, IIR 필터, 등화기 등 고정 비트 크기를 갖는 디지털 신호처리 회로의 면적감소와 저전력 설계에 폭넓게 적용 가능하다.

감사의 글

반도체설계교육센터(IDECC)의 CAD Tool 지원에 의한 연구결과의 일부임.

참고 문헌

[1] A.D. Booth, "A signed binary multiplication technique", Quarterly J. Mechanics, Appl. Math, vol. 4, Part 2, pp. 236-240, 1951.

[2] L.P. Rubinfield, "A proof of the modified Booth's algorithm for multiplication", IEEE Trans. on Computers, vol. C-24, no. 10, pp. 1014-1015, Oct. 1975.

[3] G.K. Ma and F.J. Taylor, "Multiplier policies for digital signal processing", IEEE ASSP Magazine, pp. 6-20, Jan. 1990.

[4] S.S. Kidambi, F. El-Guibaly and A. Antoniou, "Area-efficient multipliers for digital signal processing applications", IEEE Trans. on CAS-II, vol. 43, no. 2, pp. 90-95, Feb. 1996.

[5] 이광현, 임종석, "저전력 설계를 위한 절단된 Booth 곱셈기 구조", 대한전자공학회 논문지, vol. 37, SD-9, pp. 55-64, Set. 2000.

[6] J.M. Jou, S.R. Kuang, and R.D. Chen, "Design of low-error fixed-width multipliers for DSP application", IEEE Trans. on CAS-II, vol. 46, no. 6, pp. 836-842, Jun. 1999.

[7] A.Y. Kentus, H.T. Hung and A.N. Willson Jr., "An architecture for high-performance/small-area multipliers for use in digital filtering applications", IEEE Journal of Solid State Circuits, vol. 29, pp. 117-121, Feb. 1994.

저자 소개



정해현(Hae-Hyun Jung)

2001년 2월 금오공과대학교 전자공학과 졸업
2001년 3월~현재 금오공과대학교 전자공학과 석사과정

※관심분야 : 통신 및 신호처리용 집적회로 설계



박종화(Jong-Wha Park)

1999년 2월 금오공과대학교 전자공학과 졸업
2001년 2월 금오공과대학교 전자공학과 공학석사

2001년 3월~현재 (주)서두로직 EDA 사업부

※관심분야 : 집적회로 설계, CAD Tool



신경욱(Kyung-Wook Shin)

1984년 2월 한국항공대학교 전자공학과 공학사
1986년 2월 연세대학교 대학원 전자공학과 공학석사
1990년 8월 연세대학교 대학원 전자공학과 공학박사

1990년 9월~1991년 6월 한국전자통신연구소

1995. 8월~1996. 7 Univ. of Illinois at UC 방문연구

1991년 7월~현재 금오공과대학교 전자공학부부교수

※관심분야 : 통신 및 신호처리용 집적회로 설계, 암호 프로세서 설계, 반도체 IP 설계