論文95-32A-5-8

멀티비트 리코딩 병렬 승산기의 최적설계를 위한 면적-시간 복잡도 분석

(Area-Time Complexity Analysis for Optimal Design of Multibit Recoding Parallel Multiplier)

金 得 慶 *, 辛 卿 旭 *, 李 溶 錫 **, 李 文 基 **

(Deuk-Kyung Kim, Kyung-Wook Shin, Yong-Surk Lee, and Moon-Key Lee)

요 약

고속 승산기의 설계에 있어서 핵심이 되는 요소는 부분곱들을 빠르게 가산할 수 있는 방법과 회로구조를 찾는 것이다. 본 논문에서는 수정형 Booth 알고리듬을 일반화시킨 멀티비트 리코딩 병렬 승산기 (Multibit Recoding Parallel Multiplier; MRPM)의 VLSI 구현시 최적설계를 위한 면적-시간 복잡도에 대한 이론적인 모델링을 제안하였다. 멀티비트 리코딩 병렬 승산기를 구성하는 각 기능블럭에 대해 리코딩 비트수와 승산기 비트수의 변화에 따른 게이트 복잡도와 지연시간 특성을 모델링 하였으며, 이를 토대로 전체 승산기의 면적, 지연시간 및 효율을 시뮬레이션 하였다. 시뮬레이션 결과에 의하면, 승수와 피승수의 비트수가 작은 경우에는 리코딩 크기가 3-비트인 기존의 수정형 Booth 알고리듬이 최적의 효율을 갖으나, 승수와 피승수의 비트수가 큰 경우에는 리코딩 크기 4-비트에서 최적의 효율을 갖는 것을 보였다.

Abstract

The usual approach for designing a fast multiplier involves finding a way to quickly add up all the partial products, based on partial product recoding scheme and carry-save addition. This paper describes theoretical models for area and time complexities of Multibit Recoding Parallel Multiplier (MRPM), which is a generalization of the modified Booth recoding scheme. Based on the proposed models, time performance, hardware requirements and area-time efficiency are analyzed in order to determine optimal recoding size for very large scale integration (VLSI) realization of the MRPM. Some simulation results show that the MRPM with large multiplier and multiplicand size has optimal area-time efficiency at the recoding size of 4-bit.

* 正會員, 金烏工科大學校 電子工學科
(Dept. of Electronic Eng., Kumoh National University of Technology)

** 正會員, 延世大學校 電子工學科
(Dept. of Electronic Eng., Yonsei University)

接受日字: 1994年7月30日, 수정완료일: 1995年4月28日

I. 서 론

컴퓨터를 비롯한 디지탈 통신 및 신호처리 기술의 급속한 발전에 따라 대량의 데이타를 실시간 처리할 수 있는 고성능 연산장치의 필요성이 증대되고 있으며, 이의 구현을 위해서는 VLSI (Very Large Scale Integration) 기술을 바탕으로한 고성능 병렬 승산기 및 가산기 회로의 설계기술이 필수적이다.

승산기는 컴퓨터의 부동소숫점 처리기, 통신 및 디지탈 신호처리 프로세서 등에서 필수적으로 사용되며, 전체 시스템의 성능에 커다란 영향을 미치는 핵심 연산장치이다. 일반적으로, 이진수의 승산은 부분곱들에 대한 일련의 가산과정을 통해 이루어진다. 따라서, 고속 승산기의 설계에 있어서 핵심이 되는 요소는 부분곱들을 빠르게 가산할 수 있는 방법과 회로구조를 찾는 것이다. 고속 승산기 설계에 적합한 여러가지 병렬승산 방식들이 제안되고 있으며, 이들은 승수 리코딩 기법, carry-save 가산기법, 부호확장 제거기법 등을 이용하고 있다[1-3,7-8]. 특히, 병렬 승산기의 면적과 속도를 줄이기 위한 방안으로서, 승수를 2-비트로 리코딩하여 부분곱 행의 수를 감소시키는 Booth 알고리듬[4]과 이를 개선하여 승수를 3-비트씩 리코딩하는 수정형 Booth 알고리듬(Modified Booth Algorithm)[5]이 제안되어 널리 이용되고 있다.

병렬 승산기의 설계에서 가장 일반적으로 사용되고 있는 수정형 Booth 알고리듬의 경우, 승수 (피승수)의 크기가 작은 경우에는 적합하나, 승수 (피승수)의 크기가 커질 수 록 전체 승산기의 면적-시간 성능이 최적값을 갖지 못한다. 최근, 리코딩 크기를 3-비트 이상으로 하여 수정형 Booth 알고리듬을 일반화시킨 멀티비트 리코딩 병렬 승산기 알고리듬이 발표된 바 있다[6]. 리코딩 크기가 3-비트 이상인 경우에는 부분곱 행의 수가 더욱 감소하여 승산속도가 개선됨과 아울러 승산기내의 carry-save adder (CSA) 배열이 차지하는 면적이 감소하는 장점을 갖는다. 반면에, CSA 배열에서 가산될 피승수의 multiple 종류가 많아지므로 이들을 생성하고 선택하기 위한 추가적인 하드웨어 요구량이 증가하고, 또한 이들 회로에 의한 지연시간도 증가하게 된다. 따라서, 리코딩 비트수, 승수 (피승수)의 비트수, 그리고 multiple 발생을 위한 회로방식 등에따라 멀티비트 리코딩 병렬 승산기의 면적-시간 성능이 영향을 받게되며, 최적의 면적-시간 성능을 갖는 승산기의 설계를 위해서는 이들에 대한 종합적인 분석이 수반되어야 한다.

본 논문에서는 최적의 면적-시간 성능을 갖는 멀티비트 리코딩 병렬 승산기의 설계를 위해 리코딩 비트수, 승수와 피승수의 비트수 등에 따른 면적과 시간 성능을 이론적으로 모델링하고 시뮬레이션하였으며, 이를 토대로 최적의 면적-시간 효율을 얻기위한 리코딩 비트수를 분석하였다.

II. 멀티비트 리코딩 병렬 승산기 구조 및 승산방식

병렬 승산기의 설계에 많이 이용되고 있는 수정형 Booth 알고리듬은 승수를 3-비트씩 리코딩하여 부분곱의 수를 반으로 줄임으로써 승산속도와 면적을 동시에 감소시키는 효율적인 방법이다. 최근에, 승수와 피승수의 비트 수가 큰 경우에 승산기의 성능을 극대화하기 위한 방법으로 승수에 대한 리코딩 크기를 S-비트 (단, S>3) 로 확장하여 부분곱의 수를 더욱 줄이고자 하는 멀티비트 리코딩 병렬승산 알고리듬이 제안된 바 있다.

n-비트의 승수 X와 q-비트의 피승수 Y를 리코딩 비트수 S (단, $S \geq 3$)로 멀티비트 리코딩하여 병렬 승산하는 과정은 다음과 같다.[6] 단, 승수 X와 피승수 Y가 sign-magnitude 형태로 표현된 경우에는 부호교정을 위한 추가적인 처리가 필요하며, 아래의 과정중 밑줄친 부분은 sign-magnitude 입력을 갖는 승산기에만 필요하다.

(a) 승수와 피승수의 부호 비트를 '0'으로 치환한다. (sign-magnitude 표현인 경우)

(b) 승수 X의 비트 수 n이 (S-1)에 의해서 나누어 질 수 있도록 X의 부호 비트를 최대 (S-2)- 비트 확장시킨다.

(c) 승수 X의 최하위 비트 오른쪽에 '0'을 추가시키고, 이를 x_{-1}이라고 한다.

(d) x_{-1}로 부터 시작하여 S-비트씩 그룹을 형성하고, 이를 그룹 G_i라고 한다.

단, 서로 인접한 그룹들이 1 비트씩을 공유하도록 그룹을 형성한다.

(e) S-비트로 구성된 그룹 G_i는 식(1)에 의해서 각각의 signed-digit(SD) D_i를 생성한다.

$$D_i = G_i \cdot K \tag{1}$$

단, $G_i = [\, x_{(S-1)(i+1)-1} \;\; x_{(S-1)(i+1)-2} \cdots x_{(S-1)i} \;\; x_{(S-1)i-1}]$

$K = [\, -2^{(S-2)} \;\; 2^{(S-3)} \; 2^{(S-4)} \cdots 2^1 \; 2^0 \; 1]^T$

$i = 0, 1, 2, \cdots, \frac{n}{S-1} - 1$

(f) D_i와 피승수 Y의 곱으로 생성된 multiple을 $2^{(S-1)i}$만큼 쉬프팅하여 i-번째 부분곱을 만들고, 이들을 CSA 배열에서 가산하여 식(2)의 최종결과를 얻는다.

단, sign-magnitude 표현인 경우에는 부호비트를 식(3)과 같이 처리한다.

$$P = X \cdot Y = \sum_{i=0}^{n/(S-1)-1} Y \cdot D_i \cdot 2^{(S-1)i} = -p_{n+q-1}2^{n+q-1} + \sum_{i=0}^{n+q-2} p_i 2^i \quad (2)$$

$$-P_{n+q-1} = x_{n-1} \oplus y_{q-1} \quad (3)$$

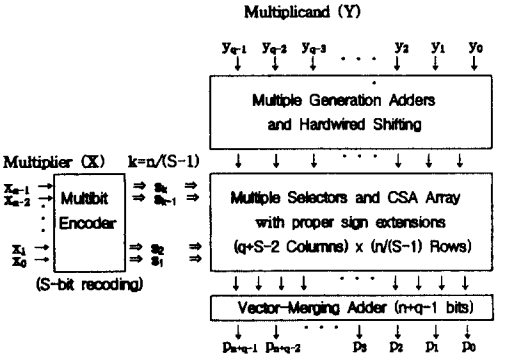


그림 1. S-비트 리코딩 병렬 승산기 구조
Fig. 1. Architecture of S-bit recoding parallel multiplier.

S-비트 (단, $S \geq 3$) 리코딩을 이용한 멀티비트 리코딩 병렬 승산기의 구조는 그림 1과 같으며, 엔코더, multiple 생성기 및 선택기, 그리고 CSA 배열과 벡터병합 가산기 등으로 구성된다. 엔코더는 승수 X를 S-비트씩 멀티비트 리코딩하여 bit set $[sign\,0x\,1x\,2x\cdots 2^{S-2}x]$ 를 생성하며, 생성된 bit set 신호는 피승수 Y의 multiple $(0, \pm Y, \pm 2Y, \pm 3Y, \cdots, \pm 2^{S-2}Y)$ 중 하나를 선택하기 위한 제어신호로 사용된다. multiple 생성기 블럭은 피승수 Y를 받아 이의 multiple들을 생성하며, multiple의 종류는 2의 누승형태 (즉, $2^k \cdot Y$, 단, k는 양의 정수)와 그 외의 multiple들로 구분할 수 있다. 2의 누승형태 multiple $(2Y, 4Y, 8Y, \cdots)$ 들은 단순 쉬프팅에 의해 생성되며, 그 이외의 multiple들은 2개 이상의 multiple들의 가산, 감산 및 쉬프팅 연산에 의해 생성된다. multiple 선택기 블럭은 부분곱 생성을 위해 multiple중 하나를 선택하는 기능을 수행한다. CSA 배열은 생성된 부분곱들을 가산하며, 그 결과로 생성된 합벡터와 캐리벡터는 벡터병합 가산기를 거쳐 최종 승산결과를 출력한다.

일반적으로, 멀티비트 리코딩 병렬 승산기에서 리코딩 비트수 S가 커질 수 록, 부분곱의 수가 n/(S-1)로 감소하여 CSA 배열의 가산기 면적과 지연시간이 감소되는 장점을 갖는 반면에, 피승수 Y의 multiple 종류가 많아지므로 multiple 생성 및 선택기 회로가 복잡해지는 trade-off 관계를 갖는다. 결론적으로, 전체 승산기의 면적-시간 성능은 리코딩 비트수 S, 승수와 피승수의 비트 수 등에 따라 영향을 받으며, 따라서 최적의 면적-시간 성능을 갖는 멀티비트 리코딩 병렬 승산기의 설계를 위해서는 승수와 피승수의 비트수에 따른 적절한 리코딩 비트수의 결정이 중요하다.

Ⅲ. 멀티비트 리코딩 병렬 승산기의 면적-시간 복잡도 분석

본 장에서는 멀티비트 리코딩 병렬승산기의 각 기능 블럭이 갖는 면적과 지연시간 복잡도를 이론적으로 모델링하고, 이를 토대로 최적의 면적-시간 성능을 갖는 승산기 설계의 조건을 분석하고자 한다. 본 논문에서는 승산기의 면적과 지연시간을 모델링함에 있어서 공정 파라메터에 독립적인 모델식을 제안하고자 하며, 이를 위해 2-입력 게이트의 면적 A_{gate}와 지연시간 T_{gate}를 기본단위로 하여 승산기의 게이트 복잡도와 지연시간을 모델링하였다.

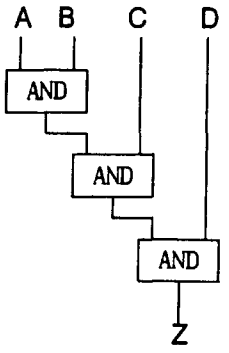
1. 지연시간 모델링

본 절에서는 멀티비트 리코딩 병렬승산기를 설계함에 있어서 리코딩 비트수 S가 승산기의 지연시간에 미치는 영향을 이론적으로 모델링한다.

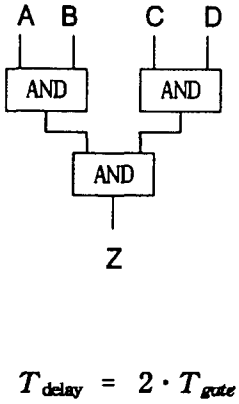
본 논문에서는 임의의 회로가 갖는 지연시간은 회로깊이 (circuit depth)에 비례한다는 가정을 토대로 모델링하였으며, 2-입력 게이트의 지연시간 Tgate를 지연시간의 기본단위로 사용하여 normalize 하였다. 한편, 부울린 함수로 표현된 임의의 논리식을 회로로 구현함에 있어서 회로깊이는 그림 2와 같이 회로의 구현방법에 따라 달라질 수 있다. 예를 들어, 부울린 함수 $Z = A \cdot B \cdot C \cdot D$ 에 대한 회로구현의 경우, 그림 2-(a)는 계단형 구현의 경우로서 회로깊이는 3이며, 그림 2-(b)는 이진나무(binary tree)형 구현의 경우로서 회로깊이는 2가 된다. 본 논문에서는 이진나무형 구조의 회로구현에 대한 회로깊이를 사용하였다.

가) 멀티비트 엔코더 블럭의 지연시간

엔코딩 블럭은 승수 X를 입력받아 이를 S-비트씩 그룹지어 리코딩을 수행하며, 그림 3과 같이 n/(S-1)개의 단위셀로 구성된다.



(a) Stair-like Structure



(b) binary tree structure

그림 2. $Z = A \cdot B \cdot C \cdot D$ 회로의 지연시간 모델링
Fig. 2. Delay modeling for a circuit of $Z = A \cdot B \cdot C \cdot D$

각각의 단위셀들은 multiple 선택기 블럭에서 사용될 선택신호인 bit set $[sign\ 0x\ 1x\ 2x \cdots 2^{S-2}x]$ 를 생성한다. 리코딩 비트수 S에 대한 단위 엔코더 셀의 진리표로 부터 부울린 함수를 추출하여 회로깊이를 구하면, S=3-비트인 경우의 회로깊이는 3이고, S>4인 경우의 회로깊이는 $T_{enc} = \lceil \log_2 S \rceil + 2$가 된다. n/(S-1)개의 단위 엔코더 셀들이 병렬동작하므로 전체 엔코더 블럭의 지연시간 T_{enc}은 승수의 비트수에 무관하며, 식(4)와 같이 모델링된다.

$$T_{enc} = 3 \quad (\text{단, } S=3) \tag{4-1}$$

$$T_{enc} = \lceil \log_2 S \rceil + 2 \quad (\text{단, } S>3) \tag{4-2}$$

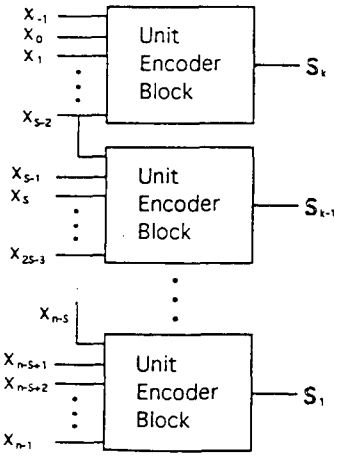


그림 3. 멀티비트 엔코더의 구성도
Fig. 3. Block diagram of multibit encoder.

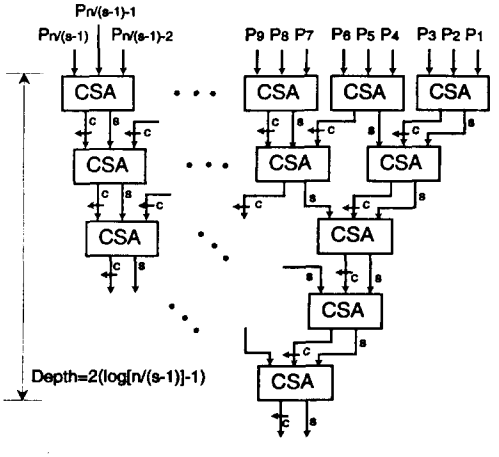


그림 4. Wallace tree 구조를 이용한 CSA 배열
Fig. 4. CSA array using Wallace tree structure.

나) CSA 배열의 지연시간

Carry-Save Adder(CSA) 배열은 부분곱들을 가산하여 합벡터와 캐리벡터를 생성하는 블럭이며, 승산기의 면적과 속도에 큰 영향을 미친다. CSA 배열은 전가산기의 2차원 배열[9], Wallace tree[10] 등 여러 가지 구조로 구현이 가능하다. 전가산기의 2차원 배열을 이용하는 경우는 규칙적인 구조의 장점을 갖는 반면에 배열의 깊이가 부분곱의 갯수에 선형적으로 비례하므로 속도가 느리다는 단점을 갖으며, 비트단위의 파이프라인을 통해 속도를 향상시킬 수 있는 분야에 이용된다. 한편, Wallace tree 구조는 부분곱을 고속으

로 가산할 수 있도록 고안된 것으로 높은 병렬성에 의해 승산속도가 빠르다는 장점을 갖으며, 구조적인 불규칙성이 단점으로 인식되고 있다. 본 논문에서는 그림 4와 같이 Wallace tree 구조로 구현하는 경우에 대한 지연시간을 모델링하였다. 그림에서 알 수 있는 바와 같이, CSA 배열의 지연시간 T_{csa}은 Wallace tree의 tree 깊이에 의해 결정되며, 따라서 승수의 비트수 n과 리코딩 비트수 S에 따른 T_{csa}는 식(5)와 같이 모델링된다. 식(5)에서 n/(S-1)은 n-비트 승수를 S-비트씩 리코딩하는 경우에 CSA 배열에서 가산되는 부분곱의 갯수를 나타내며, T_{FA}는 1-비트 carry-save 전가산기의 지연시간을 나타낸다.

$$T_{csa} = 2T_{FA} \cdot [\log_2(\frac{n}{S-1})-1] \quad (\text{단, } S \geq 3) \quad (5)$$

다) Multiple 선택기 블럭의 지연시간

Multiple 선택기는 엔코더의 출력인 bit set $[sign\ 0x\ 1x\ 2x\ \cdots\ 2^{S-2}x]$를 받아 피승수 Y의 multiple $(Y, 2Y, 3Y, \cdots, 2^{S-2}Y)$ 중 하나를 선택하여 부분곱을 생성하는 블럭이다.

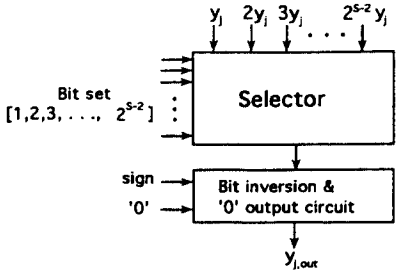


그림 5. 단위 multiple 선택기의 구성도
Fig. 5. Block diagram of unit multiple selector.

음의 부분곱을 생성하기 위해 $sign$='1'인 경우에는 선택된 multiple을 비트반전 (bit inversion) 시키며, $0x$='1'인 경우에는 무조건 0 (즉, $0 \cdot Y$)을 출력한다. Multiple 선택기 블럭은 그림 5와 같은 1-비트 선택기 셀의 배열로 구성되며, 리코딩 비트수 S가 증가함에 따라 multiple들의 종류가 많아지므로 단위 셀의 회로 복잡도도 증가한다. 표(1)은 리코딩 비트수 S에 따른 단위 셀의 회로깊이를 나타낸 것이며, 회로깊이가 S에 선형적으로 증가함을 알 수 있다. 그림 5의 단위 셀들은 CSA 배열내에서 행단위로 병렬동작하므로 전체 multiple 선택기 블럭의 지연시간 T_{sel}은 식(6)과 같이 모델링된다.

$$T_{sel} = \log_2 2^{(S-1)} + 3 = S + 2 \quad (\text{단, } S \geq 3) \quad (6)$$

표 1. 단위 선택기 셀의 회로깊이
Table 1. Circuit depth of the unit selector cell.

리코딩 크기 (S)	Multiple 갯수	선택기의 회로깊이	비트반전 & '0' 출력단회로깊이
3	2	2	3
4	4	3	3
5	8	4	3
6	16	5	3
.	.	.	.
.	.	.	.
S	2^{S-2}	S-1	3

라) 벡터병합 가산기 블럭의 지연시간

벡터병합 가산기는 CSA 배열의 출력인 합벡터와 캐리벡터를 가산하여 최종 승산결과를 생성하는 블럭이며, 승수와 피승수의 비트 수가 각각 n과 q인 경우에 (n+q-1)-비트의 가산기로 구성된다. n, q가 큰 경우에는 가산기 비트수가 매우 커지게되므로 승산속도를 향상시키기 위해서는 고속 가산기의 사용이 필수적이다. 벡터병합 가산기의 구현은 BLC (Binary Look-ahead Carry), Carry-select, Carry-skip 등[11] 여러가지 구조가 사용될 수 있으며, 본 논문에서는 BLC 구조로 구현하는 경우에 대해 지연시간을 모델링하였다. BLC 가산기는 carry-lookahead 회로가 이진트리 구조이므로 비트수가 큰 경우에도 고속 가산이 가능하며, 또한 구조적인 규칙성이 뛰어나 설계가 용이하다는 특징을 갖는다. Brent-Ewin이 제안한 BLC 가산기 생성 알고리듬[12]을 이용하여 구한 carry-lookahead 회로의 트리깊이는 $\log_2(n+q-1)$이며, 따라서 벡터병합 가산기의 지연시간 Tvma는 식(7)과 같이 모델링된다.

$$T_{vma} = \log_2(n+q-1) + T_{G,S} \quad (7)$$

단, $T_{G,S}$: BLC 가산기의 P_i, G_i 셀과 sum 발생 셀의 회로깊이

마) multiple 생성기 블럭의 지연시간

multiple 생성기는 CSA 배열에서 가산될 피승수 Y의 multiple $(Y, 2Y, 3Y, \cdots, 2^{S-2}Y)$를 생성하는 블럭이다. 2의 누승형태의 multiple들은 단순한 쉬프팅에 의해 생성되며, 그 이외의 multiple들은 2개 이상의 multiple들의 가산 또는 감산에 의해 생성된다. multiple 생성시 가산 또는 감산의 operand 수가 3 이상인 경우 (예를 들면, $43Y = 32Y + 8Y + 2Y + Y$) 에는 CSA 구조를 사용하여 지연시간을 단축시켜야하며, 이 경우에 multiple 생성블럭의 지연시간은 CSA 배열의

지연시간과 벡터병합 가산기 지연시간의 합으로 나타난다. 본 논문에서는 리코딩 비트수 S에 따른 모든 multiple들의 조합을 구한 후, 이들을 생성하기 위해 필요한 가산기의 비트 수와 CSA 배열의 깊이를 구함으로써 multiple 발생기의 지연시간 T_{odg}를 식(8)과 같이 모델링하였다. 식(8)에서 $S+q-2$는 multiple의 생성에 필요한 가산기의 비트 수이며, $\alpha \cdot T_{FA}$는 가산 또는 감산의 operand 수가 3이상인 경우에 필요한 CSA 배열의 지연시간을 나타낸다. 한편, 리코딩 비트수 S에 따른 CSA 배열의 깊이 α는 표 2와 같이 주어지며, 벡터병합 가산기는 BLC 가산기를 사용하는 것으로 가정하여 모델링하였다.

$$T_{odg} = \log_2(S+q-2) + T_{G,S} + \alpha \cdot T_{FA} \quad (\text{단, } S>3) \quad (8)$$

표 2. Multiple 생성블럭의 CSA 배열깊이 (α)
Table 2. CSA array depth (α) in the multiple generation block.

리코딩 크기 S	2	3	4	5	6	7	8	9
α	0	0	0	0	1	1	2	3

바) 전체 지연시간

멀티비트 리코딩 병렬승산기의 전체 지연시간은 앞에서 모델링된 각 기능블럭이 갖는 지연시간의 합으로 나타나며, 승산과정중 multiple 생성과 멀티비트 리코딩은 동시에 이루어지므로 이들의 지연시간중 큰 것 ($S>3$인 경우, $T_{enc} < T_{odg}$) 만이 전체 지연시간에 영향을 미친다. 전체 지연시간은 식(5)-식(8)의 합으로 식(9)와 같이 모델링된다.

$$T_{total} = T_{odg} + T_{sel} + T_{csa} + T_{vma} \quad (\text{단, } S>3) \quad (9)$$

2. 면적 모델링

본 절에서는 리코딩 비트수 S가 승산기의 면적에 미치는 영향을 모델링한다. 본 논문에서는 승산기의 면적을 기능블럭들의 게이트 복잡도를 기준으로 모델링하였으며, 2-입력 게이트의 면적 A_{gate}를 면적의 기본단위로 사용하여 normalize 하였다.

가) 멀티비트 엔코더 블럭의 게이트 복잡도

엔코딩 블럭은 그림 3과 같이 n/(S-1) 개의 단위 엔코더 셀로 구성되므로, 단위 셀의 게이트 수를 알면 쉽게 모델링할 수 있다. 단위 엔코더 셀의 진리표에서 구해진 게이트 복잡도는 표 3과 같으며, 따라서 엔코더 블럭의 게이트 복잡도 A_{enc}는 식 (10)과 같이 모델링될 수 있다.

$$A_{enc} = ((2^S-2)\cdot S - 2^{S-2} - 3)\cdot \frac{n}{S-1} \quad (\text{단, } S\geq 3) \quad (10)$$

표 3. 단위 엔코더 셀의 게이트 복잡도
Table 3. Gate complexity of the unit encoder cell.

엔코더 출력	OR gate 수	AND gate 수
Sign	0	0
0x, 2^{S-2}x	2	4(S-1)
2^{S-3}x	1	2(S-2)
나머지 출력	$3(2^{S-2}-2)$	$4(S-1)(2^{S-2}-2)$

나) CSA 배열의 게이트 복잡도

CSA 배열은 그림 4와 같이 Wallace tree 구조로 구현되므로, Wallace tree를 구성하는 CSA 셀의 갯수와 가산될 부분곱의 비트 수로 부터 게이트 복잡도를 모델링할 수 있다. S-비트 리코딩에서 부분곱의 최대 비트 수는 q+S-2이고 Wallace tree는 n/(S-1)-2개의 CSA 셀로 구성되므로, 전체 게이트 복잡도는 식(11)과 같이 모델링될 수 있다. 식(11)에서 A_{FA}는 1-비트 CSA 가산기의 게이트 수를 나타낸다.

$$A_{csa} = (\frac{n}{S-1} - 2)\cdot(q+S-2)\cdot A_{FA} \quad (\text{단, } S\geq 3) \quad (11)$$

다) Multiple 선택기 블럭의 게이트 복잡도

Multiple 선택기는 그림 5의 단위비트 선택기 셀이 부분곱의 갯수와 비트 수 만큼 배열되어 구성된다. 리코딩 비트수 S에 따른 단위 셀의 게이트 복잡도는 표 4와 같이 모델링되며, 부분곱의 갯수는 n/(S-1)개이고 비트 수는 q+S-2 이므로 게이트 복잡도 A_{sel}는 식(12)와 같이 모델링된다.

$$A_{sel} = (4+2^{S-1})\cdot(\frac{n}{S-1})\cdot(q+S-2) \quad (\text{단, } S\geq 3) \quad (12)$$

표 4. 단위 선택기 셀의 게이트 복잡도
Table 4. Gate complexity of the unit selector cell.

	OR gate 수	AND gate 수
비트 반전단 & '0' 출력단	1	4
Gating 회로	$2^{S-2}-1$	2^{S-2}

라) 벡터병합 가산기 블럭의 게이트 복잡도

1절에서 언급된 바와 같이, 본 논문에서는 BLC 구조를 이용하여 벡터병합 가산기를 구현하는 것으로 가정한다. BLC 가산기는 carry-lookahead 회로의 이진트리 부분과 P_i, G_i 생성셀 및 sum 생성셀들로 구

성되고, 이진트리 부분은 BLACK_cell들의 이진트리와 WHITE_Cell들의 역이진트리로 구성된다[12]. 따라서, 트리를 구성하는 셀들의 갯수와 각 셀들의 게이트 수로부터 (n+q-1)-비트 벡터병합 가산기의 게이트 복잡도 Avma는 식(13)과 같이 모델링된다.

$$A_{vma}=4(2n+2q-3)+2(n+q-2)\cdot\log_2(n+q-1) \quad (13)$$

마) Multiple 생성기 블럭의 게이트 복잡도

1절의 지연시간 모델링에서 언급된 바와 같이, 2의 누승형 이외의 multiple 들은 2개 이상의 multiple들의 가산 또는 감산에 의해 생성되고, 가산 또는 감산의 operand 수가 3이상인 경우에는 CSA 구조를 사용하여 속도를 향상시킨다.

표 5. 식(14)에 사용된 γ, η, β값
Table 5. γ, η, β values for Eq.(14).

리코딩 크기 S	γ	η	β
3	0	0	0
4	0	0	1
5	0	0	3
6	4	9	7
7	16	49	15
8	48	170	31
9	125	508	63

따라서, 3개 이하의 multiple들의 가산 또는 감산으로 생성되는 경우에는 벡터병합 가산기만 필요하고, 3개 이상의 multiple들의 가산 또는 감산에 의해 multiple이 생성되는 경우에는 벡터병합 가산기와 CSA 배열이 필요하다. 리코딩 비트수 S에 따른 모든 multiple들의 조합을 구한 후, 이들을 생성하기 위해 필요한 벡터병합 가산기의 비트 수 (S+q-3)와 CSA 배열의 가산기 수로 부터 전체 multiple 생성기의 게이트 복잡도 A_{mlg}를 식(14)와 같이 모델링하였다. 식(14)에서 리코딩 비트수 S에 따른 γ, η, β 값들은 표 5와 같이 주어진다.

$$A_{mlg}=\beta\cdot[4(2q+2S-7)+2(q+S-4)\cdot\log_2(q+S-3)]+(\gamma q+\eta)\cdot A_{FA} \quad (14)$$

바) 전체 면적

승산기의 전체면적은 식(10) - 식(14)의 각 기능블럭 면적의 합으로 다음과 같이 된다.

$$A_{total}=A_{enc}+A_{csa}+A_{sel}+A_{vma}+A_{mlg} \quad (\text{단, } S>3) \quad (15)$$

3. 시뮬레이션 및 결과 고찰

앞절에서 제시된 멀티비트 리코딩 병렬승산기의 게이트 복잡도와 지연시간에 대한 모델링을 토대로 시뮬레이션을 수행하였다. 모델식에 포함된 전가산기의 면적 (A_{FA})과 지연시간 (T_{FA})은 각각 전가산기의 부울린 식으로 부터 얻어진 $6A_{gate}$와 $4T_{gate}$를 사용하였으며, 시뮬레이션시에는 이 값들을 각각 T_{gate}와 A_{gate}로 normalize 하여 사용하였다.

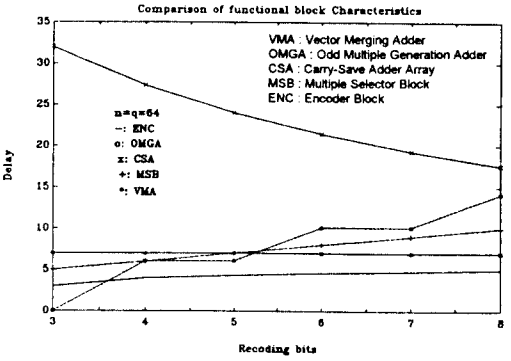


그림 6. MRPM의 각 블럭의 지연시간 특성
Fig. 6. Comparison of delay times for functional blocks within MRPM.

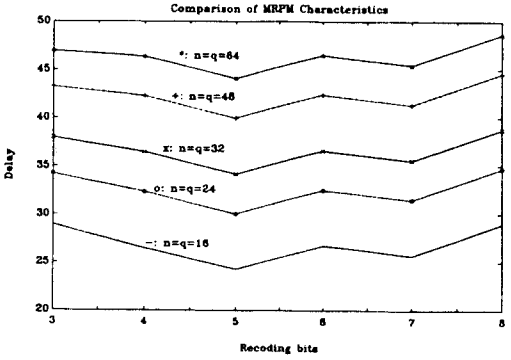


그림 7. 승산기 비트수 및 리코딩 크기에 따른 MRPM의 전체 지연시간 특성
Fig. 7. Total delay characteristics of MRPM as a function of multiplier & recoding size.

그림 6은 리코딩 비트수 S의 변화에 대한 승산기 블럭들의 지연시간 변화를 시뮬레이션한 결과이다. 리코딩 비트수가 증가할 수 록 CSA 배열의 지연시간이 감소하는 반면에, multiple 생성기 회로의 지연시간이

증가하며, 그외의 블럭들은 지연시간 변화가 매우 작음을 알 수 있다. 그림 7은 리코딩 비트수와 승수와 피승수의 비트 수에 따른 승산기의 전체 지연시간 변화를 시뮬레이션한 결과이며, 리코딩 크기 5-비트에서 최소의 지연시간을 갖음을 알 수 있다.

그림 8은 리코딩 비트수 S의 변화에 대한 승산기 각 블럭들의 게이트 복잡도 변화를 시뮬레이션한 결과이다. $S \le 5$인 영역에서는 S의 증가에 따른 게이트 복잡도의 증가율이 작고, CSA 배열과 multiple 선택기 블럭이 대부분을 차지한다. 반면에, $S \ge 6$인 영역에서는 multiple 생성기와 multiple 선택기 블럭의 게이트 복잡도가 급격히 증가함을 알 수 있다.

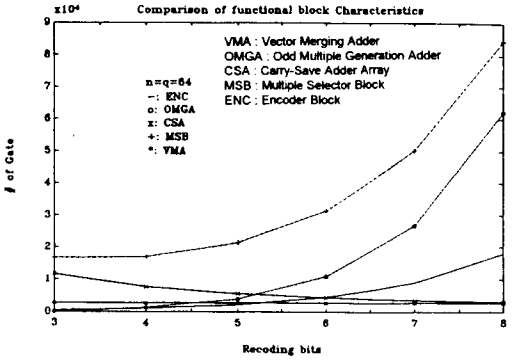


그림 8. MRPM의 블럭별 면적 비교
Fig. 8. Comparison of areas for functional blocks within MRPM.

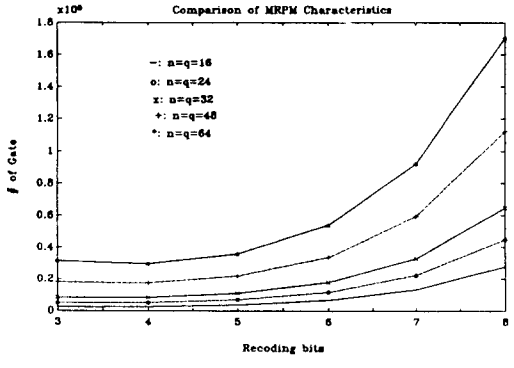


그림 9. 승산기 비트수 및 리코딩 크기에 따른 MRPM의 면적 변화
Fig. 9. Total area of MRPM as a function of multiplier & recoding size.

그림 9는 리코딩 비트수와 승산기 비트수에 따른 승산기 전체의 게이트 복잡도 변화를 시뮬레이션한 결과이다. $S \le 5$인 영역에서는 게이트 복잡도의 증가율이 매우 작은 반면에, $S \ge 6$인 영역에서는 매우 급격히 증가함을 알 수 있다. 그림 10은 리코딩 비트수와 승산기 비트 수에 따른 효율 $1/(A \cdot T)$을 시뮬레이션한 결과이다. 승산기의 비트 수가 작은 경우에는 리코딩 크기 3-비트 (즉, 기존의 수정형 Booth 알고리듬)과 4-비트에서 비슷한 효율을 갖는 반면에, 승산기의 비트 수가 클 수 록 리코딩 크기 4-비트에서 최대의 효율을 갖는다. 또한, 리코딩 크기가 5-비트 이상이 되면 승산기의 효율이 급격히 감소함을 알 수 있다. 이와 같은 시뮬레이션 결과를 토대로, 승수와 피승수의 비트 수가 큰 승산기를 설계하는 경우에는 4-비트 리코딩 방식으로 설계하는 것이 승산기 전체의 면적-시간 측면에서 최적의 효율을 갖는 다는 결론을 얻을 수 있다.

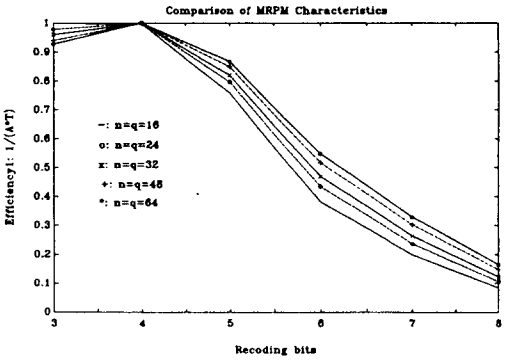


그림 10. 승산기 비트수에 따른 효율 $(1/A \cdot T)$ 특성
Fig. 10. Efficiency $(1/A \cdot T)$ of MRPM as a function of multiplier & recoding size.

Ⅳ. 결 론

본 논문에서는 승수를 3-비트 이상으로 리코딩하는 멀티비트 리코딩 병렬 승산기의 최적설계 조건을 분석하기 위한 면적-시간 복잡도를 모델링하였다. 승산기를 구성하는 각 기능블럭에 대해 리코딩 비트수와 승산기 비트 수의 변화에 따른 게이트 복잡도와 지연시간 특성을 모델링하였으며, 이를 토대로 전체 승산기의 면적, 지연시간 및 효율을 시뮬레이션 하였다. 시뮬레이션 결과에 의하면, 승수와 피승수의 비트수가 작은 경우에는 리코딩 크기 3-비트 (즉, 수정형 Booth 알고리듬)과 4-비트에서 비슷한 효율을 갖으나, 승산기 비트수가 큰 경우에는 리코딩 크기 4-비트에서 최적의 효율을 갖는 다는 결론을 얻었다. 또한, 면적보다는 승산속도가 중요한 요소인 경우에는 기존의 수정형 Booth 알고리듬보다 4-비트 리코딩 방식이 더욱 효율

적임을 확인하였다.

참 고 문 헌

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

[2] A. Y. Kwentus, 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, no. 2, pp.117-121, Feb. 1994.

[3] K. Hwang, *Computer Arithmetic : Principles*, Architecture and Design, John-Wiley & Sons, 1979

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

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

[6] H. Sam and A. Gupta, "A generalized multibit recoding two's complement binary numbers and its proof with application in multiplier implementations," *IEEE Trans. Computers*, vol. 39, no.8, pp.1006-1015, Aug. 1990.

[7] M. Ercegovac, T. Lang, "Fast multiplication without carry-propagate addition", IEEE Trans. Computers, vol.39, no.11, pp.1385-1389, Nov. 1990.

[8] S. Vassiliadis, E.M. Schwarz, B.M. Sung, "Hard-wired multipliers with encoded partial products", *IEEE Trans. Computers*, vol.41, no.11, pp.1181-1197, Nov. 1991.

[9] T.G. Noll, "Carry-save architectures for high-speed digital signal processing", *Journal of VLSI Signal Processing*, vol. 3, pp. 121-140, 1991.

[10] C.S. Wallace, "A suggestion for a fast multiplication", *IEEE Trans. on Electronic Computers*, pp. 14-17, Feb., 1964.

[11] N.H. Weste, K. Eshraghian, Principles of CMOS VLSI Design : *A Systems Perspective*, Addison-Wesley, 2nd Ed., 1993.

[12] R.P. Brent, R.R. Ewin, *"Design of an nMOS parallel adder"*, TR-CS-82-06, Dept. of Computer Science, The Australian National University, 1982.

저 자 소 개

金 得 慶(正會員)

1993년 2월 금오공과대학교 전자공학과 학사. 1995년 2월 금오공과대학교 전자공학과 공학석사. 1995년 3월 - 현재 C&S Tech. 연구원. 주관심 분야는 DSP용 집적회로, 마이크로 컨트롤러 설계등임

李 溶 錫(正會員)

1973년 연세대학교 전기공학과 학사. 1981년 University of Michigan. Ann Arbor 반도체 분야 박사학위수여. Device Engineering, Gate Array, Memory, ISDN, Telecommunication Chip, Microcontroller, RISC Microprocessor, Floating Point Unit, Cache Controller 등을 설계. Intel Corporation에서 Pentium 설계. 현재 연세대학교 전자공학과 부교수.

辛 卿 旭(正會員)

1984년 2월 한국항공대학교 전자공학과 학사. 1986년 2월 연세대학교 전자공학과 공학석사. 1990년 8월 연세대학교 전자공학과 공학박사. 1990년 9월 - 1991년 6월 한국전자통신연구소 반도체연구단 선임연구원. 1991년 7월 - 현재 금오공과대학교 전자공학과 조교수. 주관심 분야는 DSP용 집적회로 설계, 저전압/저전력 집적회로 설계 등임

李 文 基(正會員)

1965년 연세대학교 전기공학과 학사. 1967년 연세대학교 전기공학과 석사. 1973년 연세대학교 전기공학과 박사. 1980년 University of Oklahoma Ph.D. 1980년 - 1982년 한국전자기술연구소 (현 ETRI) 책임연구원. 1982년 - 현재 연세대학교 전자공학과 교수. 1989년 아식설계공동연구소 (RIAD) 설립. 1995년 대한전자공학회 회장. 주관심 분야는 마이크로 프로세서 및 DSP 집적회로 설계, 실리콘 압력센서 등임