승산시간 향상을 위한 병렬 승산기 어레이 설계에 관한 연구

이 강 현[†]

요 약

본 논문에서는 기존의 병렬 승산기 어레이에서 사용된 CSA(carry select adder)셀 구조를 수정하여 승산시간을 감소하는 새로운 병렬 승산기 어레이를 제안한다. MCSA(modified CSA)의 입력에 가수와 피가수가 자리올림보다 먼저 인가된다. 그리고 자리올림 전달 가산기를 위하여 DCSA(doubled inverted input CSA)를 설계하여 최종 승산항 다음에 추가한다. 제안된 안은 MCSA와 DCSA를 사용하여 설계하고 모의실험을 한다. 회로의 크기는 기존의 CSA셀을 사용한 기존의 승산기 어레이에 비하여 약 13% 증가했지만 연산시간은 약 52% 감소함을 확인하였다.

A Study on the Design of Parallel Multiplier Array
for the Multiplication Speed Up

Kang Hyeon Rhee[†]

ABSTRACT

In this paper, a new parallel multiplier array is proposed to reduce the multiplication time by modifying CSA(carry select adder) cell structure used in the conventional parallel multiplier array. It is named MCSA(modified CSA) that assignes the addend and augend to the inputs of CSA faster than Ci(carry input). Also the designed DCSA(doubled inverted input CSA) is appended after the last product term for the carry propagation adder. The proposed scheme is designed with MCSA and DCSA, and simulated. It is verified that the circuit size is increased about 13% compared with the conventional multiplier array with CSA cell but the operation time is reduced about 52%.

1. Introduction

There are many kinds of multiplication schemes which has been developed.

The basic schemes of multiplier include modified Booth's algorithm[1], Wallace tree [2], Dadda's scheme[3] and multiplier array [4]. The multiplier array can be implemented with modified Booth's algorithm or multi-bit recording techniques[5,6] which can reduce the number of product terms. The array mul-

tiplication scheme is widely used in high speed ALU because of its simplicity and regularity[7, 8].

In this paper, we designed the modified CSA(called MCSA) in order to reduce the operating time of conventional CSA. Also we designed DCSA(double inverted CSA) in order to reduce the operating time of the carry propagation adder. Here, MCSA is used in the multiplication term of designed multiplier array, DCSA is used as the carry propagation adder. In Section 2, it is described the operations of multiplier array with the conventional CSA and the multiplier array is designed with MCSA and DCSA in Section 3.

Then the operation of proposed multiplier array is evaluated and simulated in Section 4, and conclusions are drawn in Section 5.

2. Multiplier array with CSA

The operation of a CSA is same as a full adder. The characteristics of a full adder is described in ⟨Table 1, 2⟩. In ⟨Table 2⟩, we will assign the inputs of augend and addend as A and B. And the circuit of a full adder according to these tables can be designed in (Fig. 1).

⟨Table 1⟩ Carry propagation characteristics of a full adder

bit pair		characteristics about
addend	augend	carry propagation
0	0	carry generated (value=0)
0	1	carry in is propagated to carry out
1	0	carry in is propagated to carry out
1	1	carry generated (value=1)

⟨Table 2⟩ Sum generation characteristics of a full adder

carry	temporary	final sum
0	ST = A⊕B*	S = ST
1		S = ST'

* : A and B indicate the inputs of augend and addend.
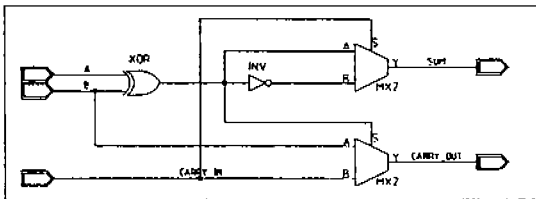
(Fig. 1) The schematic circuit diagram of a full adder

A CSA has 3 inputs. In the conventional array multiplier, an input of CSA is assigned with a bit among the product terms. These product terms are generated by the logical AND of multiplier and multiplicand. The other 2 inputs of CSA are assigned with the sum and Co from the previous row of CSA

array. So, the inputs of every CSA in CSA array are assigned with only one value at the initial time of multiplication. The other 2 inputs of CSA are assigned when the operations of previous CSA array is finished.

Let's assume that the propagation delay of every gate in CSA of (Fig. 2) is same as 1Δ t. If the 3 inputs of a CSA are simultaneously assigned then the operation delay of CSA can be said as 2Δt. In this case, 1Δt is exhausted by an XOR gate and 1Δt is exhausted by an 2:1 multiplexor.

In this case, the operation time of CSA takes 2Δt. The multiplier array uses CSA to generate the sum of the product terms and a carry propagation adder to generate the final results. The basic configuration of conventional multiplier array with CSA is shown in (Fig. 2). As the number of bits to multiply is increased, the portion of the multiplication time used by CSA array is more increased than the time used by an early propagation adder.
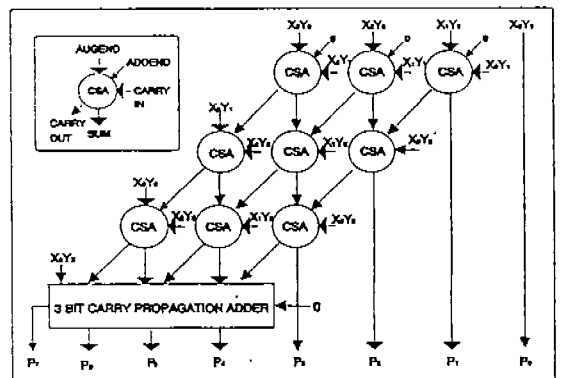
(Fig. 2) The schematic diagram of conventional multiplier array with CSA

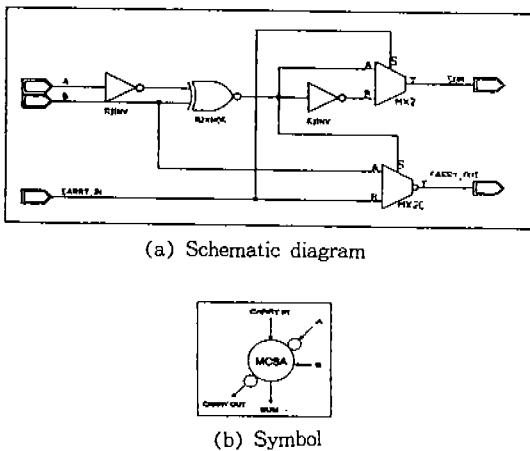## 3. Design of the multiplier array

In the configuration of CSA into the multiplier array, the augend and addend is assigned to the inputs which is connected to

XOR gate of a CSA faster than the Ci about $1\Delta t$, the operation time of each row of CSA array can be reduced to $1\Delta t$. This is the basic idea how to reduce the operation time of CSA array.

In order to assign 2 inputs of XOR gate in CSA faster than $1\Delta t$ compared with Ci of CSA, Co from the previous row of CSA array is assigned to the one of the inputs of an XOR gates of a CSA in the next raw. Because the other input of XOR is assigned to a bit from the product terms, all inputs of an XOR gate are assigned faster than $1\Delta t$ compared with the Ci and each CSA in CSA array is completely operated within $1\Delta t$.

To reduce the delay from Ci to sum of CSA, CSA cell is redesigned in (Fig. 3) and it is named MCSA(modified CSA). MCSA has an inverted input and an inverted Co. For the carry propagation adder, we designed the named DCSA (double inverted inputs of CSA) in (Fig. 4). DCSA is due to the fact that two inverted Co is generated from the two previous MCSA rows.

The designed multiplier array is configured using the proposed scheme of MCSA and DCSA in (Fig. 5).

4. Performance evaluation and simulation

The multiplication time exhausted by the conventional and proposed scheme of multiplier array can be represented by (1) and (2) respectively in the case of n by n bits multiplication.

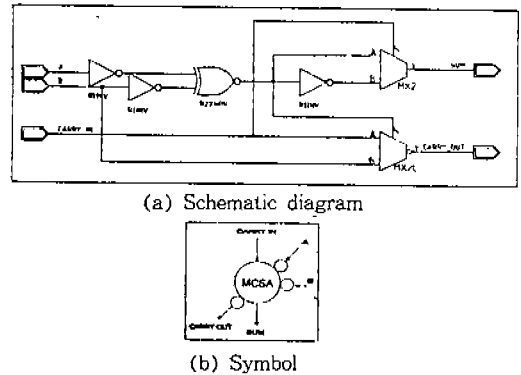$$t_{CON} = (n - 1) \times t_{CSA} + t_{CPA} \quad (1)$$
$$t_{PRO} = n \times t_{MCSA} + t_{CPA} \quad\quad (2)$$

Where

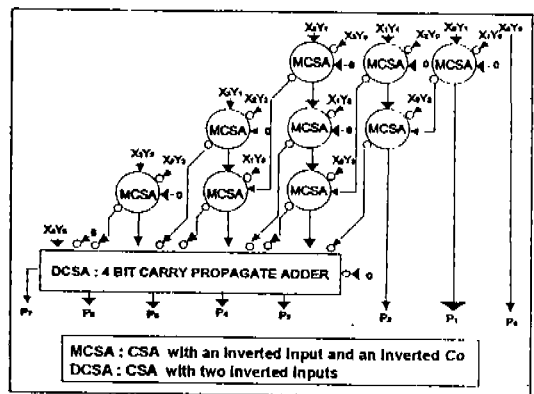t_{CON} : multiplication time by the conventional scheme

t_{PRO} : multiplication time by the proposed scheme

n : the bit size of operands

(a) Schematic diagram

(b) Symbol

(Fig. 4) DCSA having two inverted inputs and an inverted Co

(a) Schematic diagram

(b) Symbol

(Fig. 3) MCSA having an inverted input and an inverted Co

(Fig. 5) The configuration of the multiplier array with the proposed scheme

t_{CSA} : a CSA operation time in the conventional CSA array scheme

t_{MCSA} : a CSA operation time in the proposed CSA array scheme

t_{CPA} : operation time of a carry propagation adder

If the operation delay of every gate cell is supposed in a CSA as $1\Delta t$, it can be supposed that tCSA is $2\Delta t$ and tMCSA is $1\Delta t$ form Eq. (1) and (2). ⟨Table 3⟩ shows the comparison of operation time between the conventional and the proposed scheme with a several size of operands.

⟨Table 3⟩ The compared operation time between the conventional and proposed scheme

type of multiplier array	# of operands 4bits	8bits	16bits
conventional	$8\Delta t$	$17\Delta t$	$35\Delta t$
proposed	$4\Delta t$	$8\Delta t$	$17\Delta t$

The proposed multiplier array in (Fig. 5) is designed on INTERGRAPH Aceplus tool with Generic library and simulated on Advansim tool. The simulated results is shown in (Fig. 6). Here, we confirmed that 1 Δt unit time is about 6ns and the proposed

Operation delay time : ▨:4bits, ▧:8bits, ▦:16bits
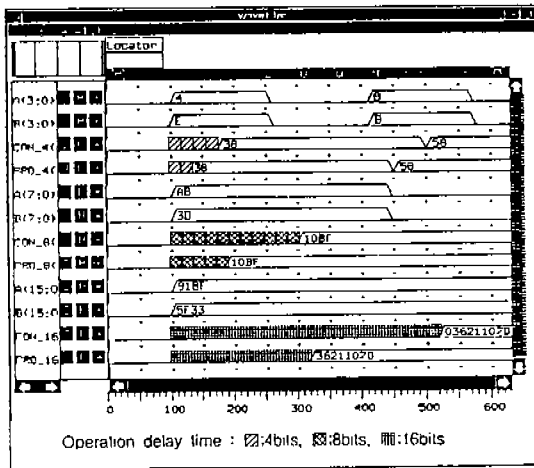
(Fig. 6) The simulated opeararion time with proposed scheme

scheme can speed up the multiplication time about twice. Thus, it is proved that the ability of proposed scheme through the simulation.

The multiplication time of multiplier array is composed of the operation time of CSA array and the operation time of the carry propagation adder. But the portion of the carry propagation addition time compared to CSA array operation time became smaller as the size of the operands is increased.

This configuration have some disadvantages over the conventional scheme. In the proposed scheme, the circuit size is increased about 13 % and has a complexity routing and a worse regularity compared to the conventional scheme.

5. Conclusion

Many of studies about the structure of the multiplier array were concentrated to reduced the number of product terms. Some of the major study of reducing the number of product terms could be the techniques which are called the modified Booth's algorithm and the multi-bit recording techniques. But in this paper, the modified configuration of conventional CSA array which can reduce the operation time of CSA array about half. The proposed scheme of CSA array can be used in the most scheme of multiplier array which reduce the number of product terms.

The proposed scheme may have disadvantage over the conventional scheme. It requires a little more gates because of the Co is transferred to the second below of CSA array, and so it needs more effort for routing. But the author can speed up the multiplication time of multiplier array about twice and its performance can be increased as large as the size of operands to multiply.

In this paper, the author have not verified

the efficiency of the suggested scheme at the physical situation. This can be done by VLSI layout and its implementation.

Acknowledgement

References

[1] O. L. MacSorley, "High speed arithmetic in binary Computers," Proc. IRE, Jan. 1961.

[2] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Compute., Feb. 1964.

[3] L. Dadda, "Some schemes for parallel multipliers," Alta Frequenza, Vol. 34. May. 1965.

[4] K. Hwang, Computer Arithmetic Principles, Architecture and Design. New York, Willey, 1979.

[5] Homayoon Sam and Arupratan Gupta, "A generalized multi bit recording of two's complement binary numbers and its proof with application in multiplier implementation," IEEE Trans. Comput., Aug. 1990.

[6] S. Vassiliadis, E. M. Schwartz, and D. J . I Hanrahan, "A general proof for overlapped multiple-bit scanning multiplication," IEEE Trans. Comput., Feb. 1989.

[7] Mark G. Anold, Thomas A. Bailey, John R. Cowles and Jerry J. Cupal, "Redundant logarithmic with difference grouping programmable logic array," IEEE Trans. Compute., Aug. 1985.

[8] N. Bandeira, K Vaccaro, J. A. Howard, "A two's complement array multiplier using true value of the operands," IEEE Trans. Comput., Aug. 1983.

이 강 현

1977년 조선대학교 공과대학 전자공학과 졸업(공학사)
1981년 조선대학교 대학원 전자공학과 졸업(공학석사)
1990년 아주대학교 대학원 전자공학과 졸업(공학박사)
1987년~현재 조선대학교 공과대학 전자공학과 교수
1991년 미 스탠포드대학 CRC 협동연구원
1994년~현재 조선대학교 VLSI/TEST공동실험실장
1995년~현재 조선대학교 수송기계부품 공장자동화 센타 산학협력실장
1995년~현재 조선대학교 자동화시스템 우수연구 센터 간사
관심분야 : FA정보통신, VLSI/TEST, 시스템 진단, 컴퓨터비젼