

Mac Architecture – Accumulator Based on Booth Encoding Parallel Multiplier

K.Hima Bindu, K.Bala Souri, K.V. Ramana Rao

Abstract- The MAC provides high speed multiplication with accumulative addition. In this paper, we study the various parallel MAC architectures and then implement a design of parallel MAC based on some booth encodings such as radix-4 booth encoder and some final adders such as CLA, Kogge stone adder and then compare their performance characteristics.

The one most effective way to increase the speed of a multiplier is to reduce the number of the partial products. Although the number of partial products can be reduced with a higher radix booth encoder, but the number of hard multiples that are expensive to generate also increases simultaneously. To increase the speed and performance, many parallel MAC architectures have been proposed. Parallelism in obtaining partial products is the most common technique used in this implemented architecture.

Keywords: Radix-4 Booth multiplier, CLA, multiplier and-accumulator (MAC).

I. INTRODUCTION

In this paper, we study the various parallel MAC architectures and then implement a design of parallel MAC based on some booth encodings such as radix-2 booth encoder and some final adders such as CLA, Kogge stone adder and then compare their performance characteristics. A Digital multiplier is the fundamental component in general purpose microprocessor and in DSP [1]. Compared with many other arithmetic operations multiplication is time consuming and power hungry. Thus enhancing the performance and reducing the power dissipation are the most important design challenges for all applications in which multiplier unit dominate the system performance and power dissipation. The one most effective way to increase the speed of a multiplier is to reduce the number of the partial products.

Although the number of partial products can be reduced with a higher radix booth encoder, but the number of hard multiples that are expensive to generate also increases simultaneously. To increase the speed and performance, many parallel MAC architectures have been proposed. Parallelism in obtaining partial products is the most common technique used in this architecture.

There are two common approaches that make use of parallelism to enhance the multiplication performance. The first one is reducing the number of partial product rows and second one is the carry-save-tree technique to reduce multiple partial product rows as two "carry-save" redundant forms.

Manuscript received October 30, 2011.

K.Hima Bindu, M.Tech(ECE), Pydah College of Engineering & Technology, India. Email: kbsjob@gmail.com

K.Bala Souri, M.Tech(ECE), Pydah College of Engineering & Technology, India.

K.V. Ramana Rao, Assoc.Professor & Head, Dept. of ECE, Pydah College of Engineering & Technology, India.

An architecture was proposed in [2] to provide the tact to merge the final adder block to the accumulator register in the MAC operator to provide the possibility of using two separate N/2-bit adders instead of one N-bit adder to accumulate the N-bit MAC results.

The most advanced types of MAC has been proposed by Elguibaly in which accumulation has been combined with the carry save adder (CSA) tree that compresses partial products and thus reduces the critical path. Later on a new architecture for a high-speed MAC is proposed by Seo and Kim. The difference between the two is that the latest one carries out the accumulation by feeding back the final CSA output rather than the final adder results.

II. BOOTH'S ALGORITHM

A.D. Booth proposed Booth encoding technique for the reduction of the number of partial products [3]. This algorithm is also called as Radix-2 Booth's Recoding Algorithm. Here the multiplier bits are recoded as Zi for every ith bit Yi with reference to Yi-1 .This is based on the fact that fewer partial products are generated for groups of consecutive zeros and ones. For a group of consecutive zeros in the multiplier there is no need to generate any new partial product. We only need to shift previously accumulated group partial product one bit position to the right for every 0 in the multiplier.

The radix-2 algorithms results in these observations [4]:

(a) Booth observed that whenever there was a large number of consecutive ones, the corresponding additions could be replaced by a single addition and a subtraction

$$2^j + 2^{j-1} + \dots + 2^{i+1} + 2^i = 2^{j+1} - 2^i$$

(b) The longer the sequence of ones, the greater the savings.

(c) The effect of this translation is to change a binary number with digit set [0, 1] to a binary signed-digit number with digit set [-1, 1].

The Radix-2 Booth algorithm Table 1 is give below:

Yi	Yi-1	Zi	Explanation
0	0	0	No String of 1s in sight
0	1	1	End of string of 1s in Y
1	0	1	Beginning of string of 1s in Y
1	1	0	Continuation of string of 1s in Y

In this algorithm the current bit is Yi and the previous bit is Yi-1 of the multiplier Yn-1Yn-2..... Y1 Y0 are examined in order to generate the ith bit Zi of the recoded multiplier Zn-1 Zn-2.....Z1 Z2.

The previous bit Y_{i-1} serves only as the reference bit. The recoding of the multiplier bits need not be done in any predetermined order and can be even done in parallel for all bit positions. The observations obtained from the radix-2 Booth recoding is listed below:

- It reduces the number of partial products which in turn reduces the hardware and delay required to sum the partial products.
- It works well for serial multiplication that can tolerate variable latency operations by reducing the number of serial additions required for the multiplication.
- The number of serial additions depends on the data (multiplicand).
- Worst case 8-bit multiplicand requires 8 additions.
- $01010101 \Leftrightarrow 1 -1 1 -1 1 -1 1 -1$
- Parallel systems generally are designed for worst case hardware and latency requirements. Booth-2 algorithm does not significantly reduce the worst case number of partial products.
- Radix-2 Booth recoding is not directly applied in modern arithmetic circuits; however, it does help in understanding the higher radix versions of Booth's recoding. It doesn't have consecutive 1s or -1s. The disadvantages of the radix-2 Booth algorithm can be overcome by using Modified Booth algorithm.

2.1 Booth Encoder Module

For the design of a faster multiplier, we should either reduce the number of partial products or increases the summation of partial products. The Booth algorithm reduces the number of partial products. Based on the available literature, we propose a few designs of the Booth encoder and selector logic. The proposed designs are based on modified Booth recoding system using radix-4 multiplication where it reduces the number of partial products to half. The multiplicands are replicated and separate carry and sum vectors are obtained at the output of the compressor. Hence Booth recoding is fully parallel and carry free. Moreover, it can be applied to design a tree and array multiplier, where all the multiples are needed at once. Radix-4 Booth recoding system works perfectly for both signed and unsigned operations.

The Booth encoder constitutes one third part of the multiplier circuit, so it is significant to have an efficient design for the partial product generator. Modified Booth algorithm successfully proved to reduce the partial products by half. To further enhance the performance of the multiplier in terms of power, area and delay, pass logic principle can be incorporated. Novel Booth encoder designs using pass logic principle are proposed in this section which combines the benefits of low power consumption and reduced chip area when compared to other conventional designs.

2.2 Booth Encoder and Partial Product Generator

Partial product generation is the very first step in binary multiplier. Partial product generators for a conventional multiplier consist of a series of logic AND gates as shown in Figure 2.

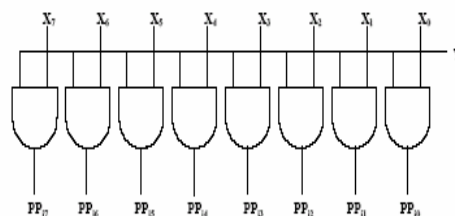


Figure 2. Partial Product generator using AND gates [4]

If the multiplier bit is '0', then partial product row is also zero, and if it is '1', then the multiplicand is copied as it is. From the second bit multiplication onwards, each partial product row is shifted one unit to the left. In signed multiplication, the sign bit is also extended to the left.

III. MODIFIED BOOTH ALGORITHM

The radix-2 disadvantages can be eliminated by examining three bits of Y at a time rather than two. The modified Booth algorithm is performed with recoded multiplier which multiplies only +a and +2a of the multiplicand, which can be obtained easily by shifting and/or complementation. The truth table for modified Booth recoding is shown below:

Table 2. Radix-4 Booth Recoding [2]

Y_{i+1}	Y_i	Y_{i-1}	Z_{i+1}	Z_i	$Z_i/2$	Explanation
0	0	0	0	0	0	No String of 1s in sight
0	0	1	0	1	1	End of strings of 1s
0	1	0	0	1	1	Isolated 1
0	1	1	1	0	2	End of string fo 1s
1	0	0	-1	0	-2	Beginning of string of 1s
1	0	1	-1	1	-1	End a string, begin a new one
1	1	0	0	-1	-1	Beginning of string of 1s
1	1	1	0	0	0	Continuation of string of 1s

The main advantage of the modified Booth algorithm is that it reduces the partial products to $n/2$.

The following gives the algorithm for performing sign and unsigned multiplication operations by using radix-4 Booth recoding.

Algorithm: (for unsigned numbers)

- Pad the LSB with one zero
- Pad the MSB with two zeros if n is even and one zero if n is odd

- Divide the multiplier into overlapping groups of 3-bits
- Determine partial product scale factor from modified Booth-2 encoding table
- Compute the multiplicand multiplies
- Sum partial products

Algorithm: (for signed numbers)

- Pad the LSB with one zero

If n is even don't pad the MSB ($n/2$ PP's)

- Divide the multiplier into overlapping groups of 3-bits
- Determine partial product scale factor from modified Booth-2 encoding table
- Compute the multiplicand multiplies
- Sum partial products

Booth recoding is fully parallel and carry free. It can be applied to design a tree and array multiplier, where all the multiples are needed at once. Radix-4 Booth recoding system works perfectly for both signed and unsigned operations

3.1 Booth Encoders

The modified Booth's algorithm based on a radix-4, generally called Booth-2 [4] is the most popular approach for implementing fast multipliers using parallel encoding. It uses a digit set $\{0, \pm 1, \pm 2\}$ to reduce the number of the partial products to $n' = \lceil (n+1)/2 \rceil$. Radix-4 encoding start by appending a zero to the right of multiplier LSB. Triplets are taken beginning at position $x-1$ and continuing to the MSB with one bit overlapping between adjacent triplets.

Table 1. Radix-4 booth encoding

Y_{n+1}	Y_n	Y_{n-1}	Z_n	Partial Product
0	0	0	0	0
0	0	1	1	1 x Multiplicand
0	1	0	1	1 x Multiplicand
0	1	1	2	2 x Multiplicand
1	0	0	-2	-2 x Multiplicand
1	0	1	-1	1 x Multiplicand
1	1	0	-1	1 x Multiplicand
1	1	1	0	0

This recoding scheme applied to a parallel multiplier halves the number of partial products so the multiplication time and the hardware requirements decrease. Radix-8 recoding [5] applies the same algorithm as radix-4, but now in this we take quartets of bits instead of triplets. The Booth-3 scheme is based on a radix-8 encoding to reduce this number to $n' = \lceil (n+1)/3 \rceil$. All digit sets $\{0, \pm 1, \pm 2, \pm 3, \pm 4\}$ are obtained by simple shifting and complementary operations, except generation of the multiple $3X$, which is computed by an adding and shifting operation, $3X = 2X+X$ and $-3X$ can be generated by complement $3X$. Radix-4 booth encoding that generates partial products is shown in Table 1 respectively. Simulation result for the 16-bit parallel MAC based on radix-4 booth encoder and using Radix-5 Kogge stone as a final adder is shown in Figure 6 respectively and performance characteristics in terms of speed and area are shown in table I. It could be seen from table that the parallel MAC based on Radix-5 Kogge stone adder is having more area as compared to others but having less delay. Higher would be the no. of slices, higher would be area. Thus, MAC based on Radix-5 Kogge stone adder shows higher performance than others but at the cost of area.

IV. CONCLUSION

The primary objective of this paper has been to present a new type of Parallel MAC using Radix-4 Kogge stone adder, to reduce the implementation to practice, and to show through simulation and design that this algorithm is competitive with other more commonly used algorithms when used for high performance implementations.

Secondarily, this paper has shown that algorithms based upon the Radix-4 Booth partial product method are distinctly superior in performance. From this paper, we conclude that Parallel MAC based on Radix-2 booth encoder and using Radix-4 Kogge stone adder has higher speed of operation and thus can be used in high performance systems. This work can be utilized in any of the following such as in DSP applications, Numerical co-processor, Calculators (pocket, graphic etc), Filtering, Modulation & Demodulation etc. As the summation networks and partial product generation logic results in higher delay and most of the area of a MAC. Thus in future, some more techniques and advancement needs to be done which further improves the performance of MAC. Also some measures should be taken which minimize the area consumption.

REFERENCES

1. J. J. F. Cavanagh, "Digital Computer Arithmetic", New York: McGraw- Hill, 2004.
2. K. Hwang. Computer Arithmetic – Principles, Architecture and Design. School of Electrical Engineering 1979.
3. Chung Nan Lyu & David W. Matula Redundant Binary Booth Recoding Proceedings of the 12th Symposium on Computer Arithmetic (ARITH '95) 1995 IEEE.
4. B. Cherkauer and E. Friedman. A Hybrid Radix-4/Radix-8 Low Power, High Speed Multiplier Architecture for Wide Bit Widths. In IEEE International Symposium on Circuits and Systems, volume 4, pages 53–56, 1996.
5. Yajuan He and Chip-Hong Chang, "A New redundant binary booth encoding for fast 2n-bit multiplier design", IEEE Transaction on circuit and systems, Vol. 56, No.6, June 2009.

AUTHORS PROFILE



K. Hima Bindu pursuing her M.Tech(ECE) in Pydah College of Engineering & Technology, under the guidance of K.V Ramana Assoc.Professor & Head, Dept.of ECE at Pydah College of Engineering & Technology, Visakhapatnam, A.P., India. My research Interests are VLSI Design , Digital Signal Processing and Instrumentation.



K. Bala Souri pursuing his M.Tech(ECE) in Pydah College of Engineering & Technology, under the guidance of K.V Ramana Assoc.Professor & Head, Dept.of ECE at Pydah College of Engineering & Technology, Visakhapatnam, A.P., India. My research Interests are VLSI Design and Digital Signal Processing.



Sri K.V. Ramana Rao working as Assoc. Professor & Head, Department of ECE at Pydah College of Engineering & Technology, Visakhapatnam, A.P., India. His research interests are Digital Signal Processing and VLSI Design.