International Journal of Advanced Information Science and Technology (IJAIST) *ISSN: 2319:268*
Vol.2, No.4, April 2013 *DOI:10.15693/ijaist/2013.v2i4.165-169*

Design Of High Accuracy Fixed Width Booth Multiplier Using Linear Compensation Method

M.Vinmathi, Pg scholor, Karpagam University, Coimbatore.
Mr.J. SHAFIQ MANSOOR, ME., Asst Professor, Karpagam University, Coimbatore.

Abstract---In the majority of Digital Signal Processing (DSP) applications the critical operations usually involve multiplications. In the last few years, the main consideration of linear compensation method is applied to fixed-with booth multiplier is to enhance its speed. This is because speed and throughput rate is always the concern of digital signal processing system. The linear compensation methods for fixed-width Booth multipliers are presented to reduce the truncation errors The errors can be reduced hugely in when compared to booth the D-T & P-T Booth multiplier. The array multipliers are the simple and easy to design. But when designing high speed multipliers for signal processing applications, this will increase the delay because of the unwanted sign bit extension till the $2N^{th}$ bit of the partial products. Hence to reduce hardware complexity and delay the special and effective sign extension reduction technique is proposed.

Index Terms-Linear compensation method, Booth multiplier, Fixed width.

I. INTRODUCTION

A. Motivation of the project

In this project we are going to design the high accuracy fixed width booth multiplier using linear compensation method. Fixed width multiplier has the same bit width of input and output. In digital signal processing system multiplier plays an important role but also consumes more power and area. In order to reduce the power and area occupied by the multiplier unit fixed width multipliers are unsigned. The fixed width property simplifies the multiplier structure with the aim of improving power and speed. In fixed width multiplication the product is truncated to n-bits, the least significant columns of the product matrix contribute little to the final result. Over the years lot of research has been done in designing and implementing multiplication functions that yield less area on the chip, consume less power and have minimal propagation time. The move towards achieving less area on chip started with the implementation of fixed width multiplier. A fixed width multiplier has smaller silicon area compared to a full width multiplier which takes in n-bit multiplicand and n-bit multiplier to yield an output that is 2n-bits wide. A fixed width multiplication can be derived by truncating a full-width multiplier. In this project a high accuracy linear compensation method for fixed width booth multiplier is proposed.

B.Objective of the Project

To propose linear compensation method is applied to the fixed-width Booth multiplier which reduces number of partial product generation which reduces delay of the partial product generation. To reduce the area of multiplier by implementing special sign extension technique. To further improve the speed by adding the partial products using compressors. To implement the multiplier in signal processing application. The fixed-width Booth multiplier will reduce the partial products into half, and hence the adder stages will be reduced. Also the multiplier area is reduced effectively.

II. PROPOSED LINEAR COMPENSATION METHOD USING FIXED WIDTHBOOTH MULTIPLIERS

This project design of low-error fixed-width modified booth multiplier is proposed. In many multimedia and DSP applications, multiplication operations have the fixed-width property. That is, their input data and output results have the same bit width. To efficiently compensate for the quantization error, Booth encoder outputs (not multiplier coefficients) are used for the generation of error compensation bias.

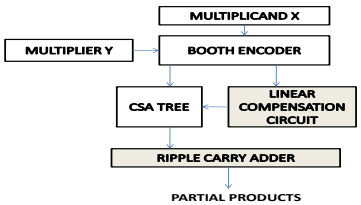


Fig. 1. The Whole Architecture of the Proposed linear compensation Booth Multiplier

The multiplier has two stages, the first stage consists of booth encoders which drive partial product generators which in turn drive a carry-save addition array to produce two final partial products. In the second stage, the two final products are added to form the final product through a ripple-carry adder. Final output is minimum error.

*A.*Linear compensation method

Parallel multipliers are fundamental building blocks in multimedia and digital signal processing systems. In many applications, the inputs and the output of the multiplier have the same bit width. These circuits are denoted in literature as fixed-width multipliers or truncated multipliers. The most obvious way to design a fixed-width multiplier uses nxn a full-width multiplier, whose output is truncated/rounded to bits by discharging n the less-significant bits of the products. The fixed-width property, however, can be exploited to simplify the multiplier structure, with the aim of improving power and speed. Basically, one can discard some of the partial-products in the partial-products array to reduce the circuit
complexity, at a price in terms of accuracy. This is the approach pursued in all the fixed-width multiplier architectures proposed in literature. It shows the architecture of a typical fixed-width multiplier. In this example, without loss of generality, the inputs and are assumed to be two unsigned fractional numbers.

B Modified booth multiplier

Multiplication consists of three steps - First step is to generate the partial products. Second step is to add the generated partial products until the last two rows are remained. Third step is to compute the final multiplication results by adding the last two rows.

y7	y6	y5	y4	y3	y2	y1	y0	0

Fig. 2. 8-Bit Modified Booth Encoding

C. Sign extension reduction technique

To avoid sign extending each row of partial products, the scheme presented by Shiann-Rong Kuang has been used. Instead of sign extension, an extra partial product for each row and a special pattern of 1's and 0's has been added. But this will increases the number of partial product rows.To further simplify implementation, the partial product array is modified according to the special sign extension reduction scheme. First all the sign terms were assumed to be '1' and are added to get default 1's and 0's for each row. By adding all the 1's using binary addition and by using the special sign extension reduction technique given below. Figure 4.4 shows the generated partial products and sign extension scheme of the 16-bit Modified Booth Multiplier. By this method only 2 bits are extended in each row, except first row. Using the following equation it is proved that the above method is equal to that of the sign extended method.

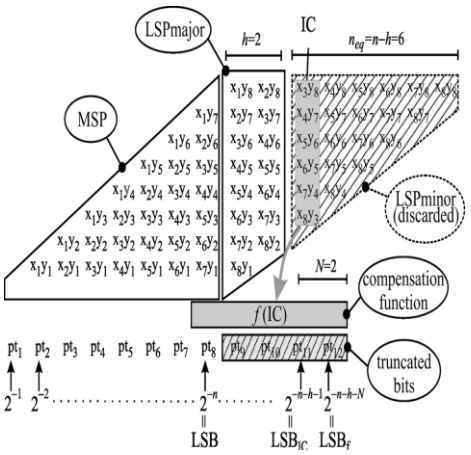


Fig. 3 Partial-products matrix of a 8 bit multiplier, truncated h=2 with and using a variable-correction scheme to compute the result.

D.Partial Product Generator

The partial product generator only uses eleven transistors. It actually is a multiplexer which controlled by the booth encoder selection signals. The output of the partial product generators are multiples of the multiplicand (-2X, -X, 0, +X, +2X).

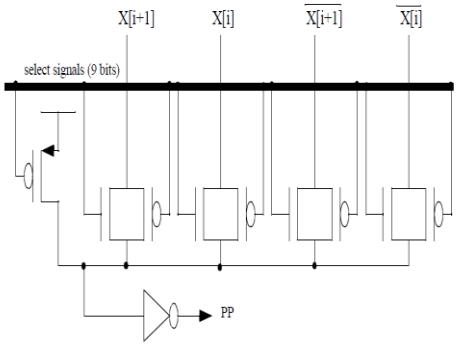


FIG 3 Partial Product Generator

C. 16-Bit Multiplication Matrix

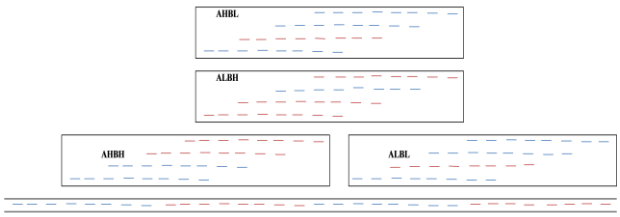


Fig. 4. Multiplication Matrixes for 16-Bit Multiplication

The Multiplication expression is divided in to four sub expressions AHBH, ALBL, AHBL and ALBH as shown in Figure 4.5. Here AH means A [15:8] and AL means A [7:0] and similarly for operand B. Four independent partial-product arrays are produced by using Radix-4 Booth Encoding approach. The partial products generated for all the individual blocks are grouped as

shown in the Figure 4.5 to obtain the final product using adders and compressors.

Wallace tree multiplier whose realization is shown in Fig. 8

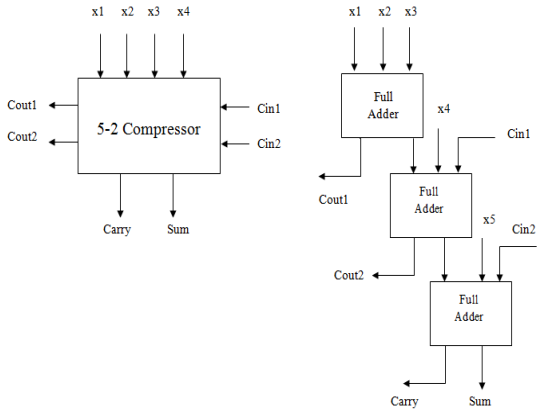
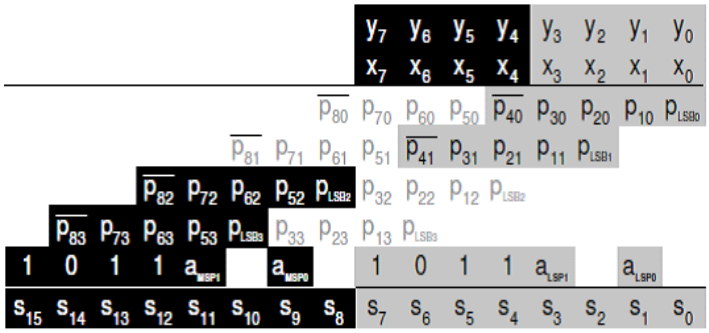


Fig. 5. 5-2 Compressor (a) Block diagram, (b) Realization using Full Adder



By calculating the partial products shown in the gray region, it is possible to compute one *N*/2-bit multiplication. If the partial products for both the gray and black regions are calculated, two *N*/2-bit multiplications can be performed in parallel. Where, Figure.10 shows the Partial-product representation of an 8-bit multiplication with two 4-bit multiplications. $P_{ij} = Y_i X_j$. The Twin-precision concept has been shown to translate into an efficient use of resources, when used for multipliers such as the radix-2 tree multiplier. It has been shown that by optionally setting different regions of partial products to zero, it is possible to compute one *N*, one *N*/2, or two concurrent *N*/2-bit multiplications using an *N*-bit radix-2 tree multiplier. Only with a minor increase of logic, this enables the computation of *N*/2-bit multiplications at reduced power, with shorter delay and with a possibility for higher throughput, as compared to an *N*-bit multiplier without the Twin-precision.the change that occur in the output of the multiplier stage. The variation in the output of the device is measured from one by one .the compression of the device at the one of the terminal .the carry save adder of the implement device through delay.

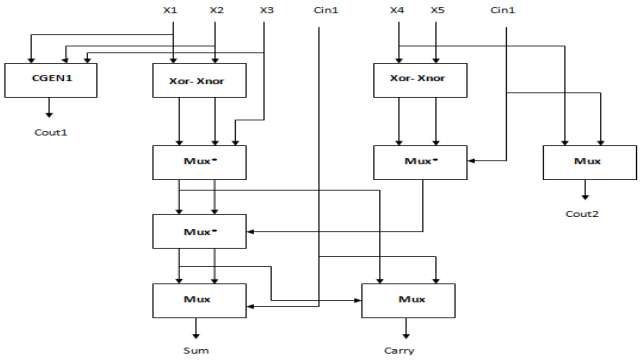


Fig. 6. High Speed 5-2 Compressor using Carry Generation Module-CGEN 1

E. Gate count and delay analysis in hs-compressors

The gate count and delay of HS compressor circuits proposed in by Veeramachaneni, and the standard designs . Since one gate is less in case of 4-2 compressor and 2 gates less in case of 5-2 compressor circuits proposed by Veeramachaneni compared compressor with FAs significant reduction in area overhead and critical delay is realized in partial product accumulation in our proposed tree multiplier. Increases. Instead of exhaustive simulation methods, the probability methods can fast and mathematically generate the compensation functions. In this way, the ACPE establishes compensation circuit by applying the conditional-probability method and can achieve higher accuracy performance than the work in that only uses the probability method. The conditional-probability method introduced in, however, does not achieve higher accuracy than existing works because the conditional bits are not chosen

F. Carry-Save-Adder Tree

Carry save adder trees are most efficiently implemented by putting together the (3,2) blocks, we must still address the issue of how to implement the (3,2) block (carry save adder) efficiently. Functionally, the carry save adder is identical to the full adder. The full adder is usually implemented with a reduced delay from Cin to Cout because the carry chain is the critical delay path in adders. Unfortunately, there is no single carry chain in the carry save adder trees in multipliers. Thus, it does not pay to make the delay shorter for one input by sacrificing delay on other inputs for carry save adders. Instead, carry save adders are normally implemented by treating the 3 inputs equally and trying to minimize delay from each input to the outputs.

*G.*Ripple Carry Adder

The general structure of an n-bit ripple carry adder is shown in fig. The A and B input vectors represent the final two n-bit partial products generated by the multiplier addition array. The carry out (Cout) propagation delay is due to the carry signal rippling serially through the block (initiated by the Cin input). Therefore, a full adder with a fast serial-carry delay is needed in the carry block.

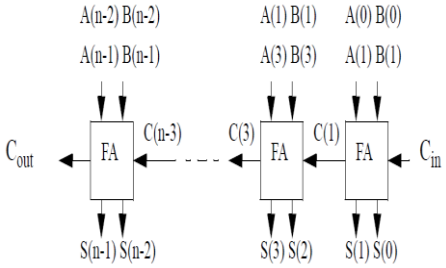


Fig 7 n-bit Ripple Carry Block(Each Block is Two Bit Adder)

H. Application to histogram

Histogram Equalization is a method that increases the contrast of an image by increasing the dynamic range of intensity given to pixels with the most probable intensity values.

III. RESULTS AND DISCUSSIONS

A. Encoder Output

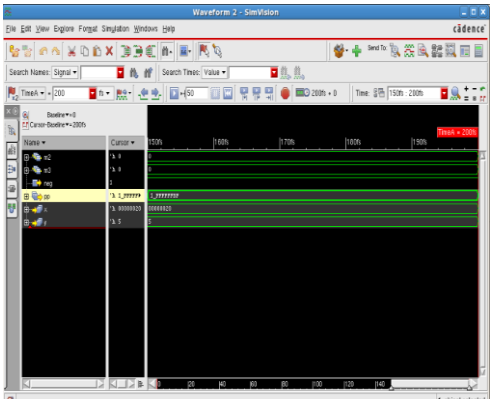


Fig. 8. Simulation results for Fixed Width Booth Encoder

The simulation result for fixed width booth encoder module as shown in fig 5.1. Here the 32-bit input are taken from a and b and 32 output obtained y.

B.CSA OUTPUT

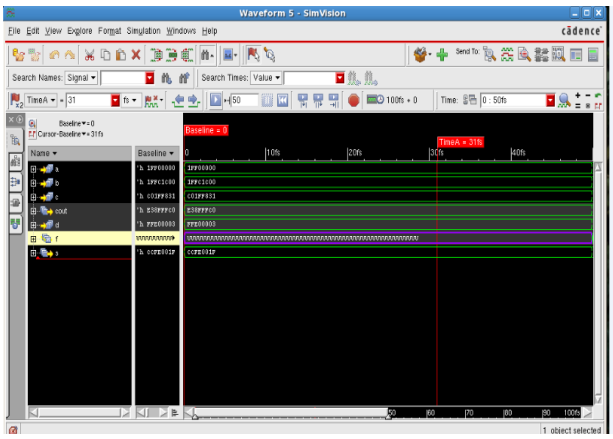


Fig. 9. Simulation results for CSA

The simulation result for CSA module as shown fig 5.2. Here the 32-bit input are taken from a and b and 32-bit output obtained Cout and S.

SIMULATION RESULT LINEAR COMPENSATION FUNCTION BY USING FIXED WIDTH BOOTH MULTIPLIER

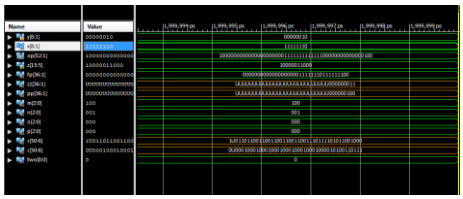
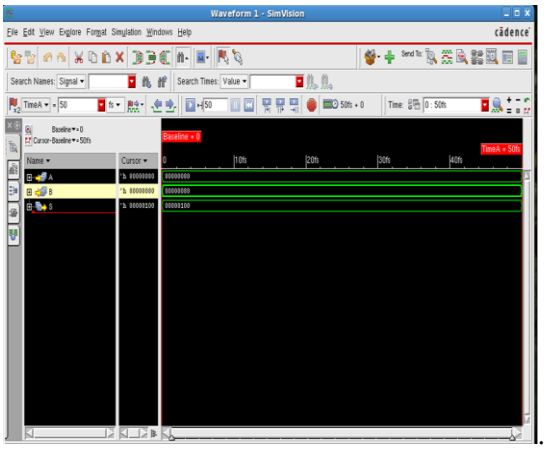


Fig 10. Simulation result for linear compensation method

The simulation result for linear compensation function using fixed width booth multiplier as shown in fig 4.3 . Here the 32-bit input are taken from a and b and 32-bit output obtained Cout and S.

C.RCA OUTPUT



. Fig 11. Simulation results for RCA

The simulation result for RCA module as shown in fig 5.3. Here the32-bit input are taken from a and b and 32-bit output obtained and S.

Table I

A. Comparisons of area, power and delay with various method

Multiplier	Area (10^3 μm^2)	Relative area	Relative power
Full-width	18.5	100	100
Full-rounded	16.2	88	84
Fixed width 1 bit h=0	9.2	50	47
Fixed width 1 bit h=1	0.140	55	51
Fixed width 1.5 bit h=1	0.13	53	50

IV.CONCLUSION

This project proposes a linear compensation method in fixed-width Booth multipliers. Linear compensation method, which can be easily, applied to large length Booth multipliers for achieving higher accuracy performance. The design parameter h information is also induced to adjust the accuracy with respect to system requirements. As a result, the proposed linear compensation method provides a flexible and high accuracy to fixed-width Booth multipliers.

V. FUTURE WORK

In future the linear compensation method which can be applied to fixed width Booth Multiplier can be extended to higher radix Booth Encoding for further reducing the number of partial product rows with 64 bit and 128 bit fixed with booth multiplier architectures.

REFERENCES

[1] R. D. Chen J. M. Jou, and S. R. Kuang,1999 "Design of low-error fixed width multipliers for DSP applications," *IEEE Trans. Circuits Syst. II,Analog Digit. Signal Process., vol. 46, no. 6, pp. 836–842.*

[2] D. D. Caro, V. Garofalo, E. Napoli, N. Petra and A. G. M. Strollo, 2011" Design of fixed-width multipliers with linear compensation function," *IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 58, no. 5, pp. 947–960.*

[3] C. H. Chang and R. K. Satzoda,2010 "A low error and high performance multiplexer-based truncated multiplier," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 12, pp. 1767–1771.

[4] C. H. Changand Y. He,2009 "A new redundant binary Booth encoding forfast -bit multiplier design," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 6, pp. 1192–1201.

[5] C. H. Chang, J. Gu, and M. Zhang,2004 "Ultra low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 10, pp. 1985–1997.

[6] D. DeCaro, N. Petra, and A. G. M. Strollo,2005 "Dual-tree error compensation for high performance fixed-width multipliers," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 8, pp. 501–507.

[7] W. S. Feng L. D. Van, S. S. Wang, 2000 "Design of the lower error fixed-width multiplier and its application," *IEEE Trans. Circuit SystII, Analog Digit. Signal Process.*, vol. 47, no. 10, pp. 1112–1118, Oct.

[8] C. Y. Guo ,S. R. Kuang, J. P. Wang, 2009"Modified Booth multipliers with a regular partial product array," *IEEE Trans. Circuits Syst. II, Exp.Briefs*, vol. 56, no. 5, pp. 404–408.

AUTHORS PROFILE

Mr.J Shafiq Mansoor received the **M.Sc.** degree in Applied Electronics from the PSG CAS, Bharathiyar university, Coimbatore, India in 2008, the **M.E.** degree in Embedded Sysytems from Karpagam University, Coimbator, India. He is currently working as a Assistant Professor, ECE Department in Karpagam, Coimbaotre, India.

Ms.M.Vinmathi received the **B.E.** degree in Electronics and Communication Engineering from the SSM College of Engineering, Namakkal, Anna University, Chennai, India, in 2009, she is pursuing **M.E.** degree in Electronics and Communication Engineering (VLSI Design) from the Karpagam University, Coimbatore, India. Area of interests VLSI Design, CMOS VLSI Design, Dighital Electronics.