



Pipeline Architecture MLCP Estimator for Fixed Width Booth Multiplier

G. Durga Prasad¹, Dr. K. Babulu²

M.Tech, Department of ECE, JNTUK, Kakinada, India¹

Professor, Department of ECE, JNTUK, Kakinada, India²

ABSTRACT: Booth multiplier is a multiplication of two binary numbers in two's complement notation and the proposed project new technique introduced i.e. fixed width multiplier is part of booth multiplier, in this 5-2 compressor is used with pipelining so that it leads to decrease the absolute error and improve the accuracy and delay also less. Unlike previous conditional-probability methods, the proposed Multi Level Conditional Probability uses entire nonzero code, namely MLCP, to estimate the truncation error and achieve higher accuracy levels. Pipeline architecture is used for decreasing delay. The entire work is done in the Xilinx model simulator tool and RTL developed written by code in VHDL.

KEYWORDS: Absolute error, Fixed-width Booth multiplier, multilevel conditional probability (MLCP), pipeline architecture.

I. INTRODUCTION

Multiplication is one of the important function in arithmetic operations. The multiplication operation currently used in many applications like digital signal processing applications like convolution, FFT, Filtering and others [1-3]. There are many types multiplications exists in the real time world such as serial multipliers and parallel multipliers and both serial and parallel multipliers and array multipliers. In the array multipliers again sub-divided in to two types, there are signed multipliers and signed-unsigned multipliers. Booth multiplier and modified booth multiplier comes on to the signed and unsigned multipliers [4-7]. Fixed width multiplier is the fixed width bits generate at output side as input apply and it is used for to reduce truncation error. To calculate the truncation error, post truncated and direct truncated methods are used. In post truncated method directly truncates half of the bits so truncation error is more and in latter one directly truncates products half of LSBs so area is decreases but truncation error increase. To overcome these problems booth multiplier concept is introduced and the present paper is the latest technique in booth multiplication [4-7]. Multiplication requires an nbit full adder at the last stage to sum all the terms from the multiplier array; hence, pipelining is not allowable in such a design. Since pipelining is the most power effective data path for low power application, we extend work with a modified Booth pipelined algorithm for 2's complement multiplication

The rest of the paper described as section II deals with booth encoder, section III deals with MLCP estimator and 5-2 compressor and fixed width booth multiplier and it deals previous technique and section IV deals with proposed method and section V describes the results and followed by a conclusion.

II. RELATED WORK

A) BOOTH ENCODER

Booth Encoder is the main operation in booth multiplier. The working of booth multiplier is at time we consider two bits simultaneously, so that the numbers of partial products are reduced at the end of computing multiplication operation so that the delay will be reduced and speed also increased. Meanwhile the booth encoder performs encoding operations.

The working of Booth encoder is it performs encoding operation according booth table and consider bits are previous and present and the next bit like that we formed a table b_{i-1}, b_i, b_{i+1} as shown in table1 and perform iteration operation so that encoding performed then number partial products are simultaneously reduced so that the absolute error is reduced.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

TABLE 1: MODIFIED BOOTH MULTIPLIER

b_{i+1}	b_i	b_{i-1}	y_i	z_i	y_{i-1}
0	0	0	0	0	-
0	0	1	1	1	-1,-2
0	1	0	1	1	1,2
0	1	1	2	1	-1,-2
1	0	0	-2	1	1,2
1	0	1	-1	1	-1,-2
1	1	0	-1	1	1,2
1	1	1	0	0	-

B) FIXED WIDTH MODIFIED BOOTH-MULTIPLIER

Fixed width modified booth multiplier is a fixed output width as followed by input and it can be used in multimedia and digital signal processing systems mainly in less error on output side[11]. the error may be truncation error or absolute error. Already some of the projects were done on low error- fixed-width multiplier [1].

$$\begin{aligned}
 A &= -a_{L-1} 2^{L-1} + \sum_{i=0}^{L-1} a_i \cdot 2^i \\
 B &= -b_{L-1} 2^{L-1} + \sum_{j=0}^{L-1} b_j \cdot 2^j \\
 P &= A * B
 \end{aligned}
 \tag{1}$$

Here A and B are multiplicand and multiplier and P means are partial products. And L is the length of the sequence.

It is the main part of the multiplier and it can be used for to calculate real-partial products and truncated using fixed-width multiplication. The fixed-width booth multiplier can be written as follows:

$$P \approx Pq = MP + TP = MP + \sigma \cdot 2^L \tag{2}$$

Here MP is the main part of the multiplier and TP is the truncation part of the multiplier, main part gives the maximum product of output and truncation part gives the output with some error.

Here σ performs the rounded operation of truncation part, which consist of minor part (T_{mi}) and major part(T_{mj})

$$\sigma = \text{Round} (T_{mi} + T_{mj}) \tag{3}$$

So σ can obtain by adding minor and major terms of truncation part and then round the obtained value. The major term of truncation part gives true information and minor term can obtain by applying MLCP concept because of this method we can obtain less truncation error [1].

III PREVIOUS METHOD

This section describes about Multi Level Conditional Probability it can uses accuracy-adjustment fixed-width Booth multiplier method to design the compensated circuit [1]. The MLCP method produces a closed form with different bit widths L and column information w; thus, the compensated circuit can be established fast, and the accuracy can be adjusted by changing w. In contrast to the conditional-probability method for ACPE[11], which uses single nonzero code to estimate truncation errors, the proposed MLCP generates estimates by employing all nonzero code, which demonstrates high levels of inter correlation. Although MLCP method has higher difficult to calculate absolute errors when compared with ACPE one, the accuracy of MLCP method is higher than that of ACPE method. Furthermore, simple and small compensated circuits are proposed from a single compensated closed form. According to the trade-off between accuracy and circuit area, the MLCP method best provides a balance between accuracy and delay. The implementation results of this brief show that the proposed MLCP fixed width Booth multiplier achieves low-cost high-accuracy and less absolute error and delay less.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

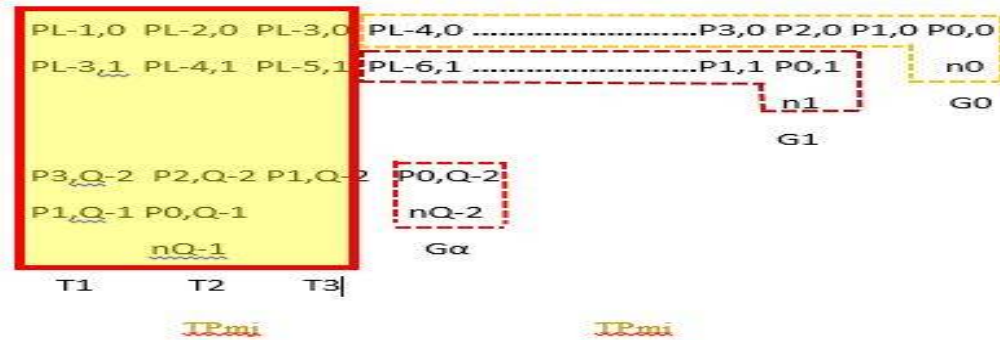


Figure1: MLCP concept for $w=3$

In figure 1, it shows that truncation partial products as divided into two parts major and minor. Major part is added using 5:2 compressors and minor part is calculate using MLCP. In MLCP all element values are same and we calculate $P_{j,0} \& n_j$ as [1]. Previously one method is used for to reduce truncation error and it also used for to improve high accuracy followed by MLCP with 4-2 compressor [1]. This section describes an accuracy-adjustment fixed-width Booth multiplier [11] that compensates the truncation error using a multi level conditional probability (MLCP) estimator and derives a closed form for different bit widths L and column information w . Compared with the exhaustive simulation strategy, MLCP estimator substantially reduces simulation time and easily adjusts accuracy based on mathematical derivations. Unlike previous conditional-probability methods, MLCP uses entire nonzero code, namely MLCP, to calculate the truncation error and achieve higher accuracy levels. Furthermore, the simple and small MLCP compensated circuit is proposed in this brief. The results of this brief show that the proposed MLCP Booth multipliers achieve low-cost high-accuracy performance

IV. PROPOSED WORK

The proposed work entirely on to reduce absolute error and to reduce delay as well possible by using MLCP with suitable 5-2 compressor circuit as shown in Figure 2

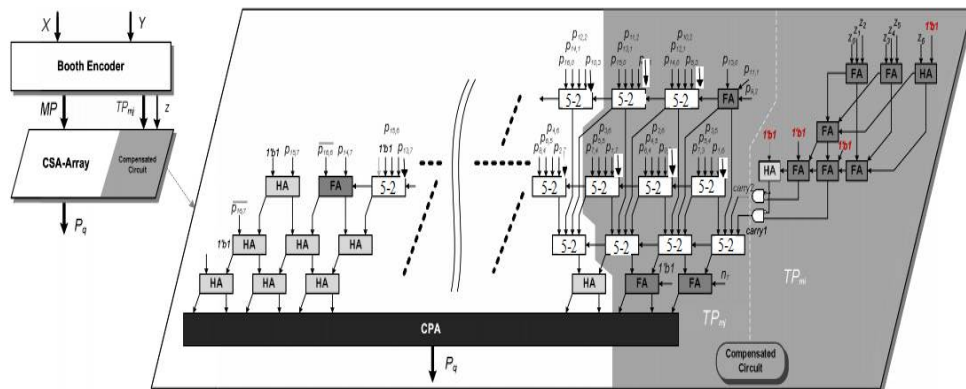


Figure2: Proposed Architecture by using 5-2 compressor

A) MAJOR REQUIREMENTS

The major requirements are needed for do the project is 5-2 compressor it consists of 5 inputs and 2 outputs and main working is it compress the given bits and another important adders are half-adder and full-adder. By using XOR-XNOR gates as well as multipliers used in latest 5:2 compressors. The main objective is to decrease the number of

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

partial products. One more important circuit is compensated circuit as shown in figure1 after performing the multiplications partial products are formed, then modified booth encoder, it performs the booth encoding according booth modified table, then after that compensated circuit performs compensate partial products so that delay will be decreased and speed will be gradually decreased finally all are fed to the CPA then the output observed is P_q [1].

Another Major Requirement is CSA with compensated circuit as shown in figure1. In that the delay will be reduced by CSA. The CSA can further reduce the partial product of booth encoder into two partial products. The working of Carry Save Adder [3] is it will be performing operation where every carry required those cases it will perform the operation remains in all cases it save the carry in special memory so that delay will be reduced.

Half adder and full-adders are basic adders in digital circuits this are play major role in all digital circuits by using this we can perform arithmetic operations. Any multiplier after perform the multiplication definitely it search for adder because after forming partial products definitely adder required that's way adders play a major role in multiplication operation.

In order to increase the instruction throughput, high performance processors make extensive use of a technique called **pipelining**. A pipelined processor doesn't wait until the result from a previous operation has been written back into the register files or main memory - it fetches and starts to execute the next instruction as soon as it has fetched the first one and despatched it to the instruction register. When the simple processor described on the previous pages is performing an add instruction, there is no need for it to wait for the add operation to complete before it starts fetching the next instruction. So a pipelined processor will start fetching the next instruction from memory as soon as it has latched the current instruction in the instruction register.

Thus a **pipelined processor** has a pipeline containing a number of stages (4 or 5 was a common number in early RISC processors) arranged so that a new instruction is latched into its input register as the results calculated in this stage are latched into the input register of the following stage. This means that there will be a number of instructions (equal to the number of pipeline stages in the best case) "active" in the processor at any one time. Here Pipeline architecture same as [13-14].

V.RESULTS

The entire project working on Xilinx model simulator and the simulation wave forms as shown below figure

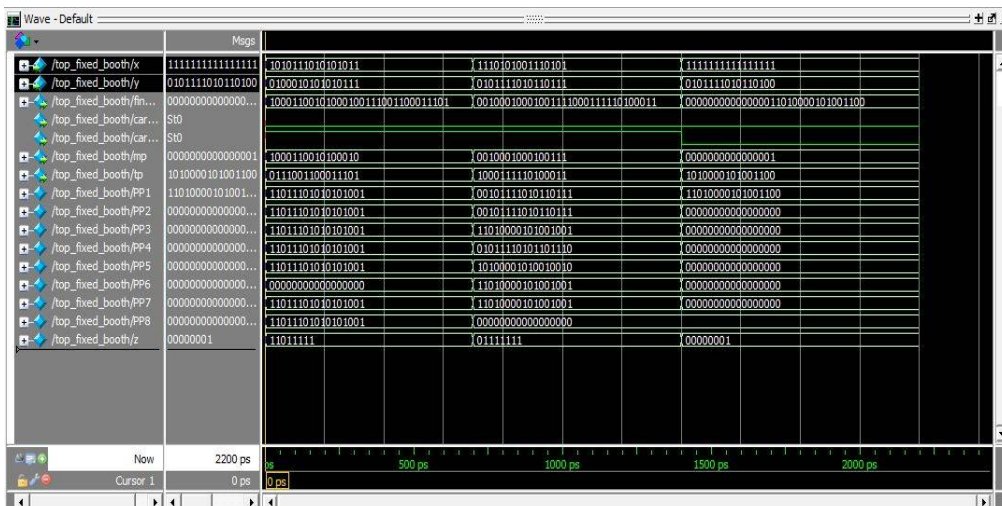


Figure3: Output waveforms for different inputs

In figure 3, it shows the outputs for different inputs those are tabled in table III. These results get from the Xilinx model simulator tool and ModelSim.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

TABLE II: COMPARISONS BETWEEN DELAYS OF MLCP WITH 4:2, 5:2 COMPRESSORS AND WITH PIPELINING ARCHITECTURE

w	Methods	L=8	L=16
2	MLCP with 4:2 compressors	8.109(ns)	10.294(ns)
3	MLCP with 5:2 compressors	7.45(ns)	8.452(ns)
3	PIPELINING	6.21(ns)	6.01(ns)

Table II shows that delay is decreased with compare to [1]. So it is gives the faster output and accurate results

Table III: ERROR OCCURS IN PROPOSED METHOD

X	Y	Actual output	Output of proposed method	Error
$(11100000)_2$	$(11111011)_2$	$(56224)_{10}$	$(11011100)_2$ $(56320)_{10}$	$(96)_{10}$
$(10101010)_2$	$(10101010)_2$	$(28900)_{10}$	$(01110001)_2$ $(28928)_{10}$	$(28)_{10}$

In table III show the error value with compare to actual multiplication value. So error is low with compare to [1]. Therefore it gives accurate output. Input X & Y values are in binary format and column 3 &4 are in decimal values.

VI.CONCLUSION

Many digital applications like FFT and convolution and other applications also the errors are play major role. So that very important to need reduce those errors, in that the main errors are truncation errors and absolute errors mainly occur at while computing partial products. The main motivation of this project is to reduce absolute error so that consider 5-2 compressor with CSA then we observed results in model simulator is less delay after performing multiplication and less absolute error. We are working on this project to try observing many points.

ACKNOWLEDGMENT

I am thankful to Dr. K. Babulu, Professor, ECE, at JNTUK-Kakinada. I am feeling glad for your kind support throughout my project Work. I express deep sense of gratitude to Dr. K. Padma Priya, Professor, HOD, ECE, JNTUK-Kakinada.

REFERENCES

- [1] Y.H.Chen, "An accuracy adjustment fixed-width booth multiplier based on multilevel conditional probability," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 1, pp. 230-207, Jan. 2015.
- [2] C. H. Chang and R. K. Satzoda, "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, Dec. 2010.
- [3] N. Petra, D. D. Caro, V. Garofalo, E. Napoli, and A. G. M. Strollo, "Truncated binary multipliers with variable correction and minimum mean square error," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 6, pp. 1312-1325, Jun. 2010.
- [4] N. Petra, D. D. Caro, V. Garofalo, E. Napoli, and A. G. M. Strollo, "Design of fixed-width multipliers with linear compensation function," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 5, pp. 947-960, May 2011.
- [5] I. C. Wey and C. C. Wang, "Low-error and hardware-efficient fixedwidth multiplier by using the dual-group minor input correction vector to lower input correction vector compensation error," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 10, pp. 1923-1928, Oct. 2012.
- [6] S. J. Jou, M. H. Tsai, and Y. L. Tsao, "Low-error reduced-width Booth multipliers for DSP applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 50, no. 11, pp. 1470-1474, Nov. 2003.
- [7] H. A. Huang, Y. C. Liao, and H. C. Chang, "A self-compensation fixed-width Booth multiplier and its 128-point FFT applications," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2006, pp. 3538-3541.
- [8] Y. H. Chen, T. Y. Chang, and R. Y. Jou, "A statistical error-compensated Booth multiplier and its DCT applications," in *Proc. IEEE Region*



ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

10Conf., Nov. 2010, pp. 1146–1149.

[9] T. B. Juang and S. F. Hsiao, “Low-error carry-free fixed-width multipliers with low-cost compensation circuits,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 6, pp. 299–303, Jun. 2005.

[10] K. J. Cho, K. C. Lee, J. G. Chung, and K. K. Parhi, “Design of low-error fixed-width modified Booth multiplier,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 5, pp. 522–531, May 2004.

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

[12] Y. H. Chen, C. Y. Li, and T. Y. Chang, “Area-effective and power efficient fixed-width Booth multipliers using generalized probabilistic estimation bias,” *IEEE J. Emerging Sel. Topics Circuits Syst.*, vol. 1, no. 3, pp. 277–288, Sep. 2011.

[13] Peipei Zhou; Hyunseok Park; Zhenman Fang; Jason Cong; André DeHon, “Energy Efficiency of Full Pipelining: A Case Study for Matrix Multiplication”, IEEE conference Pages: 172 - 175, DOI: 10.1109/FCCM.2016.50

[14] Denis Freire Lopes Nunes; Silvio Roberto Fernandes “Software Pipelining in a Non-Conventional Architecture to Improve Performance” IEEE Trans. IEEE Latin America Transactions vol. 14, no. 5, pp. 2491-2497, Aug. 2016.