International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 06 | June -2020 ISSN: 2582-3930

Design of Low Power High Speed MultiplierUsing SPST Technique

C.Vinay[1], K.Sai keerthi[2], N.Rahul[3], K.Praveen[4]
[123]B.E student, [4]Assistant professor, ECE Dept.
Department of Electronics and Communication Engineering
Matrusri Engineering College, Saidabad, Hyderabad 500 059.

Abstract

With the recent rapid advances in multimedia and communication systems, real-time signal processing like audio signal processing, video/image processing, or large-capacity data processing are increasingly being demanded. The multiplier is the essential elements of the digital signal processing such as filtering, convolution, and Inner products. The objective of a good multiplier is to provide a physically compact, good speed and low power consumption. In this project, we proposed a high speed and low-power multiplier by adopting the new SPST implementing approach. This multiplier is designed by equipping the Spurious Power Suppression Technique (SPST) on a modified Booth encoder which is controlled by a detection unit using an AND gate. The modified booth encoder will reduce the number of partial products generated by a factor of 2. The SPST adder will avoid the unwanted addition and thus minimize the switching power dissipation. In this project for simulation we use Model sim for logical verification, and further synthesizing it on Xilinx-ISE.

Keywords: *multiplier, low power, spurious power suppression technique, modified booth encoder.*

I. INTRODUCTION:

Fast multipliers are the essential elements of the digital signal processing such as filtering, convolution. Most digital signal processing methods use nonlinear functions such as discrete cosine transform or discrete wavelet transform. Because they are basically accomplished by repetitive application of multiplication and addition, the speed of the multiplication and addition arithmetic's determines the execution speed and performance purpose processors today, especially since the media processing took off. In the past multiplication was generally implemented via a sequence of addition, subtraction, and shift operations. Multiplication can be considered as a series of repeated additions. The number to be added is the multiplicand, the number of times that it is added is the multiplier, and the result is the product. Each step of addition generates a partial product. In most computers, the operand usually contains the same number of bits. When the operands are interpreted as integers, the product is generally twice the length of operands in order to preserve the information content. It is possible to decompose multipliers into two parts. The first part is dedicated to the generation of partial products, and the second one collects and adds them. The basic multiplication principle is twofold i.e. evaluation of partial products and accumulation of the shifted partial products. It is performed by the successive additions of the columns of the shifted partial product matrix. The 'multiplier' is successfully shifted and gates the appropriate bit of the 'multiplicand'. The delayed, gated instance of the multiplicand must all be in the same column of the shifted partial product matrix. They are then added to form the product bit for the particular form. Multiplication is therefore a multi operand operation. To extend the multiplication to both signed and unsigned numbers, a convenient number system would be the representation of numbers in two's complement format. Booth multiplication is a technique that allows for smaller, faster multiplication circuits, by recoding the numbers that are multiplied. It is the standard technique used in chip design, and provides significant improvements over the "long multiplication" technique. A standard approach that might be taken by a novice to perform multiplication is to "shift and add", or normal "long multiplication". That is, for each column in the multiplier, shift the multiplicand the appropriate number of columns and multiply it by the value of the digit in that column of the multiplier, to obtain a partial product. The partial products are then added to obtain the final result.

2. MODIFIED BOOTH ENCODER

In order to achieve high-speed, multiplication algorithms using parallel counters, such as the modified Booth algorithm has been proposed, and some multipliers based on the algorithms have been implemented for practical use. This type of multiplier operates much faster than an array multiplier for longer operands because its computation time is proportional to the logarithm of the word length of operands. Booth multiplication is a technique that usually allowed for smaller and faster multiplication circuits, by recoding the numbers that are multiplied. It is possible to reduce the number of partial products by half, by using the technique of radix-4 Booth recoding. The basic idea is that, instead of shifting and adding for every column of the multiplier term and multiplying by 1 or 0, we only take every second column, and multiply by ± 1, ± 2, or 0, to obtain the same results. The advantage of this method is the halving of the number of partial products. To Booth recode the multiplier term, we consider the bits in blocks of three, such that each block overlaps theprevious block by one bit. Grouping starts from the LSB, and the first block only uses two bits of the multiplier. Figure 3 shows the grouping of bits from the multiplier term for use in modified booth encoding.

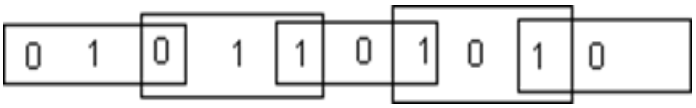

Figure 1. Grouping bits from the multiplier term

Each block is decoded to generate the correct partial product. The encoding of the multiplier *Y*, using the modified booth algorithm, generates the following five signed digits, –2, –1, 0, +1, +2. Each encoded digit in the multiplier performs a certain operation on the multiplicand, *X*, as illustrated in Table 1.

Table1
Booth encoding table

Block	Re-coded digit	Operation of X
000	0	0X
001	+1	+1X
010	+1	+1X
011	+2	+2X
100	-2	-2X
101	-1	-1X
110	-1	-1X
111	0	0X

3. SPURIOUS POWER SUPRESSION TECHNIQUE

Figure 2 shows the five cases of a 16-bit addition in which the spurious switching activities occur. The 1st case illustrates a transient state in which the spurious transitions of carry signals occur in the MSP though the final result of the MSP are unchanged. The 2nd and the 3rd cases describe the situations of one negative operand adding another positive operand without and with carry from LSP, respectively. Moreover, the 4th and the 5th cases respectively demonstrate the addition of two negative operands without and with carry-in from LSP. In such cases, the results of the MSP are predictable Therefore the computations in the MSP are useless and can be neglected. The data is separated into the Most Significant Part (MSP) and the Least Significant Part (LSP). To know whether the MSP affects the computation results or not. The detection logic unit is to detect the effective ranges of the inputs.

A MSP=A [15:8]; B MSP=B [15:8]…. (1)
A AND = A [15].A [14]…..A [8]…... (2)
B AND = B [15].B [14]…..B [8]…… (3)

$$\overline{A\ NOR} = A[15] + A[14] + \cdots \ldots . A[8] \ldots\ldots (4)$$
$$\overline{B\ NOR} = B[15] + B[14] + \cdots \ldots . B[8] \ldots\ldots (5)$$

Close = ~ $((A_{AND}+A_{NOR}).(B_{AND}+B_{NOR}))$…….(6)

We derive the Karnaugh maps which lead to the Boolean equations (7) and (8) for the Carr_ctrl and the sign signals, respectively.

$$Carr\_ctrl = (C_{LSP} \oplus A_{AND} \oplus B_{AND})\ (A_{AND}+A_{NOR})\ (B_{AND}+B_{NOR}) \ldots\ldots (7)$$

$$Sign = \overline{CLSP}.\ (A_{AND}+B_{AND}) + C_{LSP}\ A_{AND}\ B_{AND} \ldots\ldots\ldots (8)$$

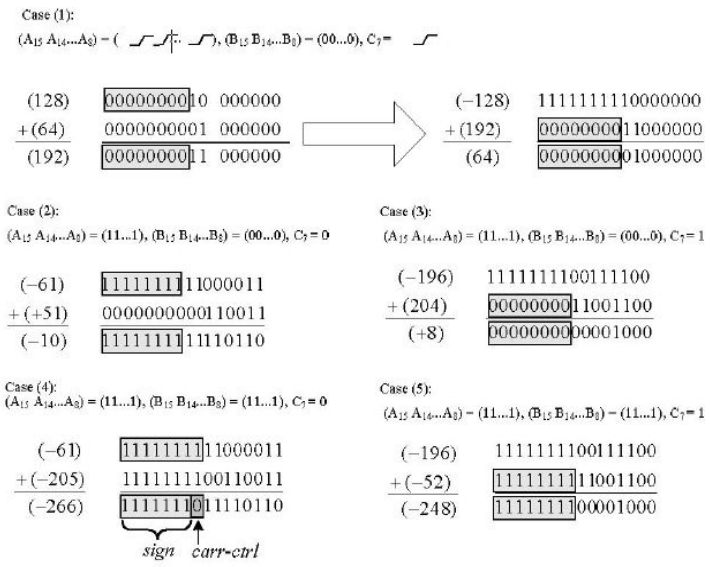

Figure 2.Transition cases in multimedia / DSP processing

4. PROPOSED SPURIOUS POWER SUPRESSION TECHNIQUE

The SPST uses a detection logic circuit to detect the effective data range of arithmetic units, e.g., adders or multipliers. When the portion of data does not affect the final computing results, the data controlling circuits of the SPST latch this portion to avoid useless data transitions occurring inside the arithmetic units. Besides, there is a data asserting control realized by using registers to further filter out the useless spurious signals of arithmetic unit every time when the latched portion is being turned on. This asserting control brings evident power reduction. Figure 3 shows the design of low power adder/subtractor with SPST. Adder /subtractoris divided into two parts, the most significant part (MSP) and the least significant part (LSP). The MSP of the original adder/subtractor is modified to include detection logic circuits, data controlling circuits, sign extension circuits, logics for calculating carry in and carry out signals. The most important part of this study is the design of the control signal asserting circuits, denoted as asserting circuits in Figure 3. Although this asserting circuit brings evident power reduction, it may induce additional delay. There are two implementing approaches for the control signal assertion circuits. The first implementing approach of control signal assertion circuit is using registers. This is illustrated in Figure 4. The three output signals of the

detection logic were close signal, Carr_ctrl signal, sign signal. The three output signals the detection logic unit are given a certain amount of delay before they assert. The delayϕ, used to assert the three output signals, must be set in a range of ψ<ϕ<Δ, whereψ denotes the data transient period and ΔDenotes the earliest required time of all the inputs. This will filter out the glitch signals as well as to keep the computation results correct. The restriction that ϕ must be greater than ψ to guarantee the registers from latching the wrong values of control usually decreases the overall speed of the applied designs.

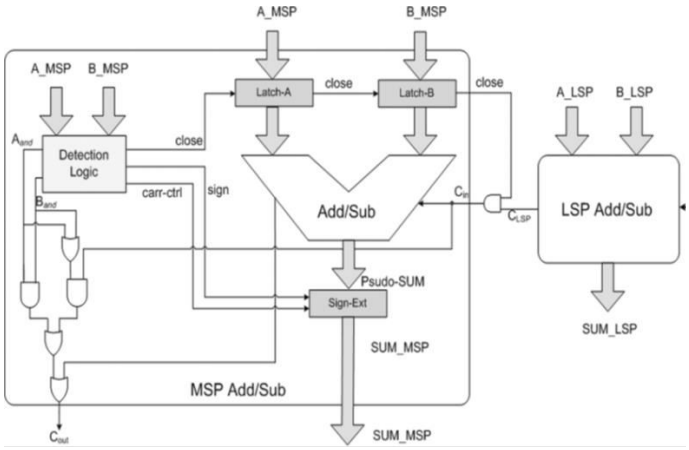

Figure 3. Low power Adder/subtractor adopting the SPST

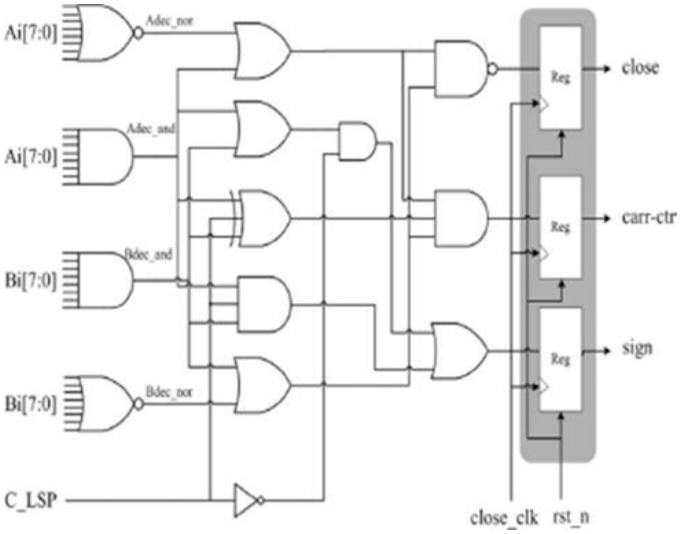

Figure 4. Detection Logic circuits using Registers

This issue should be noticed in high- end applications which demands both high speed and low power requirements. To solve this problem we adopt the other implementing approach of control signal assertion circuit using AND gate.

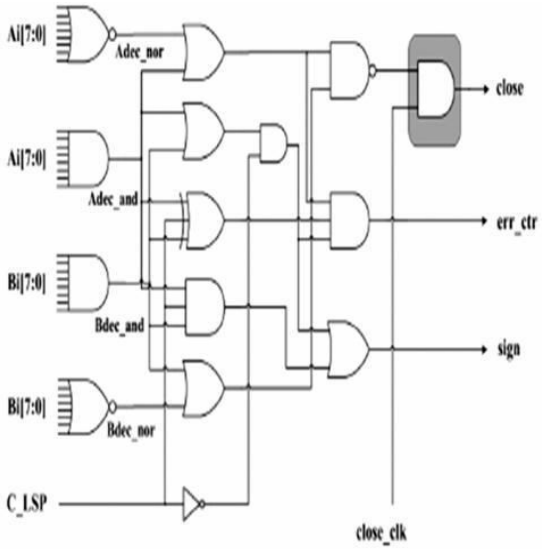

Figure 5.Detection Logic circuits using AND gate

5. PROPOSED LOW POWER HIGH SPEED MULTIPLIER

The proposed high speed low power multiplier is designed by equipping the SPST on a tree multiplier. There are two distinguishing design considerations in designing the proposed multiplier as listed in the following

A. Applying the SPST on the Modified Booth Encoder

Figure 6 shows a computing example of Booth multiplying two numbers “2AC9” and “006A”. The shadow denotes that the numbers in this part of Booth multiplication are all zero so that this part of the computations can be neglected. Saving those computations can significantly reduce the power consumption caused by the transient signals. According to the analysis of the multiplication shown in figure 6, we propose the SPST-equipped modified-Booth encoder, which is controlled by a detection unit. The detection unit has one of the two operands as its input to decide whether the Booth encoder calculates redundant computations as shown in Figure 7. The latches can, respectively, freeze the inputs of MUX-4 to MUX-7 or only those of MUX-6 tMUX-7 when the PP4 to PP7 or the PP6 to PP7 are zero; to reduce the transition power dissipation. Figure 8. Shows the booth partial product generation circuit. It includes AND/OR/ EX-OR logic.

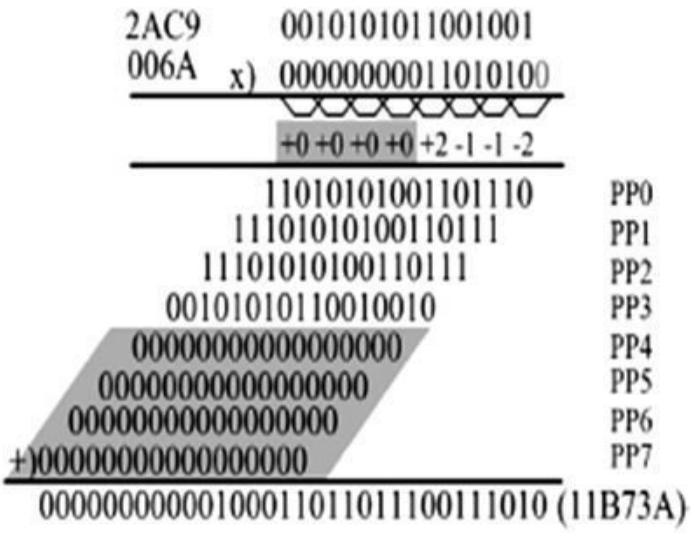

Figure 6.Illustration of Multiplication Using Modified Booth Encoding

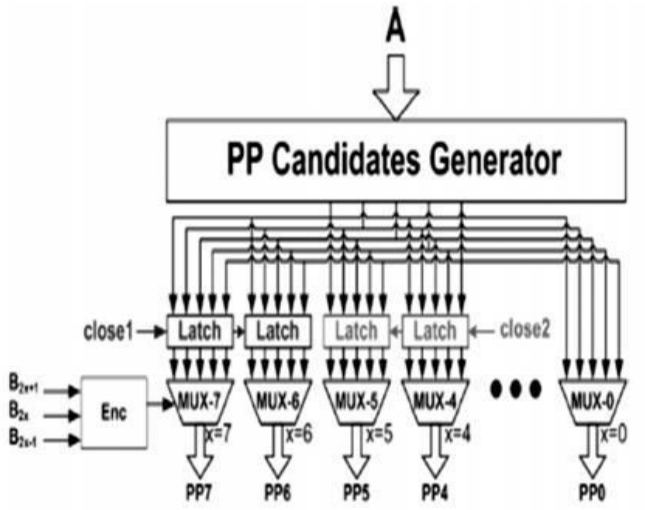

Figure 7.SPST Equipped Modified Booth Encoder

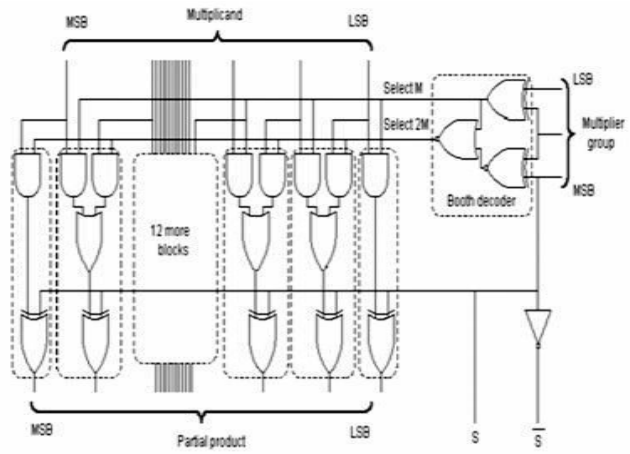

Figure 8.Booth Partial Product Selector Logic

B. Applying the SPST on the Compression Tree

The proposed SPST -equipped multiplier is illustrated in figure 9. The PP generator generates five candidates of the partial products, i.e., {–2*A*,–*A*, 0, *A*, 2*A*}. These are then selected according to the Booth encoding results of the operand *B*. When the operand besides the Booth encoded one has a small absolute value, there are opportunities to reduce the spurious power dissipated in the compression tree.

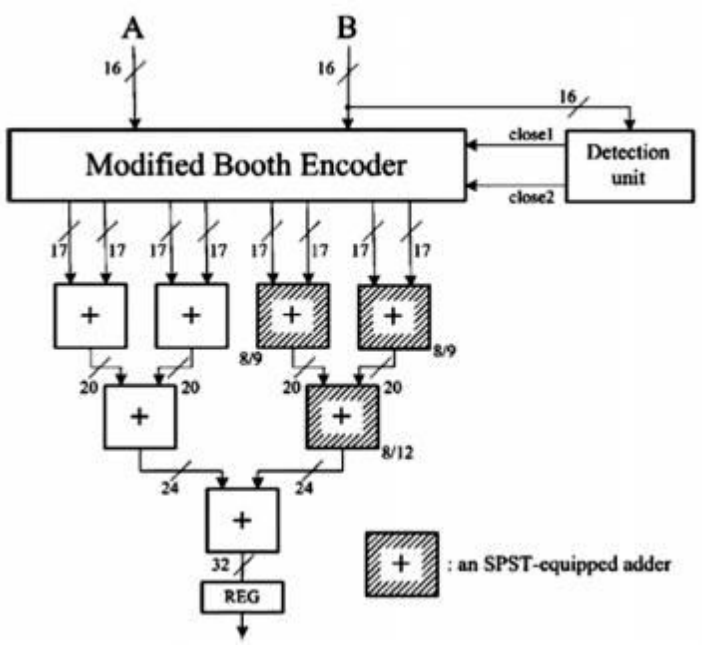

Figure 9.Proposed High Performance Low Power Equipped Multiplier

6. COMPARISION

The Proposed multiplier results are very good when compared with the existing methods and comparison table is given by following table.

Table 2

Multiplier type	CSA MULTIPLIER	SPST MULTIPLIER
vendor	Xilinx	Xilinx
family	Spartan 6	Spartan 6
Estimated delay	28.6ns	14.51ns
Power dissipation	0.218mw	0.082mw

7. RESULTS

In this project we are evaluating the performance of the proposed high speed low power multiplier by

using VHDL coding.Power report and delay report we are synthesizing these multipliers using Xilinx.

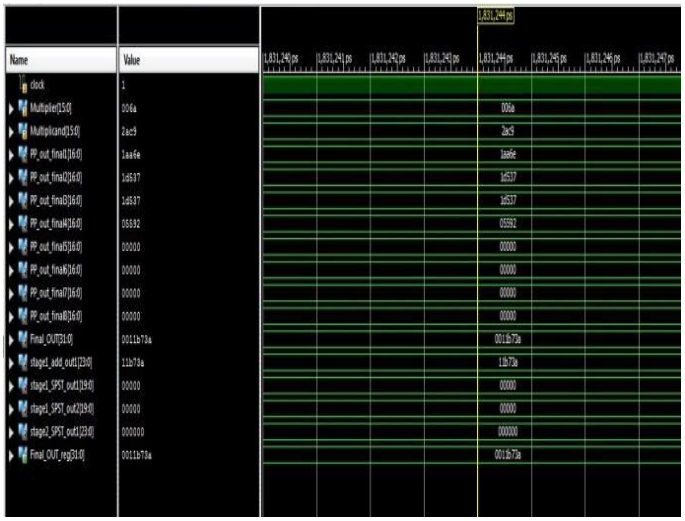

Figure 10.Simulation results of proposed multiplier

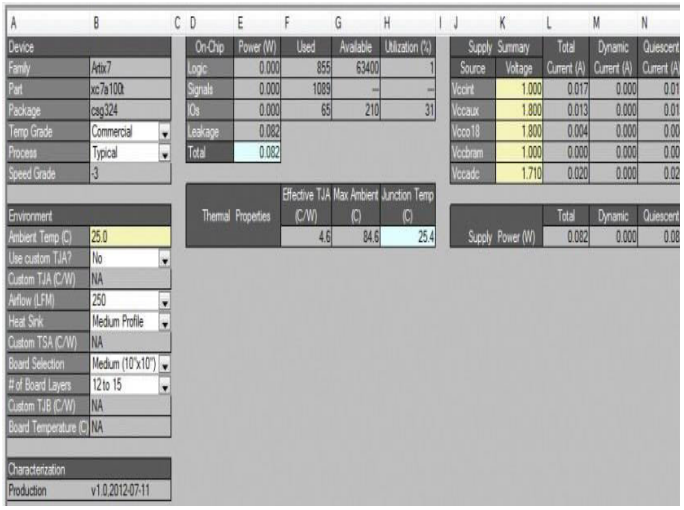

Figure 11. Power report of proposed multiplier

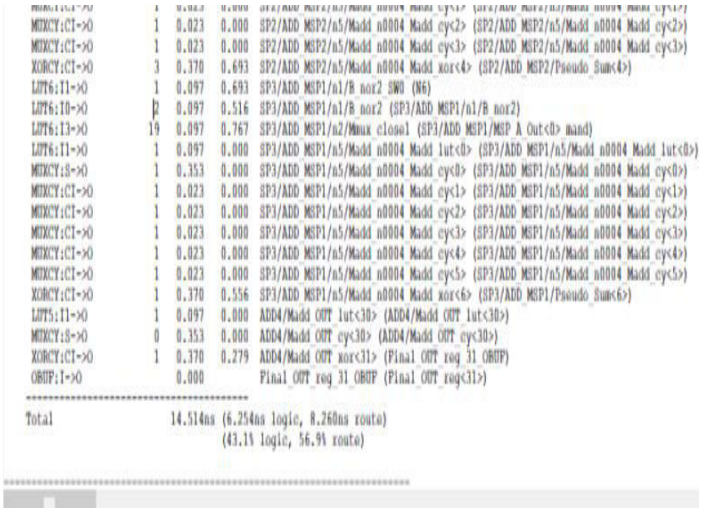

Figure 12. Delay analysis of proposed multiplie

8. CONCLUSION

In this project, we propose a high speed low-power multiplier adopting the new SPST implementing approach. The proposed multiplier isdesigned by equipping the Spurious Power Suppression Technique (SPST) on a modified Booth encoder which is controlled by a detection unit using an AND gate. The modified booth encoder will reduce the number of partial products to half .TheSPST adder will avoid the unwanted addition and thus minimize the switching power dissipation. The SPST multiplier implementation using a detection logic uses an AND gates have an extremely high flexibility on adjusting the data asserting time. This facilitates the robustness of SPST can attain speed improvement and power reduction in the Modified booth encoder. This design can be verified using Modelsim and Xilinx using Verilog.

9. REFERENCES

[1] Sathya, A.; Fathimabee, S.; Divya, S., "Parallel multiplier-accumulator based on radix-2 modified Booth algorithm by using a VLSI architecture," Electronics and Communication Systems (ICECS), 2014 International Conference on , vol., no., pp.1,5, 13-14 Feb. 2014

[2] Suriya, T.S.U.; Rani, A.A., "Low power analysis of MAC using modified booth algorithm," Computing, Communications and Networking Technologies (ICCCNT),2013 Fourth International Conference on , vol., no., pp.1,5, 4-6 July 2013

[3] Naveen Kumar, Manu Bansal, AmandeepKaur, “Speed Power and Area Efficient VLSI Architectures of Multiplier and Accumulator,” International Journal of Scientific & Engineering Research Volume 4, Issue 1, January-2013

[4] J. Y. YahwanthBabu, P. Dinesh Kumar, “A new multiplier Accumulator architecture based on high accuracy modified Booth algorithm,” International journal of advanced research in computer Engineering & Technology, vol.2,no.3,pp.1036-1040,Mar 2013.

[5] RashmiRanjan et al., "A New VLSI Architecture Of Parallel Multiplier Based On Radix-8 Modified Booth Algorithm Using VHDL," International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 3 No. 4 April,2012
[6] B. Jyothirmai, M. Premalatha, "Design and implementation of parallel MAC by Booth algorithm," International Journals of scientific Research,vol.1,no.6, pp.78-79, Nov.2012.
[7] AdalankiPurna Ramesh, Dr. A. V. N. Tilak, Dr. A. M. Prasad, "Efficient implementation of 16 bit multiplier, Accumulator using Radix – 2 modified Boothalgorithm and SPST adder using Verilog," International journal of VLSI design and communication systems, vol. 3, no.3, pp.107-118, Jun. 2012.
[8] PratibhadeviTapashetti, Dr. Praveen Kumar, "Compatible Architecture of MAC, Based on Modified Booth Algorithm". International Journal of Emerging Technology and Advanced Engineering. Oct 2017.

[9] Harilal, Durga Prasad, "High speed arithmetic Architecture of parallel multiplier Accumulator based on Radix-2 modified Booth algorithm," International journals of computational Engineering research, vol.2, no.8, pp.28-38, Dec.2012.
[10] V.ThirumalaRao, S. Girish Gandhi, C.Leela Mohan, "Performance Evaluation of Parallel Multipliers for High Speed MAC Design" ,International Journal of Innovations in Engineering and Technology (IJIET). ISSN:2319-1058.
[11] Sareddy. Sindhuja Reddy, CH. BhanuPrakash. "Design of a Novel Multiplier and Accumulator using Modified Booth Algorithm with Parallel Self-Time Adder". International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
[12] A. R. Cooper, "Parallel architecture modified Booth multiplier," Proc.Inst. Electr. Eng. G, vol. 135, pp. 125–128, 1988
[13] Young Ho seo, Dong wookkim, "A new VLSI Architecture of parallel multiplier Accumulator based on Radix 2 Modified Booth algorithm," IEEE transactions on very large scale integration (VLSI) systems, vol. 18, no.2, pp. 201-208 , Feb.2010.
[14] P. SasiBala& S. Raghavendra, "A New VLSI Architecture of Parallel Multiplier –Accumulator Based on Radix-2 Modified Booth Algorithm".International Journal of Instrumentation, Control and Automation (IJICA) ISSN: 2231-1890 Volume-1, Issue-2, 2011.

[15]. Pettam Ramesh, K. Uma Maheshwari, P.N, V. PurnaChanderRao. "A New VLSI Architecture Design based on Radix-2 Modified Booth Algorithm using ParallelMultiplier–Accumulator", International Journal of Scientific Engineering and Technology Research, 2319-8885, Vol.02, Issue.13, October-2013, Pages:1406-1415.
[16] S. Waser and M. J. Flynn, Introduction to Arithmetic for Digital Systems Designers. "New York: Holt, Rinehart and Winston, 1982."