

# An Efficient Multiplier Design with Dadda Algorithm using OFA

B.Ravina<sup>1</sup> P.Anjaneya<sup>2</sup> O.Homakesav<sup>3</sup> G.K.Rajini<sup>4</sup>  
 P.G. Student, Research Scholar, Research Scholar, Associate Prof.  
 AITS, School of Electronics Engineering, VIT University, Vellore, India

**Abstract**

This paper presents Dadda algorithm, the model of 8 bit multiplier having less power and more speed is designed, using fundamental building block as OFA, which possess low power dissipation and minimum transmission delay The OFA is designed with switch logic which is a less power logic and it has less number of transistors to implement the digital circuit design, by comparing with CMOS logic. This paper gives comparative analysis of conventional Dadda algorithm using conventional 3:2 compressor and proposed Dadda algorithm using OFA using Xilinx 14.7 for simulation and synthesis.

**Keywords\_** Dadda Algorithm, OFA, 8x8 Bit Multiplier.

**I.INTRODUCTION**

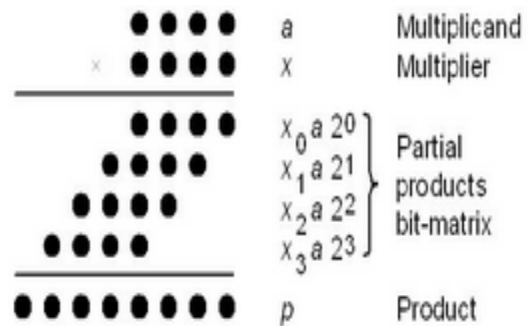
Multiplication is the essential method which is used in different electronic and in various digital communication application ,multipliers with less latency and minimum power dissipation are preferred to propose an optimized circuit by the process, within minimum response time we can achieve maximum result. Fundamental units used in multipliers are 3:2 and 2:2. Various design executions of FA and HA circuit which comprises technology apart from this various multiplier algorithm also have been used to attain optimized power and product which includes Dadda ,Wallace-tree, and vedic and booth algorithms. The latest multiplier used, reduced sp-D3L sum adder and that the algorithm technique. When compared to past design it functions at high bandwidth and consume low power, but still we need to diminish the power significantly, so, it will help in the larger circuit at where multipliers are fundamental evolutionary blocks(2).In proposed work ,a multiplier in its partial result addition method ha and 3:2 compressor are used as computational blocks, to diminish the power consumption by using switch logic and CMOS technology process. The process of using Dadda algorithm is diminish the transmission delay of the multiplier. This paper is arranged according to this flow reported the literature survey of the multiplier in sect.II; Dadda algorithm described in sect.III block diagram are

implemented in sect.IV ; and proposed system in sect.V results in sect.VI and conclusions in sect.VII.

**II.LITERATURE SURVEY**

Multiplier is an essential hardware block in digital signal processing. To perform multiplication in a easy way by using a one two input adder. By using an N bit adder, N and M bits wide are perform input, M cycles function as multiplier. Both shift and algorithm as multiplication ties jointly M incomplete result.The half-done result is created by bit by bit multiplicand of the multiplier(3). By changing the outcome into the foundation of the multiplier bit position and that important is an AND operation. The weight of the multiplier bit position is base on the addition of shift version of the multiplicand it involves as binary multiplication, each familiar long-hand radix-10 multiplication. In fact, binary multiplication is compared to decimal multiplication is very easy to perform .The result of each bit of a binary digit is either 0 or 1, thus, depending on the result of the multiplier bit, the half of the result can be same of the multiplicand, or 0,AND function is an simply digital logic.

The most disadvantage in conventional multiplier is take out propagation delay



**Fig1:Two 4-bit unsigned multiplication in dot representation**

### III.DADDA ALGORITHM

Wallace tree multiplier be nearly designed like hardware multiplier is a dadda multiplier. Dadda multipliers performs as few diminishes as likely compared to Wallace tree multipliers to achieve the decrease as much as possible, on each layer. Due to this, actually Dadda multipliers are perform low reduction phase, but feasibly little bit longer, thus more rather than bigger adder. Starting stage of the column compression tree, and extra columns in the next levels of the multipliers is simplest some columns are compressed. In the figure 2, it show the dot-diagram of an 8x8 Dadda multiplier (1). Two dots are connecting in plain diagonal lines is represents output of the full adder, although two dots are connecting in cross diagonal lines is represents output of the half adder.

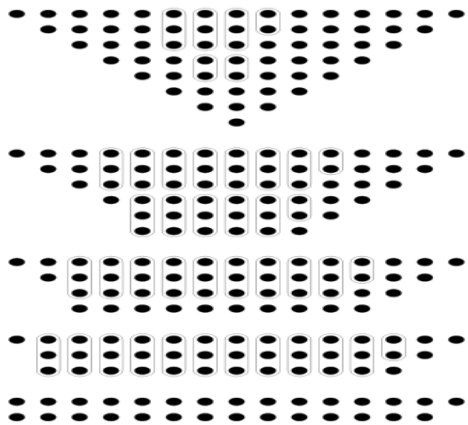


Fig2 :Operation of dadda multiplier

The Dadda algorithm includes the following steps. It follows the three steps to complete the Dadda algorithm.

Step:1

Let  $K_1=2$  and  $K_{i+1}=(1.5 \times K_i)$ , where  $K_i$  is the matrix distance upwards of the  $i^{th}$  stage from the end. To find the minimum 'i' such that atleast single column of the actual partial result matrix has additional than  $k_i$  bits.

Step:2

In the  $i^{th}$  step from the end, take on (2,2) and (3,2) counters to achieve a diminished matrix with no additional than  $k_i$  bits in any column.

Step:3

Let  $i=i-1$  and perform again step 2 up to a matrix with only two rows is produced.

The number of limiting stages need to execute Dadda structural design for different number of bits is stated in the below table.

TABLE 1: Number of steps in Dadda multiplier

Bits in Multiplier(N)	Number of Stages
3	1
4	2
$5 \leq N \leq 6$	3
$7 \leq N \leq 9$	4
$10 \leq N \leq 13$	5
$14 \leq N \leq 19$	6
$20 \leq N \leq 28$	7
$29 \leq N \leq 42$	8
$43 \leq N \leq 63$	9
$63 \leq N \leq 94$	10

### IV.BLOCK DIAGRAM

The proposed multiplier of the block diagram is follows in the below figure 3:

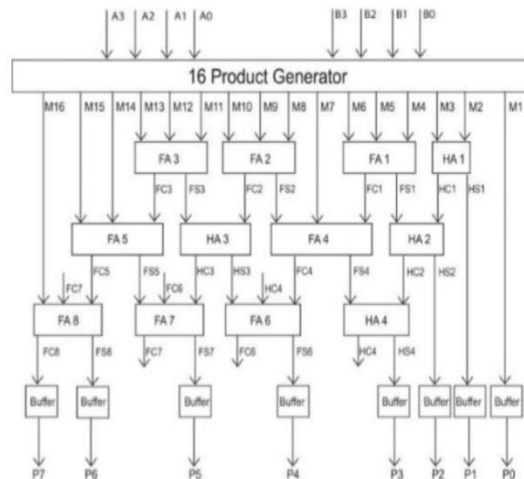


Fig3: Proposed multiplier block diagram

The proposed representation of the working stages are as follows:

1. In the initial step, 16 partial product are produced using AND gate.
2. In the second step, the height of the tree is diminished Dadda stage using three 3:2 one half adder.

3. In the third step, the scaling down is done by two 3:2 and two half adder.
4. At the ending stage of, Dadda we have used a RCA.
5. In the end the results are proceed throughout the buffer to make the output voltages better.

### V. PROPOSED MULTIPLIER

The proposed design of the multiplier uses the block which are given below :

#### A. Full Adder

3:2 compression is a most important block in the multiplier. In proposed model full adder has been reform in order to produce minimum transmission delay and less power dissipation of the circuit, so, that it can further be used to design multiplier having optimized parameters including power, delay and low layout area. with this proposed model reform of XOR module in the 3:2 is composed of four transistors(7). XOR module of the proposed schematic is follows in the below figure4. we have transpose one input i.e; B input because the B and B' will be used in the next two transistors(13). we have cascaded a NMOS and PMOS while relate the second input A to the gate terminal of both NMOS and PMOS and rather of connecting the source of NMOS to the inverted first input i.e.B' similarly we have connected the source of PMOS to vdd we connected it first input B, which form XOR module. The XOR module equation is stated as(6) :

$$\text{XOR} = A'B + AB'$$

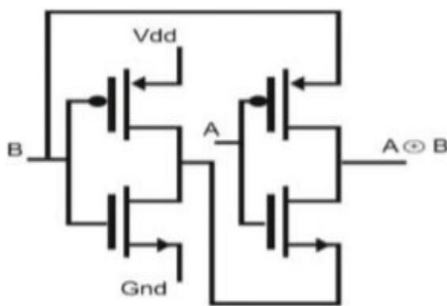


Fig4: XOR module

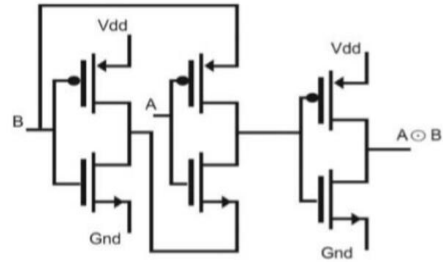


Fig5: XNOR module

The above figure5 shows the XNOR module by just inverting the XOR module output. The XNOR module equation is stated as :  
 $\text{XNOR} = AB + A'B'$

The 3:2 compressor sum is created by second XOR module. As reported above in the instance, we have connected the NMOS source to the XNOR output, while that of PMOS shows to the XOR module output.

For producing the carry we have used two pass transistors. One pass transistor we have applied an input B, while on the second pass transistor we have applied an input C. The 3:2 compressor of the schematic illustration is follows in below figure6. The FA equation is stated as:

$$\text{Sum} = A \text{ XOR } B \text{ XOR } C$$

$$\text{Carry} = AB + BC + CA$$

The equation of the carry is further diminished to if  $A = B$  then  $\text{carry} = A$  else  $\text{carry} = C$

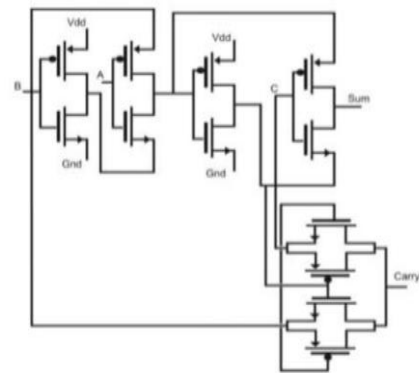


Fig6: Proposed full adder

#### B. Half Adder

A 2:2 compressor is one more building block of the multiplier. In multiplier design we can use total four HA the illustrative of the 2:2 is follows in below figure 7. The equation for the HA is stated as :

$$\text{sum} = A \text{ XOR } B$$

$$\text{Carry} = AB$$

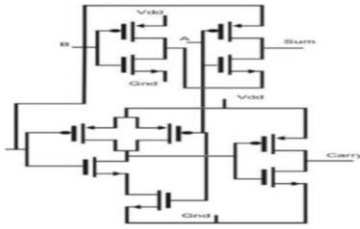


Fig7:Half adder

## VI. RESULT

	conventional 8-bit	proposed 8-bit
Delay	25.229ns	3.49ns

## VII. CONCLUSION

In this paper, an efficient multiplier is designed using Dadda Algorithm. In this algorithm addition of partial products are performed using an OFA, which has less number of transistors when compared to conventional full adder, to reduce the power.

## VIII. REFERENCES

- [1] Zain Shabbir, Anas Razzaq Ghumman, Shabbir Majeed Chaudhry, A Reduced-sp-D3Lsum Adder-Based High Frequency  $4 \times 4$  Bit Multiplier Using Dadda Algorithm, Springer Science+Business Media New York 2015.
- [2] Jorge Juan Chico, Enrico Macii, book chapter 17, "Power-Consumption Reduction in Asynchronous Circuits Using Delay Path Unequalization, page (151-160)", 13th International Workshop, PATMOS 2003 Turin, Italy, September 10-12, 2003, Proceedings.
- [3] Raminder Preet Pal Singh, Parveen Kumar, Balwinder Singh, "Performance Analysis of 32 bit Array Multiplier with Carrying Save Adder and with CarryLook ahead adder", in International Journal of Recent Trends in Engineering, Vol2, No. 6, November 2009
- [4] B.Eghbalkhah, B. Afzal, and A. Afzali-Kusha, "Speed improvement algorithm for  $16 \times 16$  multipliers using sizing optimization," 2007 International Conference on Design & Technology of Integrated Systems in Nanoscale Era, Rabat, 2007, pp. 98-101.
- [6] doi: 10.1109/DTIS.2007.4449500
- [7] Prashant D. Pawale, Venkat N Ghodke, High-speed Vedic multiplier design and implementation on FPGA, IJAR 2015(Pune, India).
- [8] M.Aguirre-Hernandez, M. Linares-Aranda, CMOS full-adders for energy-efficient arithmetic applications.IEEE Trans.Very Large Scale Integr. (VLSI) Syst. 19(4), 718–721 (2011).
- [9] V.Kayathri, C. Kumar, P. Mari Muthu, N. Naveen Kumar, POWER REDUCTION IN FULL ADDER USING NEW HYBRID LOGIC, International Journal of Current Research and
- [10] Modern Education (IJCRME)2016,(Coimbatore, Tamilnadu).
- [11] M.Nisha Angeline, S. Valarmathy, Implementation of N-Bit Binary Multiplication Using N – 1 Bit Multiplication Based on

- [12] Nikhilam Sutra and Karatsuba Principles Using Complement Method, Scientific research publishing2016,( Sathyamangalam, India).
- [13] Dursun Baran, Mustafa Aktan, Vojin G. Oklobdzija, Multiplier Structures for Low Power Applications in Deep-CMO.
- [14] M.Margala, S. Purohit, Investigating the impact of logic and circuit implementation on full adder performance. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 20(7), 1327–1331 (2012).
- [15] M.B.Damle, Dr. S. S. Limaye (2012), "Low-power Full Adder array-based Multiplier with Domino Logic", International Journal of Advance Research in Computer Engineering & Technology, Vol.1, Issue 4, June 2012, ISSN:2278-132.
- [16] K.Kuo,C.Chou, Low power and high-speed multiplier design with row bypassing and parallel architecture. Microelectron. J. 41, 639–650 (2010).
- [17] Y.Berg, M. Azadmehr,"Novel ultra low-voltage and high-speed CMOS pass transistor logic", Proceedings of the IEEE, Faible Tension Faible Consommation (FTFC), Paris, 2012.
- [18] A.Daniel Raj, S.Vijayalakshmi, P.Sathyamoorthy, "Efficient Design of 4-bit array multiplier using GDI low power cell",International Journal of Computer Technology and Electronics Engineering (IJCTEE), Volume 2, Issue 6, PP-47-51, December2012.
- [20] Radhakrishnan, D., and A. P. Preethy. "Low power CMOS pass logic 4-2 compressor for high-speed multiplication." Circuits and Systems, 2000. Proceedings of the 43rd IEEE Midwest Symposium on. Vol. 3. IEEE, 2000