

Design of Area, Power and Delay Efficient High-Speed Multipliers

Ms. Namrata Dangat
Electronics Engg. Dept
Shah and Anchor Kutchhi Engg College
Chembur, Mumbai.
namratadangat028@gmail.com

Dr.V.C.Kotak
Electronics Engg. Dept
Shah and Anchor Kutchhi Engg College
Chembur, Mumbai.
vinitkotak@shahandanchor.com

Prof. Sushma Srivastava
Electronics Engg. Dept
Shah and Anchor Kutchhi eng
Chembur, Mumbai
sushsri2012@gmail.com

Abstract:

Multiplication is the most time consuming process in various signal processing operations like convolution, circular convolution, auto-correlation and cross-correlation. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following- high speed, low power consumption, regularity of layout and hence less area or even combination of them in multiplier. However area and speed are two conflicting constraints. So improving speed results always in larger areas. So here we try to find out the best trade off solution among the both of them. To have features like high speed and low power consumption multipliers several algorithms have been introduced. In this paper, we describe Multipliers by using various algorithm in VLSI technology. The Wallace Tree Multipliers are compared with existing multipliers in terms of improvement in features like area, delay and power consumption by using different logical operation.

Keywords — Multipliers, Adders, different logics, VLSI design.

I. INTRODUCTION

[2] [8] In the signal processing techniques the most important and also commonly used function is multiplier. Multiplication is an operation of scaling one number by another multiplication operation such as convolution, Discrete Fourier Transform, Fast Fourier transform etc. As there is need for greater computing power on battery-operated mobile devices, design emphasis has shifted from optimizing conventional delay time, area size to minimizing power dissipation while still maintaining the performance.

Traditionally multiplication algorithm “Shift and Add” has been implemented to design, which is not suitable from VLSI implementation and delay point of view. In order to achieve fast multiplication some important algorithms have been proposed in literature. These algorithms have been used in VLSI implementation of multiplier to achieve fast multiplication in circuits.

1. Different Multipliers

[2] An efficient multiplier should have following characteristics:-

Accuracy- A good multiplier should give correct result.

Speed- Multiplier should perform operation at high speed.

Area- A multiplier should occupy less number of slices.

Power- Multiplier should consume less power.

Multiplication process has three main steps:-

1. Partial product generation.
2. Partial product reduction.
3. Final addition.

In first step, the multiplicand and the multiplier are multiplied bit by bit to produce partial product. The second step is most important, as it is most complicated and determines the speed of overall multiplier. In last step, the two row outputs of the tree are added using any high speed adder to generate output result.

There are two multiplication schemes:

Serial Multiplication- It is computing a set of partial products and then summing the partial products together. So it uses a successive addition algorithm. They are simple in structure because both the operands are entered in a serial manner. The multiplication result needs to be shifted by its position in the serial chain. Each processing element takes the result of the multiply and adds it to partial sum after it has been shifted. Therefore,

the physical circuit requires less hardware and minimum amount of chip area. However, the speed performance of the serial multiplier is poor due to operands being entered sequentially. Therefore the physical circuit requires minimum amount of area and less hardware with increase in the rate of speed and delay. The multiplier bit b_i are used to control 2:1 multiplexer, if $b_i=0$, and n bit 0 word is sent to the adder, while $b_i=1$ directs the multiplicand A to the input. Output of the adder is given to the product register. And this process repeats n number of times. The factor of 2^n multiplying A is used to compensate for the 2^{-n} introduced by the right shift at the end of calculation.

Parallel Multiplication- Partial products are generated simultaneously parallel implementations are used for high performance machine, when computation latency needs to be minimized. Improvements in Partial-Product Bit Reduction Techniques and then optimization of the Final Adder for the uneven signal arrival profile from the Multiplier Tree.

Advantages of parallel multiplication: - Comparing both, parallel multiplication has more advantages than serial multiplication. As parallel type utilize lesser steps comparing serial one. So it is faster than serial multiplication.

Here we discuss about two different types of parallel multipliers which are:

- a. Array multiplier.
- b. Wallace tree multiplier.

The rest of the paper is organized as follows: Objectives is discussed in Section 2. Methods and Performances are explained in Section 3. In section 4, Different logics with their outputs and results are discussed. In section 5, the conclusion is described.

II. OBJECTIVES

In today's fast technologically developing world, the shift has been towards construction of small and portable devices. As the number of these battery operated, processor driven equipment's increase and their performance demand is expected to be more, there is a need of increasing their processing speed and reducing their power dissipation. The objective of good multiplier is to provide a compactness high speed and low power as it is core part of arithmetic processing unit.

To achieve these objectives it is good to reduce number of operations so as to reduce dynamic power and even the delay in VLSI technology.

III. METHODS AND PERFORMANCES

There are number of techniques that to perform binary multiplication. In general, the choice is based upon factors such as latency, throughput, area, and design complexity. More efficient parallel approach uses some sort of array or tree of full adders to sum partial products. Array multiplier, Wallace tree multiplier and dadda multipliers are some of the standard approaches to have hardware implementation of binary multiplier which are suitable for VLSI technology.

A. Array multiplier

This is the most basic form of binary multiplier construction. Its basic principle is exactly like that done by pen and paper. The circuit which do 'shift and add' at once such a multiplier is called as "array multiplier" or "parallel multiplier". It is called as array multiplier because it is an array of adders. An array consists of lots of elements.

[9] An array multiplier is a digital combination circuit that is used for the multiplication of two binary numbers by employing an array of full adders and half adders. This array is used for the simultaneous addition of the various product terms involved. To form the various product terms, an array of AND gates is used before the adder array. Each row of this array generates a partial product. This partial product generated value is then added with the sum and carry generated on the next row. ANDed terms generated using logic AND gate. Full adder (FA) implementation showing the two bits (A, B) and Carry In (C_i) as inputs and Sum (S) and Carry out (C_o) as outputs.

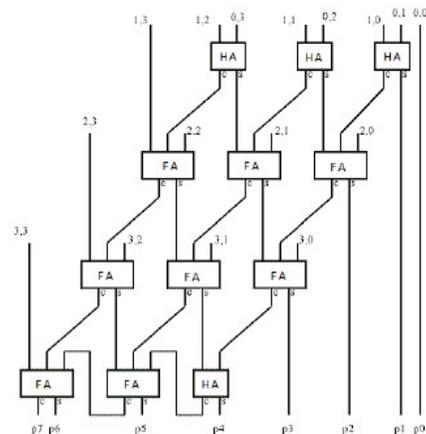


Figure 1:- Array Multiplier.

[10] Array multiplier gives more power consumption as well as optimum number of components required, but delay for this multiplier is larger. It also requires larger number of gates because of which area is also increased; due to this array multiplier is less economical. Thus, it is a fast multiplier but hardware complexity is high.

B. Wallace Tree multiplier

The reduction of partial products using full adders as carry-save adders became generally known as the “Wallace Tree”. Several popular and well-known schemes, with the objective of improving the speed of the parallel multiplier, have been developed in past. Wallace introduced a very important iterative realization of parallel multiplier. This advantage becomes more pronounced for multipliers of bigger than 16 bits. The ovals around the dots represent either a full adder (for three circled dots) or a half adder (for two circled dots). This tree is reduced to two rows for a carry-propagate adder after four stages. There are many ways to reduce this tree with CSAs, and this example is just one of them.

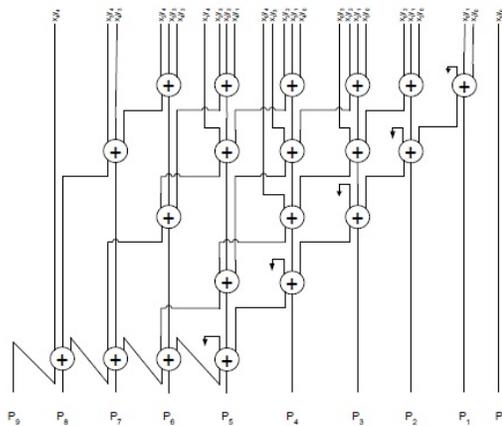


Figure 2:- Wallace Tree Multiplier.

In Wallace tree architecture, all the bits of all of the partial products in each column are added together by a set of counters in parallel without propagating any carries. Another set of counters then reduces this new matrix and so on, until a two-row matrix is generated. The final results are added using usually carry propagate adder. The advantage of Wallace tree is speed because the addition of partial products is now $O(\log N)$. A block diagram of 4 bit Wallace Tree multiplier is shown. As seen from the block diagram partial

products are added in Wallace tree block. The result of these additions is the final product bits and sum and carry bits which are added in the final fast adder (CRA).

IV. DIFFERENT LOGICS USED

A. STATIC LOGIC

[1] The most widely used logic is complementary CMOS logic due to advantages associated with it like Low sensitivity to noise, Low power consumption with no static power dissipation, Good performance and Robustness. These properties lead to implementation of large fan in logic circuits using same devices.

Static CMOS gates are implemented by using combination of two networks, the pull up network (PUN) and pull down network (PDN). Static CMOS is characterized by very good current driving capabilities and high noise margins. In Static CMOS design, at every point in time, each gate output is connected to either Vdd or Vss via a low-resistance path. A Static CMOS gate is a combination of two networks, the pull up network (PUN) and the pull down network (PDN). The function of the PDN is to provide a connection between the output and Vdd

when the output of the logic gate is supposed to be 1. Similarly, the PDN connects the output to Vss when the output is expected to be 0.

The PUN and PDN networks are constructed in a mutually exclusive manner such that one and only one of the networks are conducting in steady state. The Static CMOS gates have rail-to-rail swing, no static power dissipation. The speed of the static CMOS circuit depends on the transistor sizing and the various parasitic that are involved with it. The problem with this type of implementation is that for N fan-in gate 2N number of transistors are required, i.e., more area required to implement logic. This has an impact on the capacitance and thus the speed of the gate.

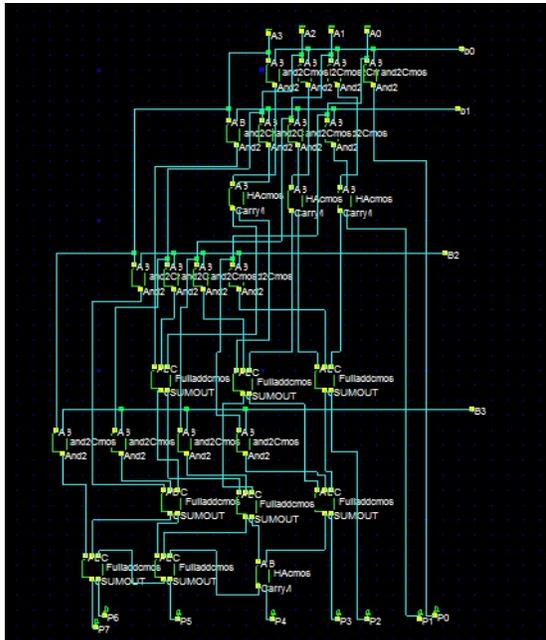


Figure 3: - 4×4 Array multiplier using static logic.

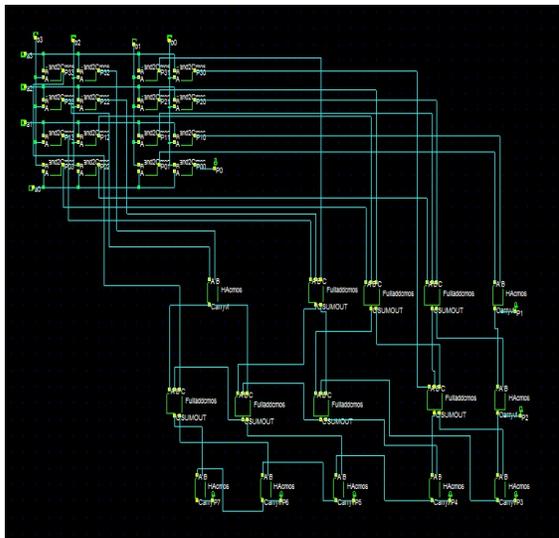


Figure 4:- 4×4 Wallace multiplier using static logic.

B. TRANSMISSION GATE LOGIC

A transmission gate is defined as an electronic element that will selectively block or pass a signal level from the input to the output. The transmission gate consists of two MOSFETs, one n-channel responsible for correct transmission of logic zeros, and one p-channel, responsible for correct transmission of logic ones. The transmission gate consists of a PMOS and NMOS connected in

parallel. Gate voltage applied to these gates is complementary of each other (C and C bar). TGs act as bidirectional switch between two nodes A and B controlled by signal C. Gate of NMOS is connected to C and gate of PMOS is connected to C bar (invert of c). When control signal C is high i.e. VDD, both transistor are on and provides a low resistance path between A and B. On the other hand, when C is low both are turned off and provide high impedance path between A and B.

When the control input is a logic zero (negative power supply potential), the gate of the NMOS is also at a negative supply voltage potential. The gate terminal of the PMOS is caused by the inverter, to the positive supply voltage potential. Regardless of on which switching terminal of the transmission gate (X or Y) a voltage is applied, the gate-source voltage of the NMOS is always negative, and the PMOS is always positive. Accordingly, neither of the two transistors will conduct and the transmission gate turns off. When the control input is a logic one, so the gate terminal of the NMOS is located at a positive supply voltage potential. By the inverter, the gate terminal of the PMOS is now at a negative supply voltage potential.

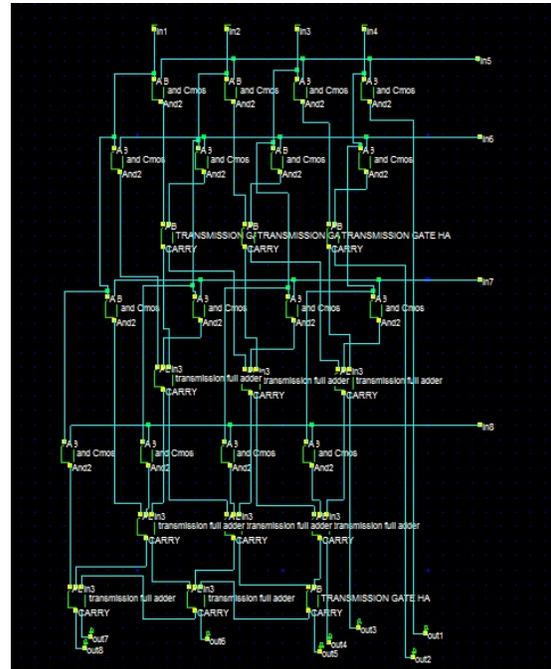


Figure 5:- 4×4 Array multiplier using transmission gate.

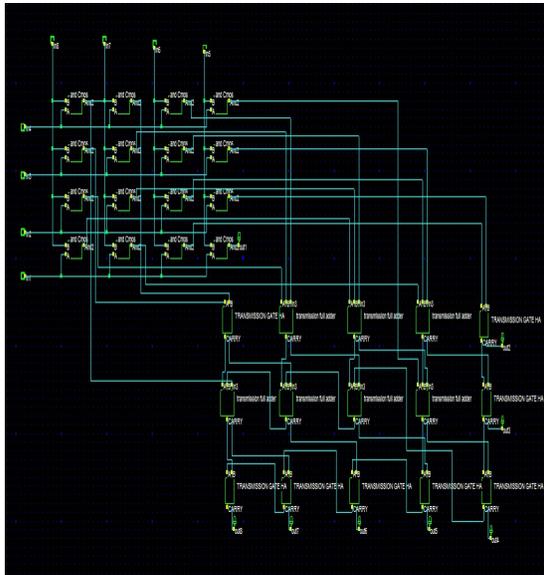


Figure 6:- 4x4Wallace tree multiplier using transmission gate.

C. 6T XOR GATE LOGIC

Adders plays vital role in scientific computations. Even sub systems like multipliers can be designed with the help of adders. So, power reduction with adder will make overall power change in a chip. A study of adder design by efficient XOR gates is made. XOR is designed with few numbers of transistors in turn help to reduce the size and power the adder circuitry.

A main objective of this work is to design low-power adder circuits with the help of efficient XOR gates. This circuit focuses more on reducing the layout area of the adder design which in turn minimize the power.

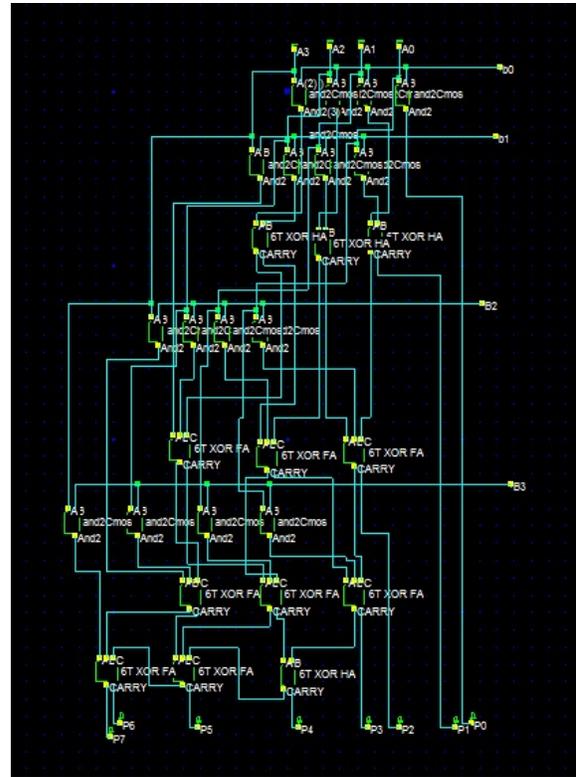


Figure 7:- 4x4Array multiplier using 6T XOR gate.

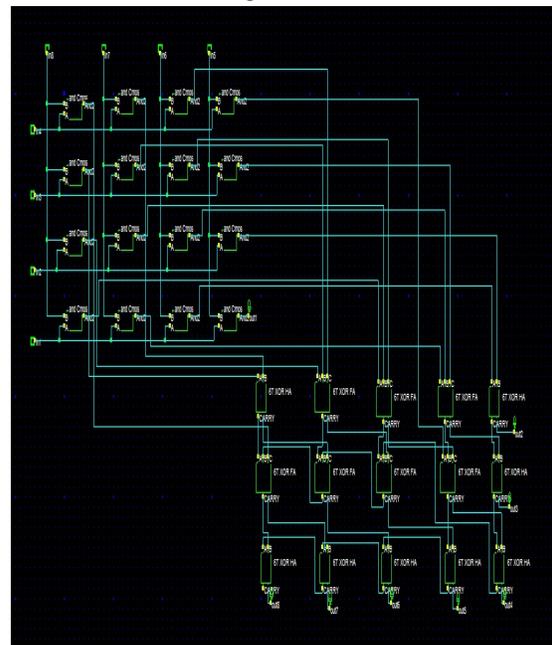


Figure 8:-4x4Wallace tree multiplier using 6T XOR gate.

RESULTS

A. DELAY IN STATIC LOGIC

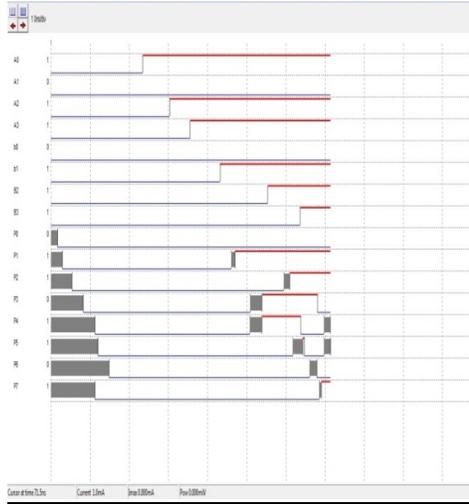


Figure 9:- 4x4 array multiplier delay with input (1101*1110).

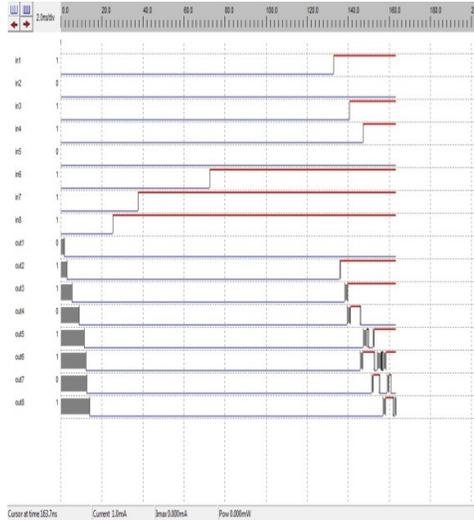


Figure 10:- 4x4Wallace tree multiplier delay with input (1101*1110).

B. DELAY IN TRANSMISSION LOGIC GATE

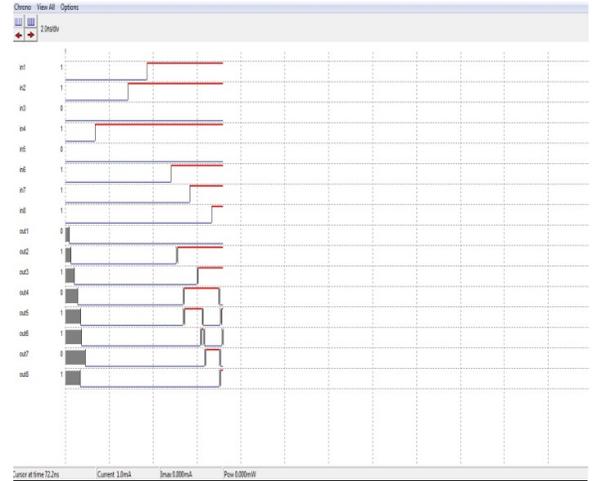


Figure 11:- 4x4 array multiplier delay with input (1101*1110).

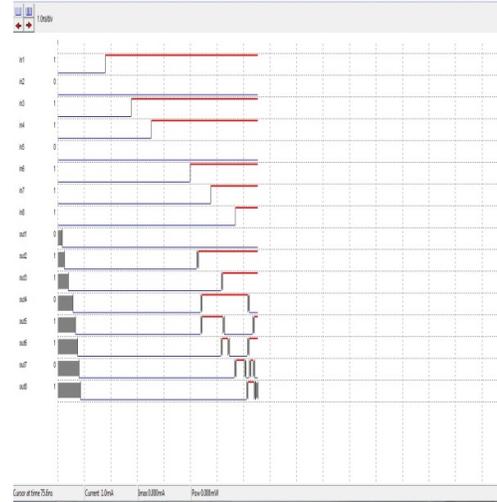


Figure 12:- 4x4Wallace tree multiplier delay with input (1101*1110).

C. DELAY IN 6T XOR LOGIC

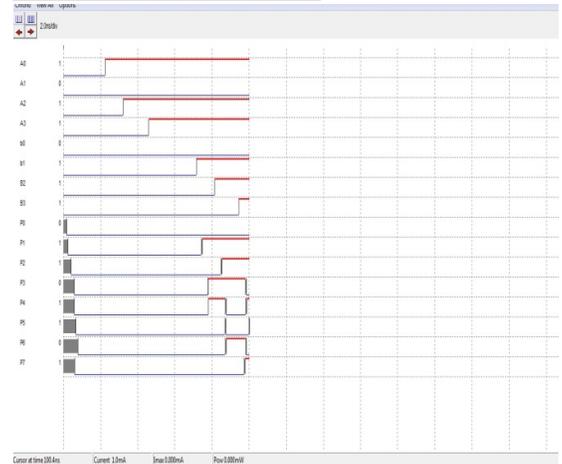


Figure 13:- 4x4 array multiplier delay with input (1101*1110).

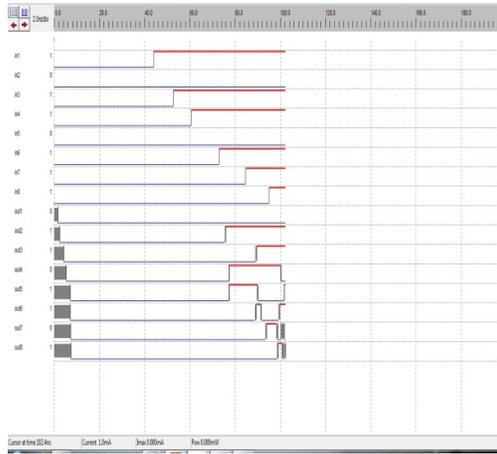


Figure 14:- 4×4Wallace tree multiplier delay with input (1101*1110).

D. COMPARISON TABLE FOR NO. OF TRANSISTORS USED

MULTIPLIER S	STATI C LOGIC	TRANSMISSIO N LOGIC	6T XOR LOGI C
ARRAY	504	384	336
WALLACE TREE	534	402	360

E. COMPARISON TABLE FOR POWER CONSUMPTION

MULTIPLIERS	STATIC LOGIC	TRANSMISSI- ON LOGIC	6T XOR LOGIC
ARRAY	0.705	0.433	0.313
WALLACE TREE	0.762	0.465	0.347

F. COMPARISON TABLE FOR DELAY PRODUCED

MULTIPLE S	STATI C LOGIC	TRANSMISSIO N LOGIC	6T XOR LOGI C
ARRAY	0.7	0.6	0.6
WALLACE TREE	0.6	0.5	0.5

V. CONCLUSION

The study of various circuits like half adder, full adder, 4*4 array multiplier and 4*4 Wallace tree multipliers using the logics like static logic, Transmission gate logic and 6T XOR gate logic has led to the comparison of the circuits followed by the conclusion of which one being the best with respect to area, delay and power dissipation.

Thus, we conclude that 4*4 array multiplier using the 6T XOR logic require least area of 336 transistors respectively and dissipating least power of 0.313mA respectively proving themselves to be the most power and area efficient as compared to its other counterparts and whereas 4*4 Wallace tree multiplier provide least delay of 0.5ns in transmission as well as 6T XOR logic hence which increases the speed of multiplier compared to others.

REFERENCES

1. John P. Uyemura, "Introduction to VLSI Circuits and Systems".
2. Soniya and Suresh Kumar ,“A Review of different type of multipliers and multiplier-Accumulator unit ”, (IJETTCS) vol.2,Issue 4, July-August 2013.
3. Ron S.Waters and Earl E.Swartzlander, “A Reduced Complexity Wallace Multiplier Reduction”,0018-9340/10/\$26.00 © 2010 IEEE.
4. Gary W.Beick, “Fast Multiplication: Algorithms and implementation”, Ph.D.Thesis STANFORD UNIVERSITY, February 1994.

5. B.Dinesh, V.Venkateshwaran, P.Kavinmalar and Dr.M.Kathirvelu, “Comparison of Regular and Tree based Multiplier Architectures with Modified Booth Encoding for 4 Bits on Layout Level using 45nm technology”, IEEE 2014.
6. Partha Sarathi Mohanty, “DESIGN AND IMPLEMENTATION OF FASTER AND LOW POWER MULTIPLIERS”, Department Of Electronics and Communication Engineering National Institute Of Technology Rourkela 2009.
7. Weste.N and Eshraghian.K, Principles of CMOS VLSI Design, A System Perspective. Reading, MA: Addison- Wesley, 1988.
8. Wallace.S "A Suggestion for a Fast Multiplier". IEEE Transactions on Electronic Computers, EC-13:14–17, February 1964.
9. Ila Gupta, Neha Arora and Prof. B. P. Singh “2x2 Array Multiplier Based on DCVS Logic”, Apr. 2012, Vol. 2(4) pp: 923-927.
10. Sumit Vaidya and Deepak Dandkar, “Delay-Power Performance Comparison of Multipliers in VLSI Circuit Design”, (IJCNC), VOL.2,No.4, July 2010.