

Performance Analysis of 4-Bit Multiplier using 90nm Technology

Anurag Chauhan
Department of ECE,
Delhi Technological University,
Delhi, India
anuragchauhan@dtu.ac.in

Amit Kumar Meena
Department of ECE,
Delhi Technological University,
Delhi, India
amitkumarmeena_2k18ec028@dtu.ac.in

Amit Kumar
Department of ECE,
Delhi Technological University,
Delhi, India
amitkumar_2k18ec026@dtu.ac.in

Amit Kumar
Department of ECE,
Delhi Technological University,
Delhi, India
amitkumar_2k18ec027@dtu.ac.in

Abstract— This paper explores three different 4-bit multipliers built using a modified full adder at the 90nm technology and compares low-power, high-speed multiplier designs with their CMOS counterparts. Since the multiplier block consumes a lot of power and plays a big part in the circuit's speed, the proposed multiplier will help in optimizing and enhancing the circuit performance. Our analysis results suggest that the proposed multipliers offer a decrease in power up to ~47.6%, a decrease in delay up to ~63.96%, and a decrease in transistor count up to ~58.7% when compared with the CMOS based designs.

Keywords— Multiplier, Low Power Application, Delay, Power Dissipation, Area, Pass Transistor Logic (PTL), Adder

I. INTRODUCTION

With the advancement of integration scale, the density of functionalities needed to be implemented on a chip is also increasing many folds. These functionalities use a lot of energy and computing power. Power dissipation along with speed and area has become an issue for integrated circuit designers [1]. Low-power systems were developed as a result of two key considerations. Firstly, an increase in integration has resulted in an increased capacity in processing. It leads to high current flow and therefore overheating of ICs. Secondly, the low power design allows for longer operation of these devices even with the finite battery capacity [2].

Multiplication has its great significance in the majority of signal processing methods. The multiplier is usually the slowest part of the circuit, hence its performance directly determines the system's total performance [3]. The multiplier's speed and the area is a crucial design considerations. Since all multipliers employ full adders, the multiplier's can be improved by using more efficient adders. However, area requirement and the speed of the circuit are mostly inversely proportional to each other therefore it would be interesting to design various multipliers with optimized speed-area constraints. In this paper, we report a PTL based full adder. The most commonly used multipliers array, Wallace tree, and DADDA multiplier have also been designed and analyzed based on the PTL based full adder. The simulations to analyze power and delay have been carried out at 90nm technology using the cadence virtuoso toolkit.

II. RELATED WORK

A. Array Multiplier

Array multiplier is basically an effective design of a combinational multiplier as shown in Fig. 1. All the product bits can be formed at once by multiplying two binary numbers for one micro-operation and this makes it a quick way of multiplying two numbers because the only delay for the signals is the period that radiates through the gates which form the multiplication array [4]. Multiplication is done only on the number's logic 0 and logic 1, i.e., binary numbers.

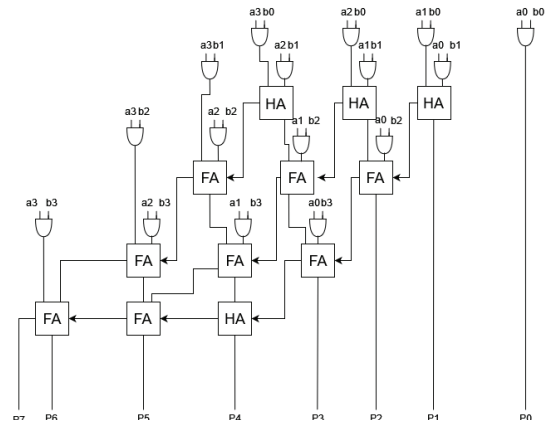


Fig. 1. 4x4 Array Multiplier

The multiplication is carried out using the following basic multiplication rules: $0 \times 1 = 0$, $0 \times 0 = 0$, $1 \times 1 = 1$ and $0 \times 1 = 0$. If we multiply two 4 bit numbers, 0100 and 0110, then the result of that multiplication is an 8-bit value i.e., 00011000.

For n bit multiplier, $n(n-2)$ FA, n HA, and n^2 AND gates are required [5].

B. Wallace tree Multiplier

C.S. Wallace proposed a method for performing multiplication quickly in 1964. The amount of computation required for this type of multiplication is considerable, but the delay is very less. here the delay is proportional to $\log(N)$, where N is the word length. This methodology is generally applied when the speed requirement is a priority over the area. Fig. 2 depicts a design of the Wallace tree multiplier. The

multiplication operation is processed in three phases by the Wallace technique [6]. They are:

- Bit product formation
- Using a standard adder, combine all product matrices to generate two vectors (sum and carry) results in the first row.
- To create a product, quick carry propagate adder is used and then by using it the remaining two rows are added.

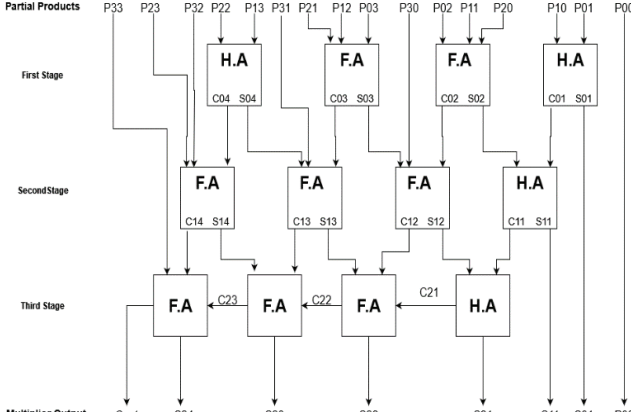


Fig. 2. 4x4 Wallace Tree Multiplier

Wallace tree structure is a widely utilized multiplier design in various memory units and processors. There are two steps in the Wallace tree multiplication process. The input numbers are given to the AND gate in phase 1 to form the partial products and in a step-by-step process, these partial products are divided into three rows to obtain the final product output every three rows added together simultaneously in phase 2 by using HA and FA [7] as shown in Fig. 3.

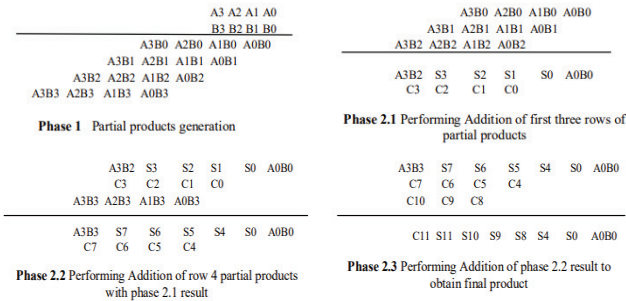


Fig. 3. multiplication of various phases in a Wallace tree multiplier

The first step includes creating partial products of the input numbers by multiplying each bit of it with each other. The length of the inputs is 4-bits, hence four rows of partial products are formed. The addition of the partial products acquired in phase 2 is broken down into multiple sub-phases. Half and full adder is used to perform the addition function. Initially, in phase 2, summation operation is conducted on the first three rows of partial products obtained in phase 1, resulting in two rows of results, one with sum terms and the other with carry terms. Then, the partial product's final row from phase 1 is then added to the carry and sum row, giving two rows with one carry row and one sum row each. The carry row and sum row are added to get the final product.

C. DADDA Multiplier using Ripple Carry Adder

In 1965, Luigi Dadda, a computer scientist, created the DADDA multiplier. The DADDA multiplier is one of the scheme derived out of parallel multiplier [8]. The DADDA method is a parallel multiplier strategy that decreases the number of adder stages required to complete partial product summing. This is accomplished by using HA and FA in a matrix to decrease the number of bits and number of rows at every summing stage. Therefore, the DADDA multiplier is less expensive than the Wallace tree multiplier as it requires lesser no of gates. Despite having a less complex and a simpler structure, the DADDA multiplication method is slow due to the serial multiplication process. DADDA multiplier can be designed by using a ripple carry full adder (RCA).

RCA is a method for chaining a number of additions that must be done with the C_{out} and C_{in} . As a result, the ripple carry adder uses many adders. For multiple-bit numbers addition, a logical circuit can be built using several full adders [9]. C_{out} of the previous FA becomes an input of the next full adder as C_{in} . Because each carry bit "ripples" to the next FA, this type of adder is called a ripple carry adder.

Fig. 4 depicts the design of the DADDA multiplier with RCA.

The steps involved in creating a 4x4 DADDA multiplier with RCA are described below.

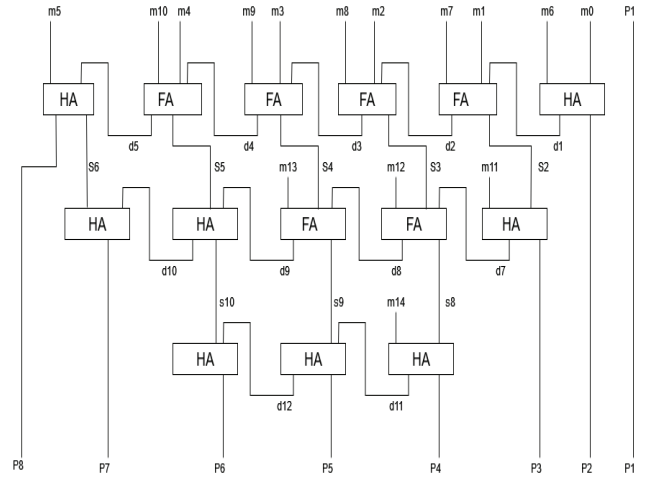


Fig. 4. 4x4 DADDA Multiplier with RCA

Feed any three wires having the same weights into a FA as input. As a result, the output wire would be the same weight [10].

- At the first stage, the partial products is obtained after multiplication. Then, the data is collected using three wires and added with adders, after that each stage's carry being added in the same stage with the next two data.
- With the same procedure, partial products are reduced to two layers of full adders.
- The same ripple carry adder approach is used in the last stage, and product terms P1 to P8 are generated.

III. PROPOSED WORK

In this section, we propose multiplier structures that outperforms existing CMOS based multiplier structures. The simulations are carried out at 90nm technology using cadence virtuoso toolkit. To perform the addition of intermediate bits in a multiplier, the proposed FA circuit using pass transistor logic (PTL) and a HA circuit is used. The proposed PTL based FA designed by using a 10 transistor is shown in Fig. 5. The usage of pass transistor logic has resulted in a significant lower number of transistors as a result, the area has also reduced.

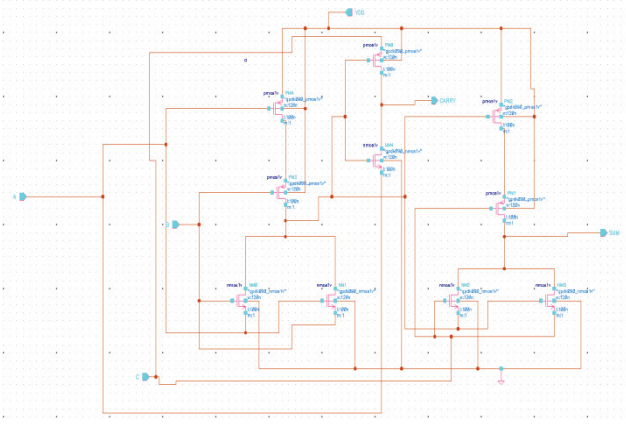


Fig. 5. Proposed Full Adder

The supply voltage is 0.9V for our proposed FA.

The sum's expression is given by (1)

$$\begin{aligned}
 Sum &= I_1 \oplus I_2 \oplus I_3 \\
 &= (I_1 \oplus I_2) \oplus I_3 \\
 &= (\bar{I}_1 I_2 + I_1 \bar{I}_2) \oplus I_3 \\
 &= (\bar{I}_1 I_2 + I_1 \bar{I}_2) I_3 + (\bar{I}_1 I_2 + I_1 \bar{I}_2) \bar{I}_3 \\
 &= (\bar{I}_1 I_2 + I_1 \bar{I}_2) I_3 + (\bar{I}_1 I_2 + I_1 \bar{I}_2) \bar{I}_3 \\
 &= \bar{I}_1 I_2 I_3 + I_1 I_2 \bar{I}_3 + \bar{I}_1 I_2 \bar{I}_3 + I_1 \bar{I}_2 I_3
 \end{aligned} \quad (1)$$

Similarly, The carry's expression is given by (2)

$$\begin{aligned}
 Carry &= I_1 I_2 + I_2 I_3 + I_3 I_1 \\
 &= I_1 I_2 + I_3 (I_2 + I_1) \\
 &= I_1 I_2 + I_3 (I_1 + I_2) (I_1 + \bar{I}_1) \\
 &= I_1 I_2 + I_3 (I_1 + \bar{I}_1 I_2) \\
 &= I_1 I_2 + I_1 I_3 + \bar{I}_1 I_2 I_3 \\
 &= I_1 (I_2 + I_3) + \bar{I}_1 I_2 I_3 \\
 &= I_1 (I_2 + I_3) (I_2 + \bar{I}_2) + \bar{I}_1 I_2 I_3 \\
 &= I_1 (I_2 + I_3 \bar{I}_2) + \bar{I}_1 I_2 I_3 \\
 &= I_1 I_2 + I_1 I_3 \bar{I}_2 + \bar{I}_1 I_2 I_3 \\
 &= I_1 I_2 + (I_1 \bar{I}_2 + \bar{I}_1 I_2) I_3
 \end{aligned} \quad (2)$$

In our proposed full adder, we are getting (1) at the output of second NOR gate and (2) at the output of NOT gate.

By using a proposed full adder, we have designed a proposed array multiplier, DADDA multiplier and Wallace tree multiplier. Fig. 6, Fig. 7 and Fig. 8 depicts a schematic of the proposed multipliers.

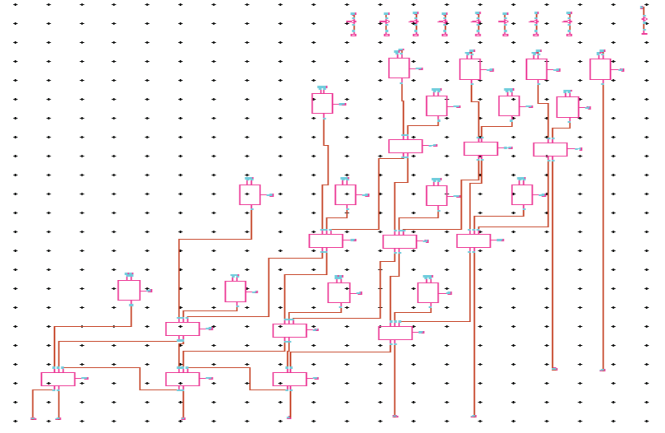


Fig. 6. 4x4 Array Multiplier

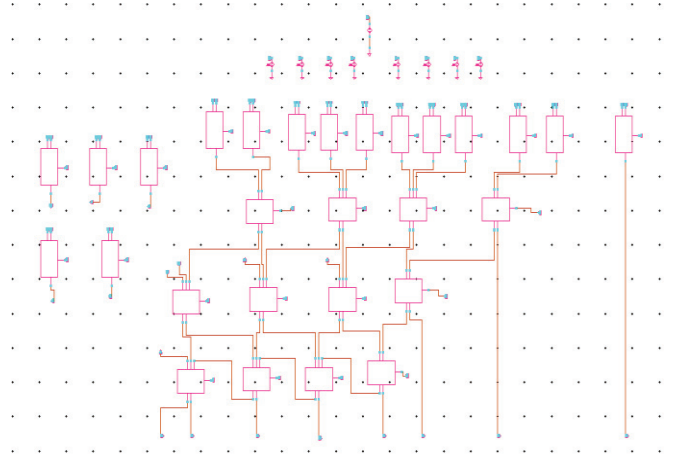


Fig. 7. 4x4 Wallace Tree Multiplier

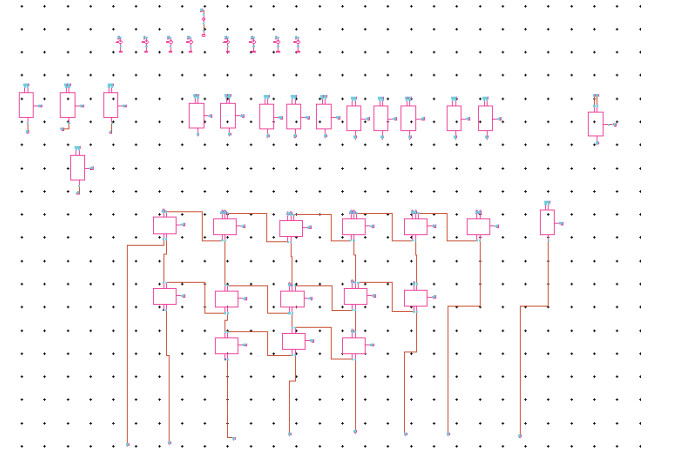


Fig. 8. 4x4 DADDA Multiplier

IV. RESULTS AND ANALYSIS

The summary of the results can be seen from the Table 1. The proposed multipliers were compared to the CMOS version of multipliers. The output for the proposed multipliers is shown in Fig. 9, Fig. 10 and Fig. 11.

Total power consumption, delay, and area are calculated using simulations of our proposed multipliers.

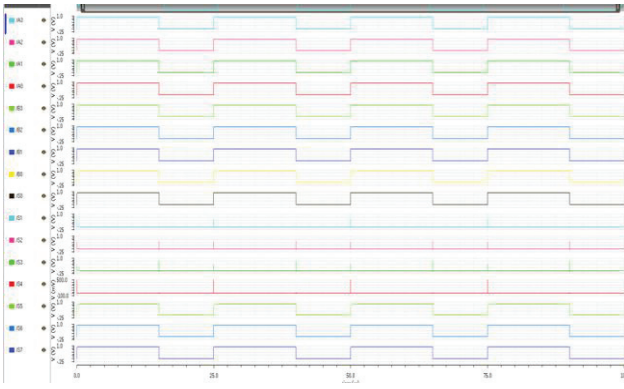


Fig. 9. Output of Proposed 4x4 Array Multiplier

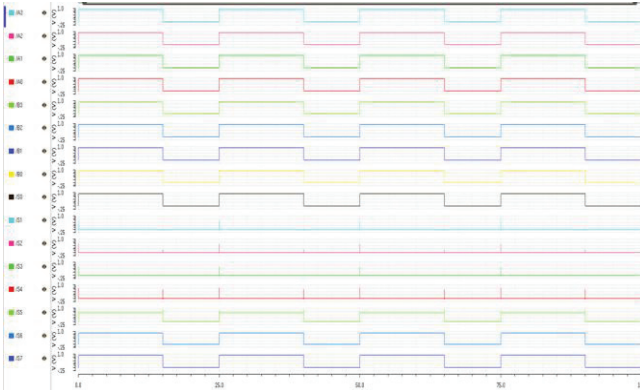


Fig. 10. Output of Proposed 4x4 Wallace Tree Multiplier

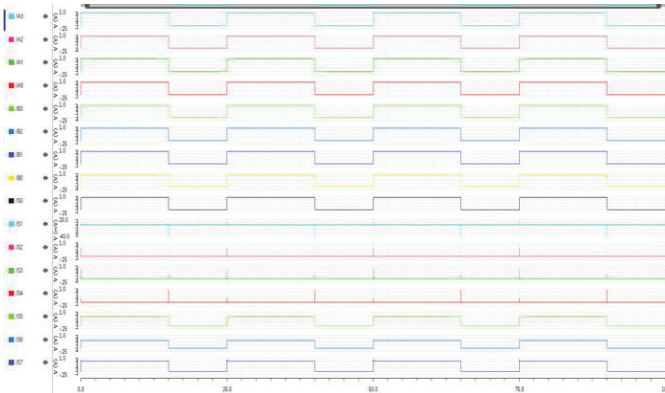


Fig. 11. Output of Proposed 4x4 DADDA Multiplier

The comparison of area, delay and power consumption of the proposed multipliers is shown in Fig. 12, Fig. 13 and Fig. 14.

Our proposed Array multiplier offers a decrease of 47.6% in power, a decrease of 63.63% in delay and a decrease of 58.7% in the transistors count i.e., area, compared to existing CMOS Array multiplier. Similarly, our proposed Wallace tree multiplier offers a decrease of 37.8% in power, a decrease of

63.6% in delay and a decrease of 58.74% in the transistors count i.e., area, when compared to existing CMOS

Wallace Tree multiplier, this is a significant improvement. Similarly, our proposed DADDA multiplier offers a decrease of 22.00% in power, a decrease of 63.96% in delay and a decrease of 55.29% in the transistors count i.e., area, in comparison to existing CMOS DADDA multiplier.

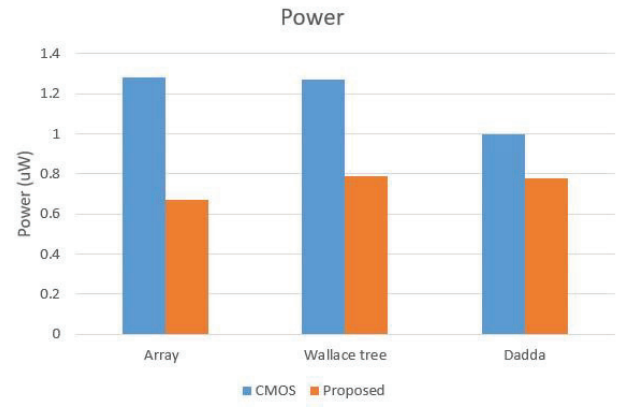


Fig. 12. Comparison of Power Consumption for Multipliers

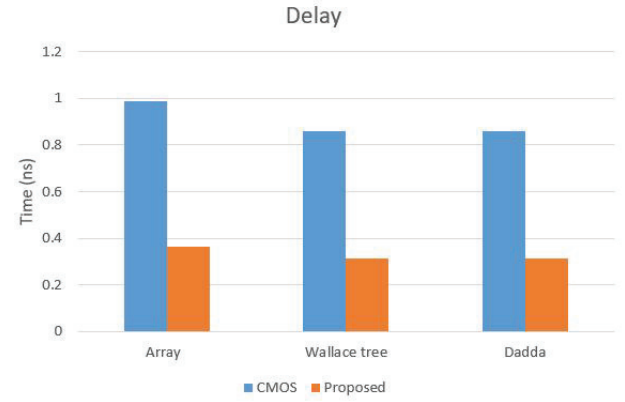


Fig. 13. Comparison of Delay for Multipliers

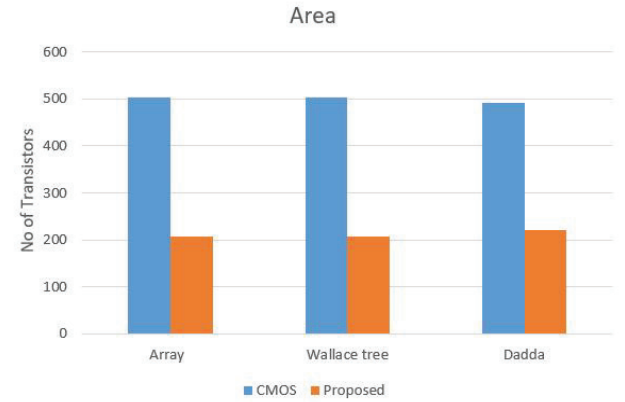


Fig. 14. Comparison of Area for Multipliers

TABLE I. SUMMARY OF DELAY, AREA AND POWER OF VARIOUS MULTIPLIER

Multiplier	Design	Area (no. of T)	Delay(ns)	Power(uW)
Array Multiplier	CMOS	504	0.99	1.28
	Proposed	208	0.36	0.67
Wallace Tree Multiplier	CMOS	504	0.86	1.27
	Proposed	208	0.31	0.79
DADDA Multiplier	CMOS	492	0.86	1.00
	Proposed	220	0.31	0.78

V. CONCLUSION

In this paper, we successfully investigated three different 4-bit multipliers at 90nm technology. Our analysis shows that our proposed array multiplier has a least amount of power consumption, the delay is lower in the DADDA and Wallace tree multiplier and the number of transistors used is lesser in the Wallace tree and array multiplier. The proposed structures will be advantageous in applications that employ multipliers as a key element in the circuit. It can be utilized in a variety of applications, including ALUs and DSPs.

REFERENCES

- [1] P. Mittal, Y. S. Negi and R. K. Singh, "Impact of Source and Drain Contact Thickness on Performance of Organic Thin-Film Transistors", *Journal of Semiconductors* (Published by IOP Sciences, SJR/SCImago, Scopus Indexed), Vol. 35, No. 12, pp. 124002-1–124002-7, Dec. 2014.
- [2] M. Janveja, G. L. Bajaj Institute of Technology and Management, India, V. Niranjana, and Indira Gandhi Delhi Technical University for Women, India, "High performance Wallace tree multiplier using improved adder," *ICTACT j. microelectron.*, vol. 3, no. 1, pp. 370–374, Apr 2017.
- [3] V. Aruna and P. Deepthi, "High Performance Low power Dynamic Multiplier," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN (2013): 2278-3075.
- [4] K. Yugandhar, V. G. Raja, M. Tejkumar and D. Siva, "High Performance Array Multiplier using Reversible Logic Structure," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), 2018, pp. 1-5.
- [5] S. Srikanth, I. T. Banu, G. V. Priya and G. Usha, "Low power array multiplier using modified full adder," 2016 IEEE International Conference on Engineering and Technology (ICETECH), 2016, pp. 1041-1044.
- [6] C. S. Wallace, "A Suggestion for a Fast Multiplier," in *IEEE Transactions on Electronic Computers*, vol. EC-13, no. 1, pp. 14-17, Feb. 1964.
- [7] Y. d. Ykuntam, K. Pavani and K. Saladi, "Design and analysis of High speed wallace tree multiplier using parallel prefix adders for VLSI circuit designs," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-6.
- [8] K. C. Bickerstaff, M. Schulte and E. E. Swartzlander Jr., "Parallel reduced area multipliers," *Journal of VLSI Signal Processing Systems*, vol. 9, no. 3, pp. 181–191, Apr. 1995.
- [9] R. Uma, Vidya Vijayan, M. Mohanapriya, Sharon Paul, "Area, Delay and Power Comparison of Adder Topologies," *International Journal of VLSI design & Communication Systems*, vol.3, No.1, pp.153-168, Feb. 2012.
- [10] Samundiswary. P. and K. Anitha, "Design and Analysis of CMOS Based DADDA Multiplier," *IJCEM International Journal of Computational Engineering & Management*, Vol. 16 Issue 6, Nov. 2013.