Comparison of Regular and Tree based Multiplier Architectures with Modified Booth Encoding for 4 bits on Layout Level using 45nm technology

B.Dinesh[1],V.Venkateshwaran[2],P.Kavinmalar[3],Dr.M.Kathirvelu[4]
Pg. student[1,2,3], HOD/ECE[4],Department of ECE
Department of ECE, KPR Institute of Engineering and Technology
Coimbatore-641407, Tamil Nadu, India
dineshkprit@gmail.com

***Abstract*—Multipliers are key components of many high performance systems such as FIR filters, microprocessors, digital signal processors, etc. A system's performance is generally determined by the performance of the multiplier as the multiplier is generally the slowest element in the system. The analysis of performance parameters of different multiplier logics is essential for design of a system intended for a specific function with constraints on Power, Area and Delay. The paper presents a detailed analysis of all the serial-parallel and parallel architectures. The multipliers are designed for 4 bit multiplication using DSCH tool and the corresponding layouts are obtained using Microwind 3.5 tool using 45nm technology. From the analysis it is observed that the array multipliers provide a regular routing structure which will be optimum for FPGA based systems. Among the tree based multipliers Dadda multipliers have a slight advantage over Wallace tree multipliers in terms of performance. The Modified booth multiplier is comparatively inefficient for bits lesser than or equal to 4, due to the increased area involved for realization of the booth encoder and booth selector blocks. The analysis shows that for lower order bits Dadda reduction is the most efficient.**

***Keywords*—PDP, Low Power, Multipliers, Layout.**

I. INTRODUCTION

Multiplication can be considered as a series of repeated additions. The number to be added is called the multiplicand, the number of times it is added is called multiplier and the result obtained is called the product. The multiplier architecture can be generally classified into following categories. Serial - includes shift and add multiplier based on sequential logic. Serial-parallel- Array multipliers in which serial CPA logic is used along with parallel row based reduction. Parallel- Tree based multipliers where the partial products are reduced in a parallel manner. Serial multipliers are mainly aimed at reducing the area involved hence they are relatively slow since there is no parallelism involved. The array multiplier belongs to the class of serial-parallel multipliers. The tree based multiplication involves parallel CSA based technique for reduction of partial products. The Modified booth algorithm is used for reducing the number of partial product rows to (N+2)/2 by means of Booth encoding. The reduced partial product rows are added by any of Array, Wallace or Dadda reduction methodologies

II. ARRAY MULTIPLIER

Array multiplier [7] is well known due to its regular structure. Multiplier circuit is based on add and shift algorithm. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial products are shifted according to their bit orders and then added. The addition can be performed with normal carry propagate adder. N-1 adders are required where N is the multiplier length. The delay of an array multiplier is shown in the Equation. 1.

$$\text{Delay} = 2*N*\text{ Adder delay} \quad (1)$$

The regular 4 × 4 array multiplier is synthesized using the DSCH tool and is shown in Fig. 1 and the corresponding layouts are obtained using MICROWIND 3.5 using 45nm technology.

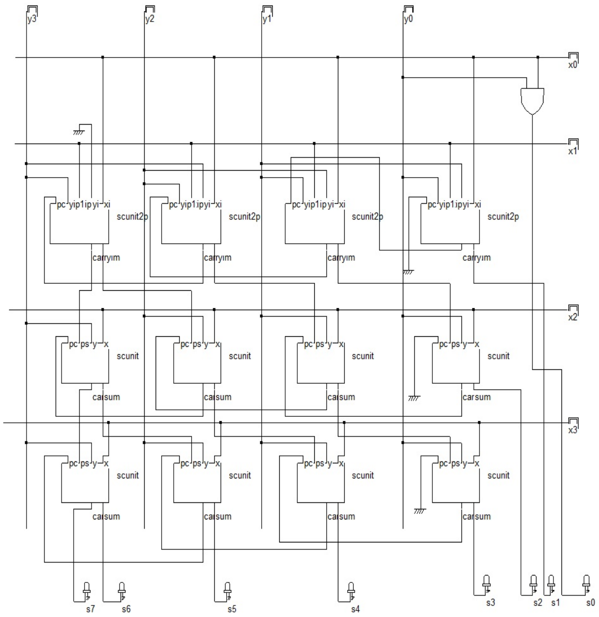

Fig.1.Schematic of the array multiplier using DSCH 3.5 tool

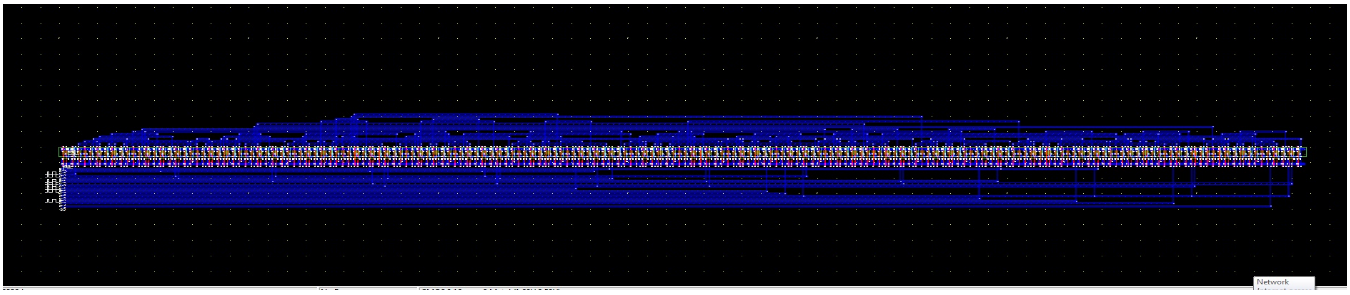

Fig.2. Layout of an array multiplier using 45nm technology

III. Wallace Tree Multiplier

Tree based multiplier employ parallel multiplication logic based on the concept of CSA.The tree based architectures are mainly designed in the aim of reducing the delay along long chain of adders. The delay in an array multiplier is relative high due to reason that it employs carry propagate addition. Carry propagate additions are relatively slow. The delay in array multiplier is greatly governed by the bitwise length of the multiplicand and multiplier inputs of an array multiplier. The Wallace tree multipliers are based on parallel reduction using the concept of carry save addition [2]. The delay of a Wallace tree multiplier is given by the Equation.2. The "N" in the Equation.2. Denotes the number of bits of the multiplier or the multiplicand. The delay is reduced considerably compared to the array multiplier. The basic dot diagram describing the working of a Wallace tree multiplier is shown in Fig. 3. From the dot diagram shown in the Fig.3. It can be inferred that the partial products can be added and reduced to 2 rows without carry propagate addition. A single carry propagate addition is only needed in the final step to reduce the 2 numbers to a single product. The synthesized Wallace tree multiplier using DSCH 3.5 tool is depicted in the Fig. 5.The layout level architecture of the Wallace tree multiplier for 4bit multiplication is shown in the Fig. 6. The Wallace tree multiplier suffers from irregular routing. The routing complexity of Wallace tree multiplier is high when implemented on a FPGA.The routing complexity can be reduced in a Wallace tree multiplier by reducing the number of half adders in a Wallace tree multiplier structure. The architecture with reduced complexity is given in [6].

$$\text{Delay} = \text{Log}\,(N)^{*}\ \text{Adder delay} \qquad (2)$$

The architecture discussed in [6] will be beneficial for FPGA implementation of the Wallace tree multiplier. The Wallace tree multiplier reduction is used in Booth multipliers. Due to the reduced delay involved.

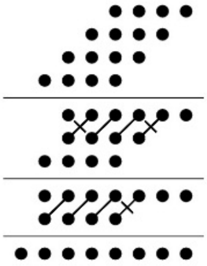

Fig. 3. Dot diagram for a Wallace tree multiplier

IV. Dadda Multiplier

The use of half adders in early stages to distribute the partial products among the rows horizontally is done to ease the use of full adders in the latter stages in a Dadda multiplier. The concept involving the reduction of columns instead of rows this is given in detail in [3]. The goal of the Dadda algorithm is to minimize the number of logical elements used. The algorithm is based on the reduction of the maximum height columns by the use of half adders to achieve a partial product arrangement as shown in the dot diagram in Fig. 4. The synthesized architecture of the Dadda multiplier using DSCH 3.5 is shown in the Fig. 7. The layout level analysis of the Dadda multiplier is done with all combinations of input patterns and the PDP calculated is given in Table.2. The layout of the Dadda multiplier is given in Fig. 8

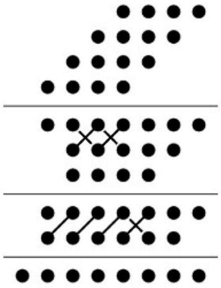

Fig. 4. Dot diagram for a Dadda multiplier

$$\text{Delay} = \text{Log}\,(N)^{*}\ \text{Adder delay} \qquad (3)$$

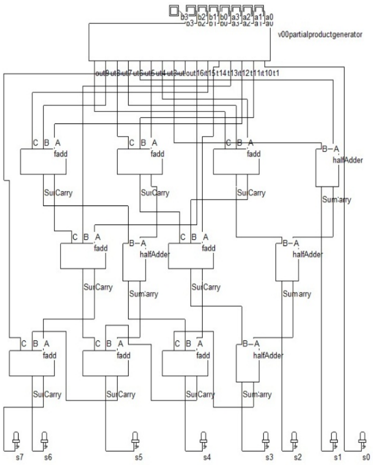

Fig. 5. Schematic of a Wallace tree for 4 × 4 synthesized using DSCH tool

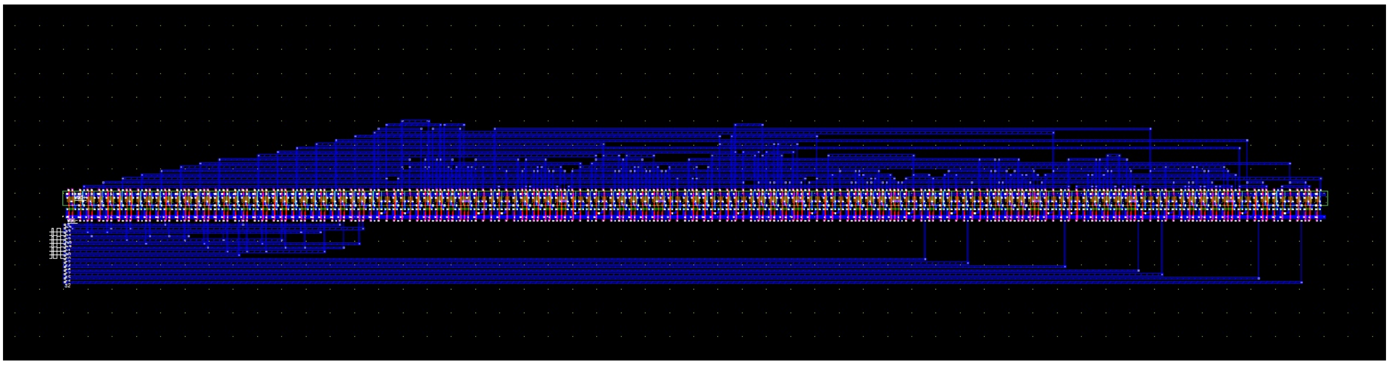

Fig.6. Layout for a Wallace tree multiplier using 45nm technology

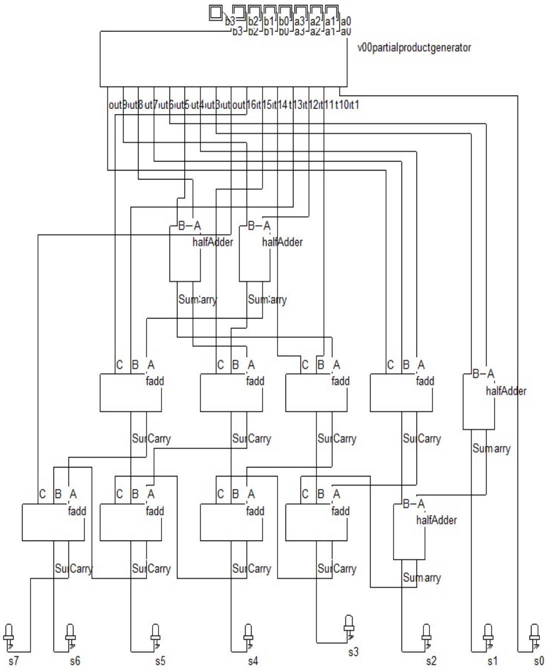

Fig.7. Schematic of a Dadda Multiplier for 4 × 4 synthesized using DSCH 3.5 tool

V. MODIFIED BOOTH MULTIPLIER

Modified Booth Multiplier based on the Modified Booth Algorithm [4] is based on the concept of encoding by means of Booth encoder to reduce the number of the partial product rows to (N+2)/2 where N is the number of bits of the Multiplier or the Multiplicand. The Booth Encoding combined with the selector implements the transformations shown in the Equations.3, 4, 5, 6. The resulting rows of the partial products thus obtained can be reduced by means of CPA or tree based reduction schemes. The Booth Multiplier offers a block based architecture that offers a regular routing for FPGA implementation. The Booth Multiplier is synthesized for a 4 bit Multiplication using DSCH 3.5 is shown in the Fig. 11. The layout transformation is done using Microwind 3.5 Tool, The layout of a 4 bit booth multiplier is shown in Fig.10. The Corresponding output waveforms are obtained by the simulation by means of using all combinations of input patterns and common settings. Booth encoding for 3 bits is given in Table. 1 and the functioning of the selector is based on the encoding process and is given in Eqn.3 to Eqn.5.

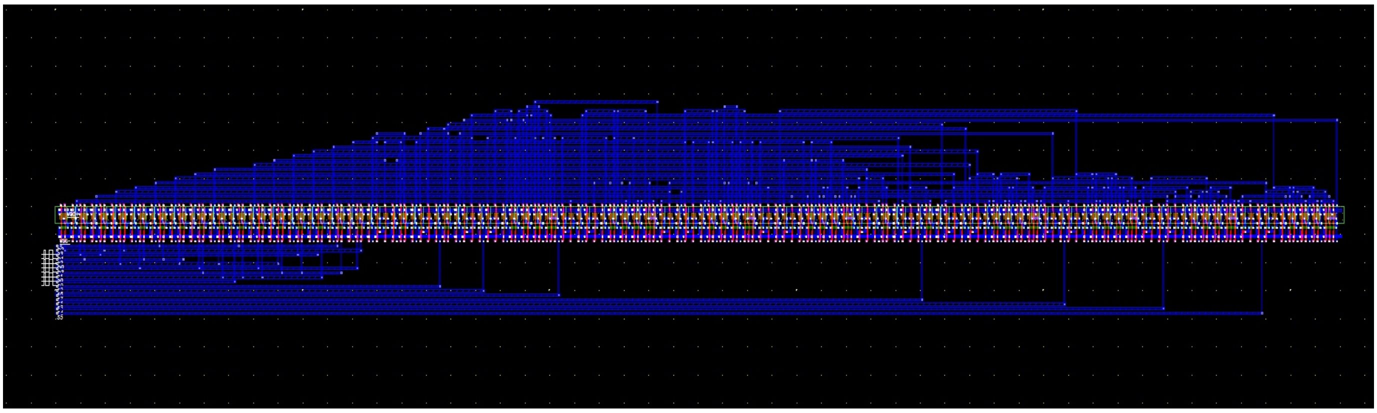

Fig.8. Layout of a Dadda multiplier using 45nm technology

Table I .Radix 4 Booth encoding

X_{i+1} X_i X_{i-1}	PPi	X_i $2X_i$ M_i
0 0 0	0	0 0 0
0 0 1	Y	1 0 0
0 1 0	Y	1 0 0
0 1 1	2Y	0 1 0
1 0 0	-2Y	0 1 1
1 0 1	-Y	1 0 1
1 1 0	-Y	1 0 1
1 1 1	-0(=0)	0 0 1

$$Mi = x_1 \quad (3)$$

$$x = x_0 \oplus x_{-1} \quad (4)$$

$$x_2 = \overline{x_1} x_0 \, x_{-1} + \overline{x_{-1} x_0} \, x_1 \quad (5)$$

The 3 bit encoded values and corresponding partial product selection are given in the Table.1. Booth selection is based on the selection of the partial products based on the encoding scheme shown in Table.1 The Equation governing the partial product generation using the booth selectors is given in Equation. 6.

$$pp_{ij} = (Y_j X\overline{x_2} + Y_{j-1} X_2 \overline{x}) \oplus M_i \quad (6)$$

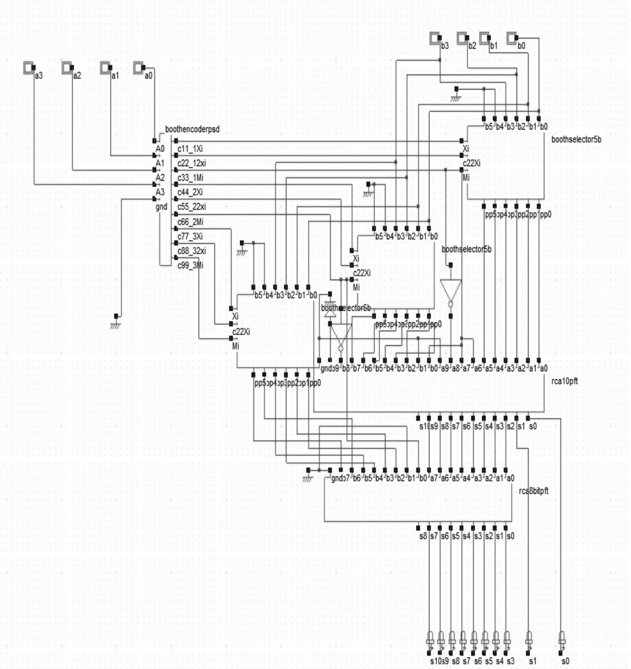

Fig. 9. Schematic of a Booth multiplier for 4 × 4 synthesized using DSCH 3.5 tool

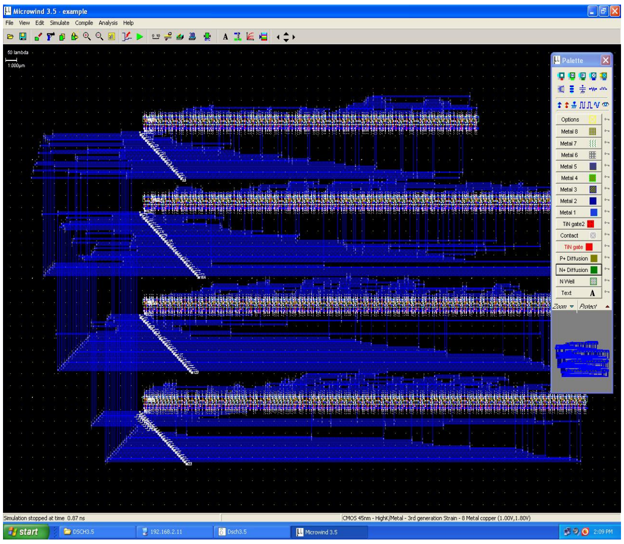

Fig. 10. Layout level of a Booth multiplier using 45nm technology

The various multipliers are synthesized and simulated using Microwind 3.5 tool using 45nm technology. Synthesis and analysis shows that the Dadda multipliers exhibit slightly reduced delay when compared to the Wallace tree multipliers, reason for the Dadda multiplier's delay reduction when compared to the Wallace tree is that the Half adders are used earlier in Dadda multipliers for distribution of the partial products among different levels in the partial product array is shown in Fig. 11. It can be also inferred that the booth multipliers exhibit lesser due to the partial product row reduction this is shown in Fig.11. But Booth multipliers are inefficient for multiplication of order lesser than 4. The Booth multiplication when applied for higher order multiplication is better than array and tree multipliers as it involves reduction of partial product rows by (N+2)/2. The various parameters are compared and shown in the graphical forms in the Fig. 11 to Fig. 14.

Table II .Comparison of various multiplier architectures

Multiplier	**Delay (pS)**	**Number of transistors**	**Power (mW)**	**PDP (pJ)**
Array multiplier	40	424	0.151	0.604
Wallace tree	31.4	268	0.080	0.251
Dadda	30.9	268	0.081	0.250
Booth multiplier	29.4	1026	0.213	0.628

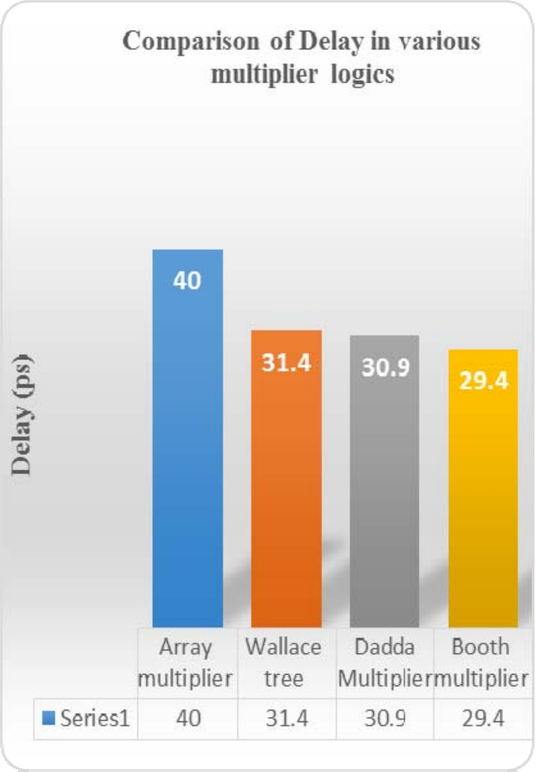

Fig. 11. Comparison of Delay among multipliers

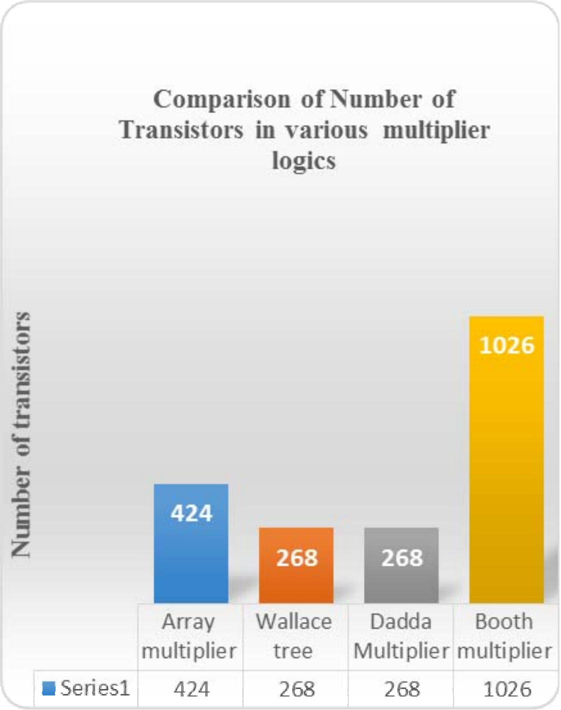

Fig. 12. Comparison of Number of Transistors Among Multipliers

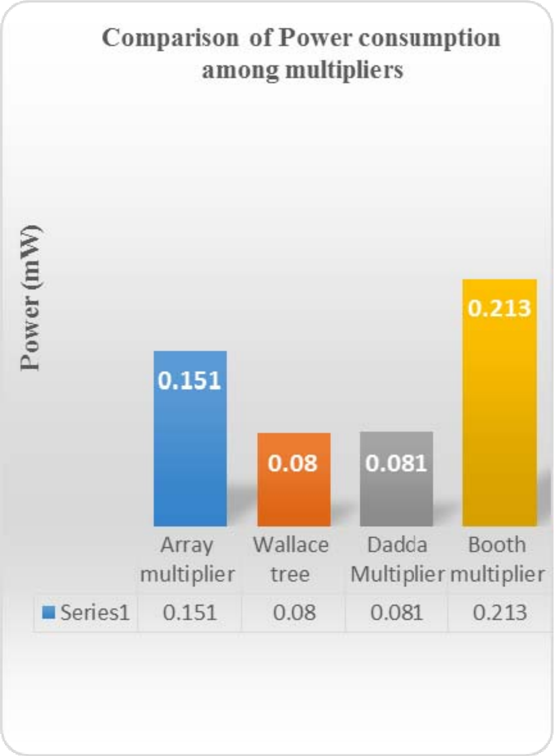

Fig. 13. Comparison of power Consumption Among Multipliers

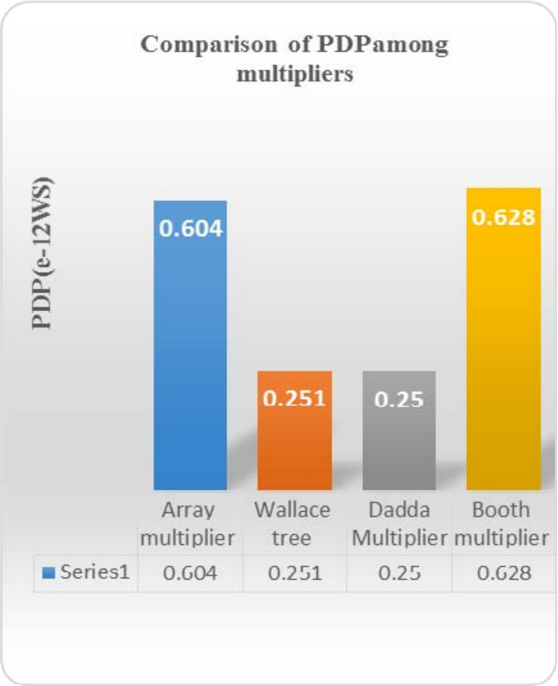

Fig.14. Comparison of PDP among multiplier

VI. CONCLUSION

The layout level Analysis of 4 bit multiplier logics using 45nm technology shows that the Tree based multiplier logics are the most efficient when fewer number of bits of data are considered. Of the tree based reduction methodologies the Dadda reduction technique is more efficient than the Wallace tree reduction due to the efficient use of half adders. The Booth reduction is inefficient when fewer bits are considered. But the reduction technique combined with tree based reduction will be highly efficient when higher order are considered. From the results it is evident that for lower order multiplier design the Dadda based reduction will benefit Reduction in Power, Delay and PDP.

VII. REFERENCES

[1] Wallace.S "A Suggestion for a Fast Multiplier". IEEE Transactions on Electronic Computers, EC-13:14–17, February 1964.

[2] Dadda .L. "Some Schemes for Parallel Multipliers". Alta Frequenza, 36(5):349–356, May 1965.

[3] Booth.D," A Signed Binary Multiplication Technique", Quarterly Journal of Mechanics and Applied Mathematics, 4(2):236–240, June 1951

[4] Gary W. Bewick," fast multiplication: Algorithms and implementation" Ph.D.Thesis STANFORD UNIVERSITY, Februray1994

[5] Kathirvelu .M , Dr.T.Manigandan," Design and Imple - mentation of High speed ALU using Optimized PDP adder and Multiplier", Journal of Applied Sciences Research, 8(4): 2100-2108, 2012 ISSN 1819-544X.

[6] Ron S. Waters, Member, IEEE, and Earl E. Swartz - lander, Jr., Fellow, IEEE," A Reduced Complexity Wallace Multiplier Reduction," IEEE transactions on computers, vol. 59, no. 8, August 2010.

[7] Weste.N and Eshraghian.K, Principles of CMOS VLSI Design, A System Perspective. Reading, MA: Addison-Wesley, 1988.

[8] Aziz, S.M.; Sch. of Electr. & Inf. Eng., Univ. of South Australia, Mawson Lakes, SA, Australia ; Sicard, E. ; Ben Dhia, S."Effective Teaching of the Physical Design of Integrated Circuits Using Educational Tools “Education, IEEE Transactions on (Volume:53 , Issue: 4),03 November 2009