Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2018, 5(7):340-349**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Design and Comparative Performance Analysis of Various Multiplier Circuits

Garima Thakur, Harsh Sohal, Shruti Jain

Department of Electronics and Communication Engineering, JUIT, Solan, India

Abstract This paper discusses the comparison of various multiplier algorithms for different performance parameters like speed, area and power. We have studied and implemented different multipliers like Array multiplier, Wallace multiplier, and Vedic multiplier. Multipliers have slow processing of multiplication, so adders are used for summing up the partial products. Adders play an important role in multipliers. Verilog coding is used for comparative analysis of various multipliers. Using Xilinx ISE 14.1 Design Suite various multipliers are simulated and synthesized for Spartan 3E FPGA. We have proposed all the three multipliers using Kogge Stone Adder (KSA) which gave the best results compared to the existing work. Among all the proposed multipliers, Wallace multiplier results less delay (18.024ns) and more power (46mW).

Keywords Array multiplier, Wallace multiplier, Vedic multiplier, Xilinx ISE 14.1

Introduction

Multiplication is an important fundamental function in arithmetic operations. Many researchers have tried to design multiplier which offers either of the following- high speed, less area and low power consumption. The number which is to be added is called the multiplicand, the number of times which is added is called the multiplier and the result being given is known as the product. We have described various types of multipliers: Array multiplier, Wallace tree multiplier, Vedic multiplier. Designer mainly concentrates on efficient circuit design [1]. The characteristics of efficient multipliers are: its speed (should be high), accuracy, area (less no. of LUT's and slices should be occupied) and power (consumed power should be less). There are three main steps for implementation of multiplication process: generation of partial product, addition of partial product and final addition.

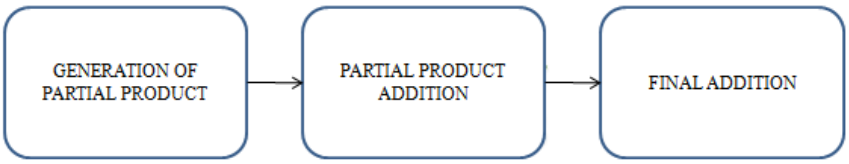


Figure 1: Block diagram of Multiplier architecture

Block diagram consist of three stages, in the first stage partial products are generated by multiplying bit by bit of multiplier and multiplicand. In the next stage there is an addition of generated partial product, this stage is complex and the speed of circuit was derived and last stage generates the output result by adding the two-row outputs. Parallel multipliers are the most rapid multiplier type. The earlier performances of multipliers are enhanced to develop number of techniques.

Let the multiplicand and multiplier be *A* and *B* respectively:

$A = a_{(M-1)}.a_{(M-2)}\ldots.a_1a_0 = \sum_{i=0}^{M-1} a_i.2^i$ (1)

$B = b_{(N-1)}.b_{(N-2)}\ldots.b_1b_0 = \sum_{i=0}^{N-1} b_i.2^i$ (2)

The value of their product $P = A \times B$ is given by Equation 3:

$$P = \sum_{i=0}^{M-1} \sum_{i=0}^{N-1} (a_i b_i . 2^{i+j}) \quad (3)$$

Equation 4 and 5 expressed signed binary number and Equation 6 defines the product of *A* and *B*.

$$A = - a_{M-1} . 2^{M-1} + \sum_{i=0}^{M-2} a_i . 2^i \quad (4)$$

$$B = -b_{N-1} . 2^{N-1} + \sum_{i=0}^{N-2} b_i . 2^i \quad (5)$$

The product $P = A \times B$ is given by Equation 6:

$$P = (-a_{M-1} . 2^{M-1} + \sum_{i=0}^{M-2} a_i . 2^i) \times (-b_{N-1} . 2^{N-1} + \sum_{i=0}^{N-2} b_i . 2^i) \quad (6)$$

Array Multiplier (AM): The structure of AM is regular and to move from one block to adjacent block; short wires are used. In VLSI its layout is efficient and simple. *N* partial product was generated when there is multiplication of multiplier and multiplicand bit by bit as expressed by Equation 3. Multiplication depends on Add/Shift algorithm. 4×4 AM is shown in Figure 2 [4].

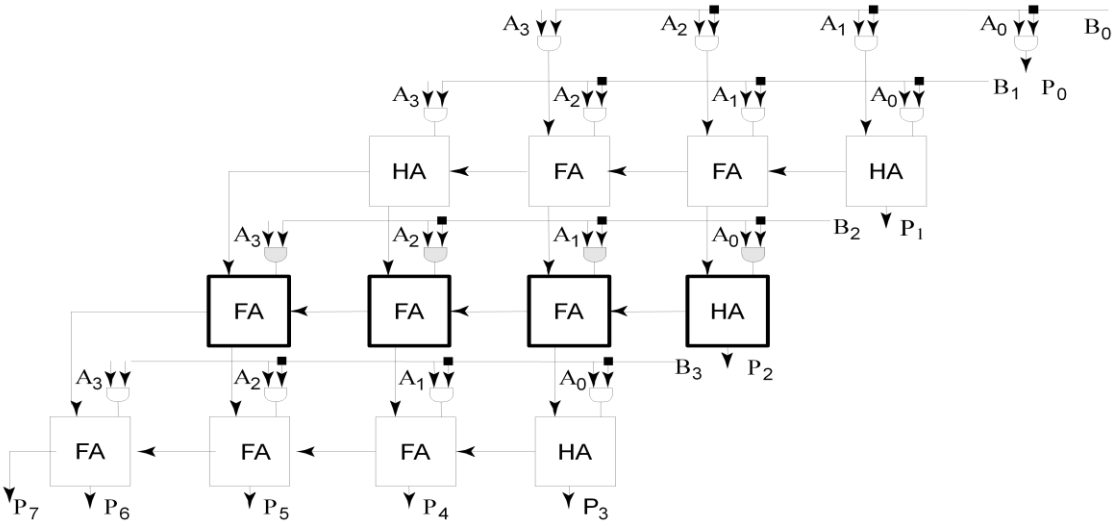


Figure 2: Array multiplier block diagram for a 4×4 multiplier

Wallace Multiplier (WM): This multiplier uses parallel addition of generated partial products, so it takes less time for accumulation than AM because in AM the partial products are added in series. The arrangement of WM is more complex and much less regular but it is high speed multiplier in comparison with other multipliers. 8×8 bit partial product reduction is shown in Figure 3. In this figure the two circled dots represent Half Adder (HA) and tree circled dots represent Full Adder (FA). After four stages partial product was reduced to two rows. There are so many ways to reduce the tree structure but only one method of reduction is shown [5]. For multiplication of two numbers the three steps are used.

- Formation of partial products
- Reduction of the partial products matrix into a two row matrix
- Using faster adder's addition of remaining two rows.

Vedic Multiplier (VM): The word "Vedic" is derived from the word "Veda" which means the store house of knowledge. Veda consists of 16 sutras which encapsulate the branches of Mathematics- geometry, calculus, arithmetic, trigonometry etc. These sutras are [6]: *Shunyamanyat (Anurupye), Chalana-Kalanabyham, Ekadhikina Purvena, Ekanyunena Purvena, Gunakasamuchyah, Gunitasamuchyah, Nikhilam Navatashcaramam Dashatah, Paraavartya Yojayet, Puranapuranaabhyam, Sankalana-vyavakalanabhyam, Shesanyankena Charamena, Shunyam Saamyasamuccaye, Sopaantyadvayamantyam, Urdhva-tiryakbyham, Vyashtisamanstih, Yaavadunam.*

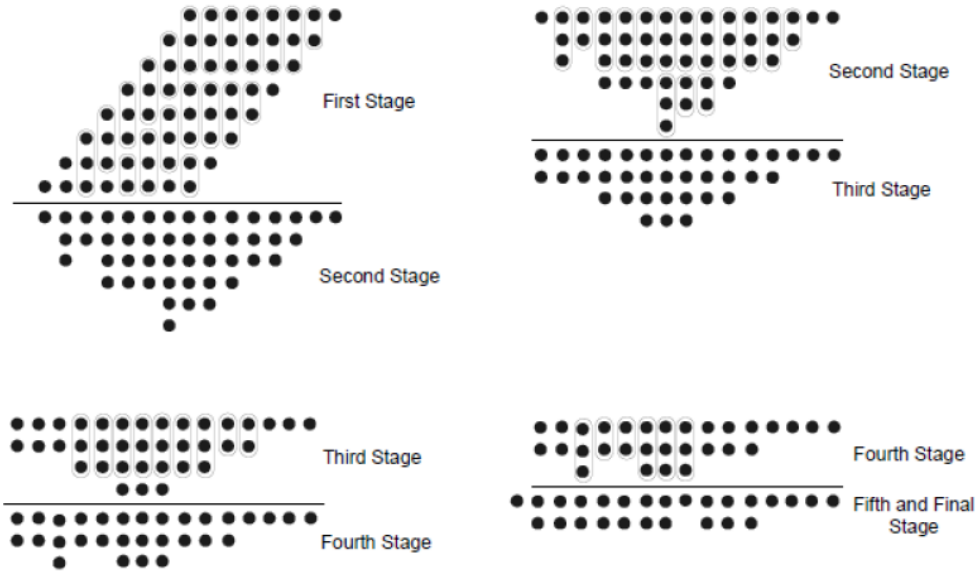


Figure 3: Wallace tree for an 8×8 partial product tree

Vedic Multiplier using "Urdhva Tiryakbyham" Sutra: In Sanskrit literature the 'Urdhva' means 'vertically' and 'Tiryakbyham' means 'crosswise'. Urdhva Tiryakbyham is applicable to all cases of multiplication. In one step the algorithm produces sum and partial product. Once the number of bits was increased, multiplier is advantageous as compared to other multipliers as its area and gate delay increases slowly. Let's consider one example we have to multiply 131 × 121. Table shows the different steps of multiplication [19].

Step	Explanation	Process	Result
1.	The numbers that lie on ones place are multiplied vertically and output is generated and stored result in ones place of the final result	1 3 1 ↕ 1 2 1 1	Result=1 Carry=0
2.	The numbers that lie on ones and tens place are multiplied by crossover multiplication, resultant was added. Final result was stored on tens place	1 3 1 ⤫ 1 2 1 5 1	Result=3+2=5 Carry=0
3.	The numbers that lie on ones and hundredth place are multiplied by crossover multiplication and number that lie on tens place are multiplied by vertical multiplication. The result of these multiplications is summed and final result was stored in hundredth place.	1 3 1 ✳ 1 2 1 8 5 1	Result=1+6+1=8 Carry=0
4.	The numbers that lie on tens and hundredth place are multiplied by crossover multiplication and result was stored on thousand place	1 3 1 ⤫ 1 2 1 5 8 5 1	Result=3+2=5 Carry=0

5.	Finally, vertical multiplication of two numbers on hundredth place are multiplied, 1 bit output was generated and stored result in ten thousand place of the final result	1 3 1 ↕ 1 2 1 1 5 8 51	Result=1 Carry=0

Nikhilam Sutra: It literally means "all from 9 and last from 10" i.e. subtract last digits from 10 and rest of digits from 9 and when large numbers are involved it is more efficient. To perform the multiplication, the compliment of the large number was find out from its nearest base.

For example: 94×96

Nearest Base =100

94 – 100 = -6

96 – 100 = -4

94 -6

96 -4

9 0 2 4 ⟶ Result

1. Both the numbers are close to 10 power (base 100).
2. 94 is 6 less than 100 & 96 is 4 less than 100.
3. (-6)×(-4) = 24
4. 94-4 or 96-6 = 90
5. Final result = 9024

Later section of the paper is organized as: Section 2 provides a brief literature review of the related work on multipliers. Section 3, explains the simulation work done for implementation of 4-bit multiplier. In section 4 , design of high speed multipliers was proposed and finally conclusion and future work was explained.

Literature Review

Akhter S *et al.* 2017 [8]: In this paper various digital adders are used for comparative analysis of Vedic multiplier. Using CBL adder the 8-bit Vedic multiplier is 20% faster than BEC and is approximately 5% faster in terms of delay than RCA-CSA, SQRT-CSA and RCA. They have calculated different result in term of delay, area and leakage power as the width size increases.

Gowreesrinivas K V *et al.* 2016 [9]: This paper used different types of adders and by incorporating Vedic multiplier, a new type of single precision floating point multiplier was developed. The main problem in digital signal processor of the single precision floating point multiplier was the optimization of the speed and area. By reducing interconnections and complexity the overall performance can be improved. It was observed that using combination of prefix sklansky adder and Vedic multiplier has better performance in terms of complexity and speed in single precision multiplier.

Gokhale G R *et al.* 2015 [10]: In this paper Vedic multiplier was implemented by using lesser number of gates and area, which was required by proposed CSLA. The Booth multiplier has more area and delay compared to proposed Vedic multiplier, so it is superior. In the architecture of Vedic multiplier the addition block plays a important role for increasing and decreasing the performance of the circuit.

Murugeswari S *et al.* 2014 [11]: In this paper a low power and an area efficient modified Wallace and truncated multiplier was implemented by using full adder which was based on mux. In the end it was concluded that reduction in area of modified truncated multiplier shows improvement in device utilization compared to modified Wallace multiplier.

Anjana R *et al.* 2014 [12]: They proposed a novel high speed architecture by combining Kogge stone adder with the multiplier to design the fastest multiplier.

Rajaram S *et al.* 2011 [13]: This paper proposed that multipliers have less delay than the conventional multiplier. Proposed multiplier was Wallace multiplier which used Parallel prefix adder at the final stage, so there was an improvement in multiplier.

Kesava R B S *et al.* 2016 [14]: In this paper a simple approach was proposed for Wallace tree multiplier using CSLA , so to reduce the area. They implemented CSLA with BEC in Wallace tree multiplier to occupying less

power, less area and memory when compared to Wallace tree multiplier using CSLA and Wallace tree multiplier.

Srikanth S *et al.* 2016 [15]: In this paper a modified full adder was proposed by using multiplexers and XOR gate. In Wallace tree multiplier, the modified full adder was incorporated in the reduction stage. An average delay, power and area reduction was achieved compared to existing method.

Paradhasaradhi D *et al.* 2014 [16]: This paper proposed an area efficient Wallace tree multiplier which was implemented by using CBL and was based on square root CSLA. There was reduction in delay and area by reducing the number of gates. Duplicated adder cells are removed in the regular CSLA by sharing CBL term.

Implementation of 4-Bit Multipliers

For implementation of 4-bit multipliers we have used Xilinx ISE 14.1 Design Suite, area and delay values are calculated from synthesis report while power was calculated by Power analyzer in which we calculated IOs Power and Leakage Power. The terms used in Table 1 are explained as follows:

a) **Look-Up Tables (LUT):-** In Configurable Logic Block (CLBs) function generators are implemented using LUT.
b) **Slices: -** In FPGA slices are the basic building block components. All of the Flip flop and LUT's are packed into slices after mapping.
c) **Input/Output Block (IOB):-** In FPGA device, input and output functions are implemented from the grouping of basic elements. Such collection and grouping of basic elements are termed as an IOB.
d) **Delay: -** Delay is the time required for the input to be propagated to the output. There are two types of delays: Router delay which is app 40% of total delay and Logic delay which is more than 50% of total delay.
e) **Power: -** Power dissipation of two types: static (due to current leakage in the transistors of an FPGA) and dynamic (due to signal alteration).

The comparison of different multipliers in terms of area, delay and power is shown in Table 1

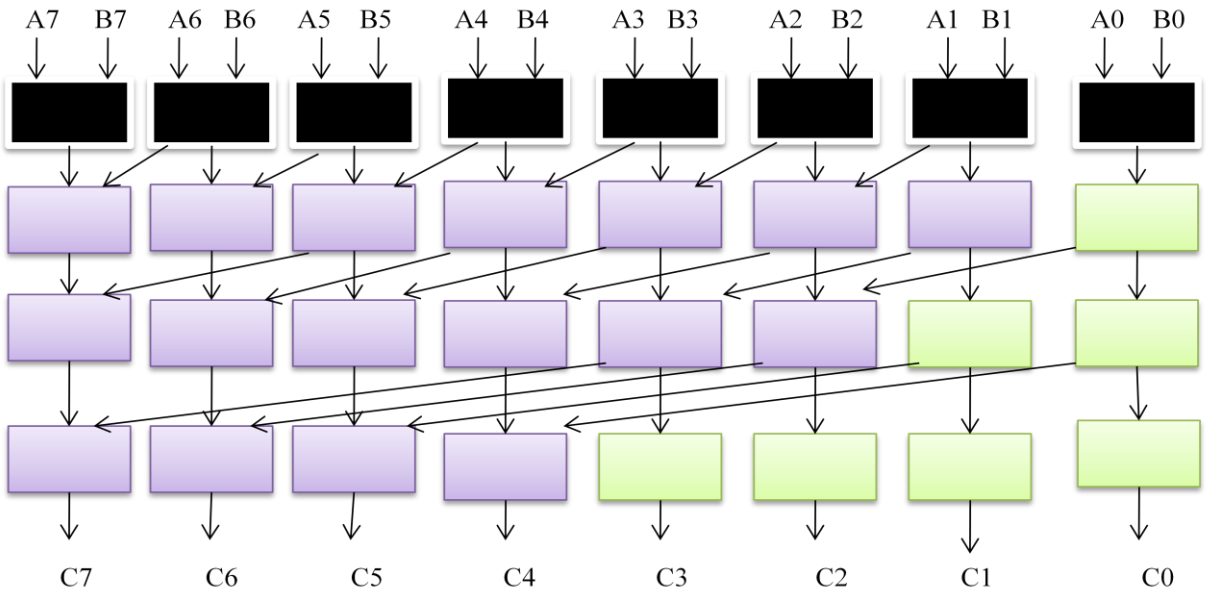
Table 1: Comparison of different 4-bit multipliers

Sr. No.	Design	No. of 4 I/P LUT	No. of occupied slices	No. of bonded IOB		Delay (ns)		Power Total (W)		Power Delay Product
				I-Buf	O -Buf	Logic Delay	Router Delay	Power IOs	Power Leakage	
1.	4 bit Array multiplier	29	17	8	8	9.171	4.486	0.001	0.034	0.4779
2.	4 bit Wallace multiplier	33	19	8	9	7.947	3.928	0.001	0.034	0.4156
3.	4 bit Vedic multiplier	39	22	9	9	8.837	3.995	0.029	0.034	0.8084

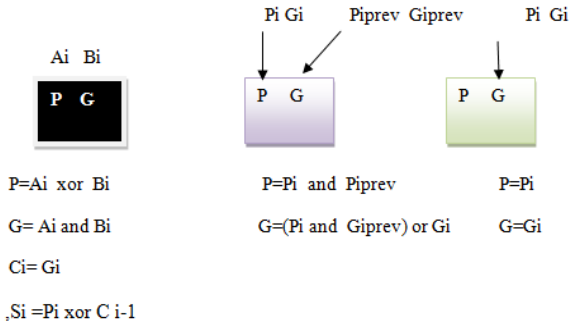
4 bit WM gives the best result as its delay is less and power is less. 4 bit multipliers are used to implement 8 bit multipliers architecture. The speed and power of multiplier depends on the architecture of the multiplier.

Proposed Design

8 bit Multiplier: 8-bit multipliers are implemented using Kogge stone adder (KSA). There are different types of adders like Carry Select Adder (CSLA), Carry Skip Adder (CSkA), Carry Lookahead Adder (CLA), Ripple Carry adder (RCA) etc. We have implemented all the adders among all the adders KSA was the best in terms of speed and it is basically a prefix based adder [7]. An illustration of 8-bit KSA is shown in Figure 4.



(a)



(b)

Figure 4: (a) 8- bit KSA , (b) representation of each block

We have implemented AM, VM and WM using KSA for different performance parameters. In terms of delay, WM have best delay i.e 18.024ns but there was increased power consumption. Each multiplier has its own advantage and disadvantage depending on logic we are using.

8 bit Multiplier Architecture: We have implemented 8 bit multiplier using 4-bit AM, VM and WM. From the synthesis report, the performance parameters like area and delay are obtained and from power analyzer power was calculated which was shown in Table 2. Flow chart of 8 X 8 multiplier architecture is shown in Figure 5.

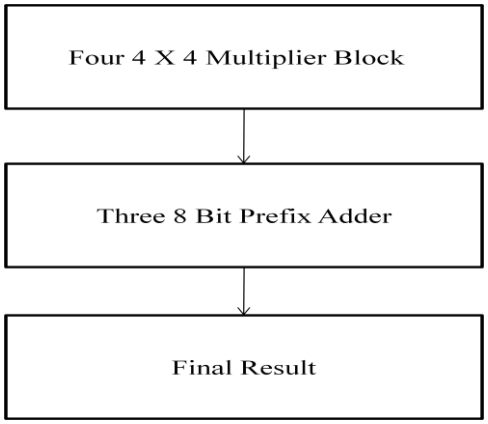


Figure 5: Flow Chart of 8 X 8 multiplier

Table 2: Area, Delay and Power calculation of 8 bit different multipliers

Sr. No.	Design	No. of 4 I/P LUT	No. of occupied slices	No. of bonded IOB		Delay (ns)		Power Total (W)		Power Delay Product
				I – Buf	O- Buf	Logic Delay	Router Delay	Power IOs	Power Leakage	
1.	Wallace_KSA	183	104	19	17	11.285	6.739	0.012	0.034	0.8291
2.	Array_KSA	171	98	19	17	13.121	7.850	0.001	0.034	0.7339
3.	Vedic_KSA	216	120	17	17	14.011	8.104	0.001	0.034	0.7740

8 X 8 Array Multiplier Block: 8 by 8 AM was implemented by considering two 8-bits binary numbers A = A7 A6 A5 A4 A3 A2 A1 A0 and B = B7 B6 B5 B4 B3 B2 B1 B0. To implement 8 X 8 AM, 4 X 4 Array multipliers are used to generate partial products. For addition of generated partial product, three KSA of 8 bits are used. We have used four 4 X 4 AM block, in the first block least significant bits (LSBs) of A and B are multiplied to generate S [3:0] of final result. In second block most significant bits (MSBs) of A was multiplied with LSBs of B to generate input bits for first block of KSA and in third block LSBs of A was multiplied with MSBs of B to generate input bits for first block of KSA. In fourth block, MSBs of A and B are multiplied to generate input bits for third block of KSA. Carry generated by first two KSA are ORed. ORing these two KSA, a carry was generated which was applied a input to next KSA. In some blocks of KSA, zero inputs are applied according to the requirement. KSA arrangements are made in such way that the speed of working was increased. Finally sum [15:0] and carry (C3) was generated and the architecture of 8 X 8 AM was shown in Figure 6.

Table 3 gives the comparison of designed 8-bit AM with the existing multipliers. Our proposed multiplier circuit gives the best delay which is 20.971 ns in comparison to Maiti A *et al* 2016 [18] whose delay was 25.3 ns and Thomas A *et al* 2016 [17] whose delay was 44ns. We also calculated power which is more in our case (35 mW) in comparison to Maiti A *et al* 2016 [18] whose power was 0.0606 mW.

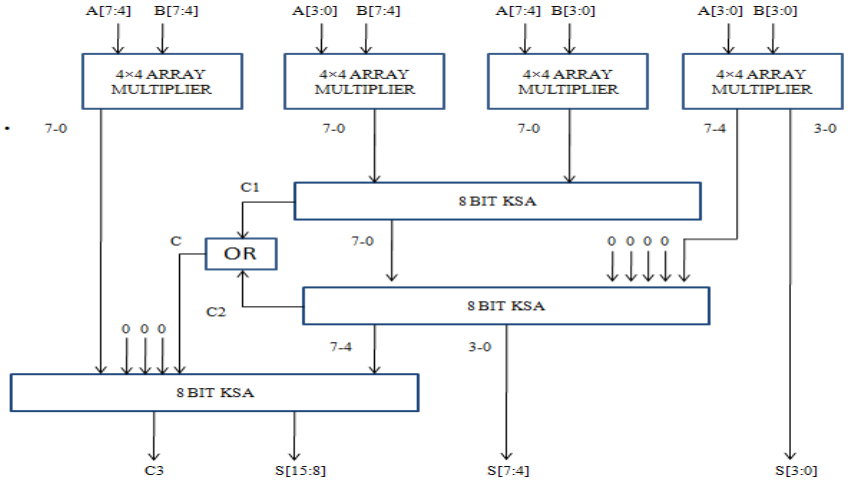


Figure 6: 8×8 Array multiplier architecture

Table 3: Area, Delay and Power calculation of 8 bit Array Multiplier

	Width	No. of LUTs	Delay(ns)	Power(mW)
Propsed work Array Multiplier	8	171	20.971	35
Thomas A *et al* 2016[17]	8	126	44	-
Maiti A *et al* 2016[18] Using CMOS	8	-	25.3	0.0606

a) 8 X 8 Vedic Multiplier Block: Figure 7 represents the block diagram of 8 X 8 vedic multiplier. The steps were same as explained in array multiplier except the 4 X 4 array multiplier was replaced by 4 X 4 vedic multiplier.

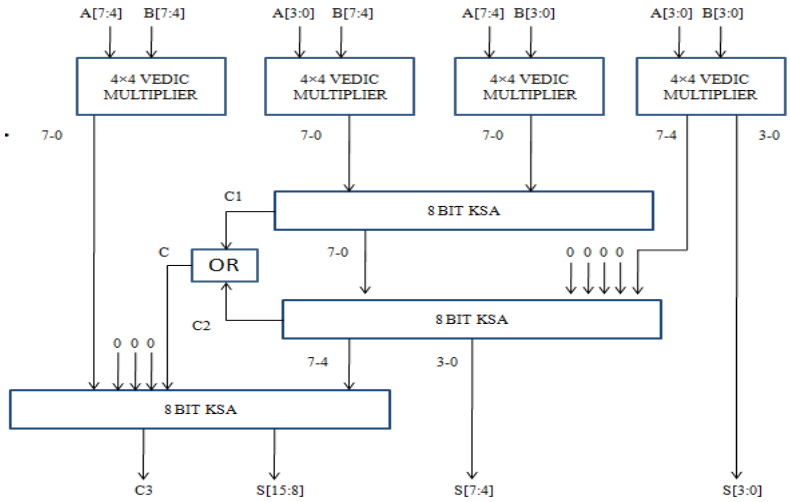


Figure 7: 8×8 Vedic multiplier architecture

Table 4 gives the comparison of designed 8-bit VM with the existing multipliers. Our proposed circuit of which multiplier gives the best delay which was 22.115ns in comparison Gokhale GR *et al* 2015 [10] whose delay is 44.358ns and Thomas A *et al* 2016 [17] whose delay is 34 ns using RCA and 30 ns using CLA. We have also calculated Power which was 35mW while Gokhale GR *et al* 2015[10] and Anjana R *et al* 2014 [12] has not reported any power. Anjana R *et al* 2014[12] calculated difference between logic delay and router delay which is 5.588ns and our proposed circuit difference between logic delay and router delay is 5.907ns which is more but the no. of LUTs required are less than Anjana R *et al* 2014 [12] .

Table 4: Area, Delay and Power calculation of 8 bit Vedic Multiplier

	Width	No. of LUTs	Area(gate count)	Delay(ns)	Power(W)
Propsed work Using KSA	8	216	-	22.115	0.035
Gokhale GR *et al* 2015[10]	8	-	1293	44.358	-
Anjana R *et al* 2014[12]	8	309	-	5.588	-
Thomas A *et al* 2016[17] Using RCA	8	166	-	34	-
Thomas A *et al* 2016[17] Using CLA	8	167	-	30	-

b) 8 X 8 Wallace Multiplier Block: Figure 8 represents the block diagram of 8 X 8 wallace multiplier. The steps were same as explained in array multiplier except the 4 X 4 array multiplier was replaced by 4 X 4 wallace multiplier.

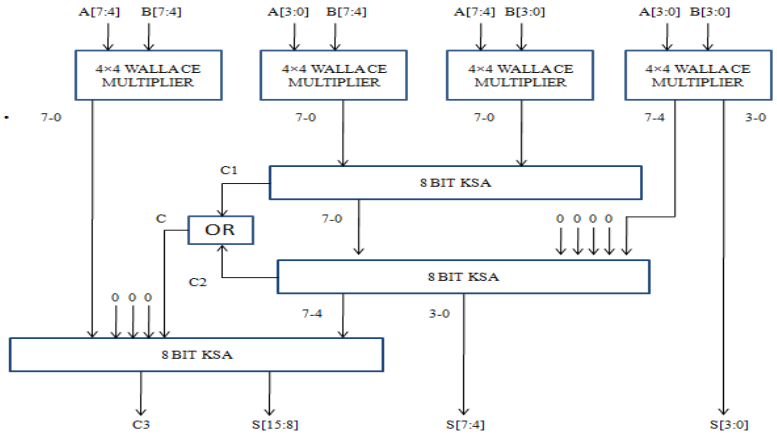


Figure 8: 8×8 Wallace multiplier architecture

Table 5 gives the comparison of designed 8-bit Wallace multiplier with the existing multiplier. Our proposed circuit of which multipliers gives the less delay which was 18.024 ns in comparison with Rajaram S *et al* 2011[13] whose calculated delay is 27.457 ns and Thomas A *et al* 2016 [17] whose delay is 39 ns. We have also calculated power which is less (46mW) then Murugeswari S. *et al* 2014 [11] whose power is 264mW (using full adder), 231mW (using mux based full adder) while Rajaram S *et al* 2011 [13] has not reported any power.

Table 5: Area, Delay and Power calculation of 8 bit Wallace Multiplier

	Width	No. of occupied slices	No. of LUTs	Area (gate count)	Delay (ns)	Power (mW)
Proposed work Using KSA	8	104	183	-	18.024	46
Rajaram S *et al* 2011[13]	8	-	-	-	27.457	-
Murugeswari S. *et al* 2014[11] using Full adder	8	87	163	-	17.223	264
Murugeswari S. *et al* 2014[11] using MUX based Full adder	8	84	155	-	17.789	231
Thomas A *et al* 2016[17]	8	-	133	-	39	-

It can be observed that the proposed design for 8bit Wallace Multiplier has better delay performance which was the desired goal of this research work. In future we will work in the applications of multipliers [19, 20].

Conclusion

The performance of any circuit in VLSI design limits by the constituent factors like power, delay and area. In this paper Array multiplier, Vedic multiplier and Wallace multiplier are implemented using KSA. It is concluded that KSA requires less delay and power as compared to other adders, so it is best suited for implementation of modified multiplier. Wallace multiplier has less delay i.e. 18.024ns compared to other multipliers but there was increase in power consumption. The design was tested and verified by Verilog HDL coding and simulation was carried out in Xilinx ISE 14.1 design suite and synthesized for Spartan 3E FPGA. Future work may be dedicated to decrease the power consumption of multipliers and used efficient multiplier in any application.

References

[1]. Soniya & S. Kumar ,"A Review of Different Types of Multipliers and Multiplier-Accumulator Unit", International journal of Emerging Trends and Technology in Computer Science (IJETTCS),Volume 2,Issue 4. July-August 2013.

[2]. V. S. Mamatha & S. Babitha, "Area and Power efficient Wallace Multipler", Imperial journal of Interdiscilinary Research (IJIR),Vol. 2,Issue 13,2016

[3]. K. K. Parhi , "Vlsi Digital Signal Processing Systems", John Wiley and Sons Inc., 2010.

[4]. J. M. Rabaey, A. Chandrakasan and B. Nikolic, Digital Integrated Circuits, A Design Perspective, Prentice Hall, Upper Saddle River, NJ, 2003.

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

[6]. H. Budhiraja, M. Syed & A. Ramya, "Verilog Implementation Of Vedic Multiplier", International Journal of Advancement in Engineering Technology, Management and Applied Science, Vol. 3, Issue 5, May 2016, ISSN No:2349-3224.

[7]. M. Nandini & A. Jayavani, "High Speed and Power Optimized Parallel Prefix Modulo Adders using Verilog", International Journal of Advanced Technology and Innovative Research, Vol.07, Issue.01, January 2015, pp. 0216-0133.

[8]. S. Akhter, V. Saini & J. Saini, " Analysis of Vedic Multiplier using Various Adder Topologies", 4th International Conference on Signal Processing and Integrated Networks (SPIN), 2017.
[9]. K. V. Gowreesrinivas & P. Samundiswary, "Comparative Study on Performance of Single Precision Floating Point Multiplier using Vedic Multiplier and different types of Adders", International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2016.
[10]. G. R. Gokhale & S. R. Gokhale, "Design of Area and Delay Efficient Vedic Multiplier Using Carry Select Adder", International Conference on Information Processing (ICIP), Dec 16-19, 2015.
[11]. S. Murugeswari & S. K. Mohideen, "Design of Area Efficient and Low Power Multipliers using Multiplexer based Full Adder", 2nd International Conference on Current Trends in Engineering and Technology, ICCTET, 2014.
[12]. R. Anjana, B. Abishna, M. S. Harshitha, E. Abhishek, V. Ravichandra & M. S. Suma, "Implementation of Vedic Multiplier using Kogge-Stone Adder", International Conference on Embedded Systems - (ICES), 2014.
[13]. S. Rajaram & K. Vanithamani, "Improvement of Wallace multipliers using Parallel prefix adders", International Conference on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN), 2011.
[14]. R. B. S. Kesava, B. L. Rao, K. B. Sindhuri & N. U. Kumar, "Low Power And Area Efficient Wallace Tree Multiplier Using Carry Select Adder With Binary To Excess-1 Converter", Conference on Advances in Signal Processing (CASP), Jun 9-11, 2016.
[15]. S. Srikanth, J. Thahira Banu, G. Vishnu Priya & G. Usha, "Low Power Array Multiplier Using Modified Full Adder", 2nd IEEE International Conference on Engineering and Technology (ICETECH), 17th- 18th March, 2016.
[16]. D. Paradhasaradhi, M. Prashanthi & N. Vivek, "Modified Wallace Tree Multiplier using Efficient Square Root Carry Select Adder", International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), 2014.
[17]. Thomas, A. Jacob, S. Shibu & S. Sudhakaran, "Comparison of Vedic Multiplier with Conventional Array and Wallace Tree Multiplier", International Journal of VLSI System Design and Communication Systems, Vol. 04, Issue. 04, April 2016.
[18]. Maiti , K. Chakraborty, R. Sultana & S. Maity, "Design and implementation of 4-bit Vedic Multiplier", International Journal of Emerging Trends in Science and Technology, Vol. 03, Issue 05, Pages 3865-3868, May 2016.
[19]. S. S. Gadekar & M. S. Badmera, "Design of Low Power Multiplier Architectures Using Vedic Mathematics", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 6, Issue 4, April 2017.
[20]. R. Subalakshmi, "Low Power Multiplier For DCT Applications", International Journal of Science, Engineering and Technology Research (IJSETR), Vol. 6, Issue 2, February 2017, ISSN: 2278 -7798.