Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 10, Number 5 (2017) pp. 1257-1272
© Research India Publications
http://www.ripublication.com

A Review on Comparative Performance Analysis of Different Digital Multipliers

Vikas Kaushik[1] and Himanshi Saini[2]

[1, 2] ECE Department, Deenbandhu Chhotu Ram Univ. of Science & Technology, Sonipat, India.

Abstract

The importance of digital electronics is increasing day-by-day in our day-to-day life. Digital multipliers have great importance in designing modern gadgets, in digital signal processing and in many other applications. For the improvement in performance of modern devices and softwares, there is a need of designing a multiplier having high speed, less area, low power consumption, simple and regular design. Here is a review presented by comparing performance of various multipliers. The multipliers are mostly compared in terms of delay power consumption and area required etc. parameters. This paper brings all important digital multipliers together for comparative analysis. This comparative analysis helps us to select one most suitable multiplier for a particular application.

Keywords: digital multipliers, comparison, delay, power, area, vedic, wallce, booth.

I. INTRODUCTION

There are many types of multiplier-architectures. Among them, (1) Array multipliers are the traditional multipliers and are analyzed for reference purposes. In addition to these, followings are the fanciest multipliers, e.g., (2) Vedic series of multipliers, (3) Booth series of multipliers, (4) Wallace-tree multipliers, (5) Dadda multiplier, and (6) Braun multiplier.

This work is divided into two phases. In 1st phase, which has series-wise comparison, we will analyze group of multipliers from a particular series separately. This is to be done to find the best one multiplier from each series. After analyzing all multipliers, we will make a list of seven multipliers out of all, which have high performance in their respective groups. These seven multipliers include the Array-multiplier for reference purpose to compare performance. In 2nd phase, which has inter-series comparison, we will compare these seven multipliers and tabulate them together. This enables us to analyze relative performance of these multipliers. This comparison helps us to select a most suitable multiplier for an application.

Now there is a list of all multipliers to be analyzed. (1) In Array multipliers, three array multipliers are taken for analysis. These are distinguished on the basis of adder used to design multiplier; (i) Array Multiplier using RCA (Ripple carry adder), (ii) using CLA (Carry Look-ahead adder), and (iii) using CSA (Carry save adder). (2) In Vedic series of multipliers, there are seven multipliers are taken for analysis, (iv) Vedic Multiplier using RCA, (v) using CLA, (vi) using CSA, (vii) using KSA (Kogge Stone Adder), (viii) using Sklansky Adder, (ix) using Brent Kung Adder, (x) using Han Carlson Adder, (3) In Booth series of multipliers, there are mainly two types of multipliers, (xi) Conventional Booth multiplier, (xii) Modified Booth multiplier, (4) In Wallace-tree multipliers, there are three multipliers; (xiii) Conventional Wallace-tree, (xiv) Modified Wallace-tree, (xv) Wallace-Booth multiplier. (5) In Dadda multipliers, (xvi) Regular dadda multipliers using RCA, (xvii) using CLA, (xviii) Partitioned dadda multipliers (xix) using RCA, (xx) using CLA. (6) In Braun multipliers, two multipliers are taken here for comparison, (xxi) using RCA, (xxii) Using KSA. Therefore, there are almost 22 multipliers to be analyzed.

1.1 ARRAY MULTIPLIERS

These are the conventional multipliers having regular structures. Add and shift algorithm is used for its operation and hence its circuit is based on this algorithm. By direct mapping of the manual multiplication into hardware, an Array multiplier circuit can be developed [2][14]. An array of adder circuits can be used to accumulate partial products. The partial products are generated by multiplying the multiplicand with each bit of multiplier. The bit order decides the amount of shift of partial products. At the final stage, the partial products are added. The number of generated partial products is equal to that of multiplier bits. If multiplier length is equal to N, then N-1 numbers of adders are required to implement array multiplier [3].

1.2 VEDIC SERIES OF MULTIPLIERS

Vedic Mathematics is an ancient system of mathematics existed in India. In this eminent approach, methods of basic arithmetic are simple, powerful and logical. Another advantage is its regularity. These advantages make Vedic Mathematics an important topic for research [4][13]. Vedic Mathematics rules are mainly based on sixteen Sutras. Out of these sixteen Sutra's Urdhva Triyakbhyam sutras and Nikhilam

sutras are used for multiplication. Vedic multipliers are considered to be the best compared with conventional multipliers and Urdhva Triyakbhyam Sutra based multiplication is more efficient compared to that of Nikhilam Sutra [20]. Implementation of Vedic mathematics on FPGA is easy due to its regularity and simplicity [4][15]. All partial products required for multiplication are calculated much before actual multiplication begins. This is the big advantage of this multiplication. Based on the Vedic Mathematics algorithm, these partial products are added to obtain final product which leads to a very high speed approach [21]. Multiplier designs based on Vedic mathematics are with high speed and consume relatively low power. Multipliers are the basic and key blocks of a Digital Signal processor. Multiplication is the key process in improving the computational speed of Digital Signal Processors [16][17]. Convolution, Fast Fourier transforms and various other transforms make use of multiplier blocks [11]. Among various methods of multiplications in Vedic mathematics, Urdhva Tiryagbhyam is efficient. Urdhva Tiryagbhyam is a general multiplication formula applicable to all cases of multiplication [17].

1.3 BOOTH SERIES OF MULTIPLIERS

1.3.1 Convetional Booth Multiplier

There is no need to take the sign of the number into deliberation in dealing with unsigned multiplication. However in signed multiplication the process will be changed because the signed number is in a 2's compliment pattern which would give a wrong result if multiplied by using similar process for unsigned multiplication [6]. Booth's algorithm is used for this. Booth's algorithm preserves the sign of the result. Booth multiplication allows for smaller, faster multiplication circuits through encoding the signed numbers to 2's complement, which is also a standard technique used in chip design, [6] and provides significant improvements by reducing the number of partial product to half over "long multiplication" techniques. Radix 2 is the conventional booth multiplier.

(i) Radix 2

In booth multiplication, partial product generation is done based on recoding scheme e.g. radix 2 encoding. Bits of multiplicand (Y) are grouped from left to right and corresponding operation on multiplier (X) is done in order to generate the partial product [19]. In radix-2 booth multiplication partial product generation is done based on encoding which is as given by Table1. Parallel Recoding scheme used in radix-2 booth multiplier is shown in the Table 1.

1.3.2 Modified Booth Multiplier

(ii) Radix 4

One of the solutions attaining high speed multipliers is to improve parallelism. It helps in decreasing the number of consecutive calculation stages [31]. The Original version of Booth's multiplier (Radix – 2) had two drawbacks [07],[28]. (1). The

number of Add or Subtract operations became variable and hence became difficult while designing Parallel multipliers. (2). The Algorithm becomes disorganized when there are isolated 1s. These problems are overthrown by using Radix 4 Booth's algorithm which can browse strings of three bits with the algorithm. The above recoding has the nice feature that they translate into the partial products shown in table 2.

Table 1. Booth recoding for radix 2 [19]

Q_n	Q_{n+1}	Recoded Booth	Operation
0	0	0	Shift
0	1	+1	Add x
1	0	-1	Subtract x
1	1	0	Shift

Table 2. Booth recoding for radix 4 [6]

Multiplier Bits Block			**Recoded 1-bit pair**		**2-bit booth**	
i+1	i	i-1	i+1	i	Multiplier Value	Partial Product
0	0	0	0	0	0	Mx0
0	0	1	0	1	1	Mx1
0	1	0	1	-1	1	Mx1
0	1	0	1	0	2	Mx2
1	0	0	-1	0	-2	Mx-2
1	0	1	-1	1	-1	Mx-1
1	1	0	0	-1	-1	Mx-1
1	1	0	0	0	0	Mx0

(iii) Radix 8

Radix-8 Booth recoding is based on the same algorithm as that of Radix-4. But in radix 8, we take quartets of bits instead of triplets [27][1]. This algorithm reduces the number of partial products to n/3, where n is the number of multiplier bits. Thus it allows a gain in time in the partial products summation. Radix 8 booth encoding multiplier uses 4-bit encoding scheme. But the circuit complexity increases as compared to previous version. Radix-8 booth multiplier also lacks in most parameter like delay, speed from radix 4 booth multiplier due to the complexity of the circuit.Radix-8 booth multiplier which scan strings of four bits. An algorithm similar

to previous radix algorithm is made for radix8 booth multiplier. After encoding the multiplier the resulting partial product will be 0,y,+2y,+3y,+4y,-4y,-3y,-2y,-y where y represent the multiplicand. We have to represent all the no. in 2‘s complement form [28].

(iv) Radix 16

For binary radix-16 Booth recoded multipliers has the maximum height of the partial product columns equal to [n+1/4] for n = 64-bit unsigned operands can be done [26] [29].

(v) Radix-32

Radix32 booth multiplier reduces the no. of partial products by one fifth. An algorithm similar to previous radix algorithm is made for radix 32 booth multiplier [28].

1.4 WALLACE-TREE SERIES OF MULTIPLIERS

The Wallace tree method is used for implementing high speed designs. Two rows of partial products are produced that can be added in the last stage [27]. There is a reduction in critical path and the number of adders when compared to the conventional parallel adders. The Wallace tree structure can be with 3:2 compressors and 4:2. In this regard, we can expect a significant reduction in amount of computing in multiplications [27].

1.4.1 Convetional Wallace-Tree Multiplier

In terms of performance characteristics, designing a multiplier by using wallace tree architecture is superior over other architectures [22]. A multiplier designed by using wallace tree architecture is known as a wallace multiplier. As compared to other multiplier architectures, Wallace multiplier consumes less power and its switching speed is faster. Due to the intense interest of researchers in wallace multiplier, as result of which, different architectures are introduced to design a better wallace multiplier architecture. A conventional wallace multiplier and a reduced complexity wallace (RCW) multiplier are two architectures among them [22].

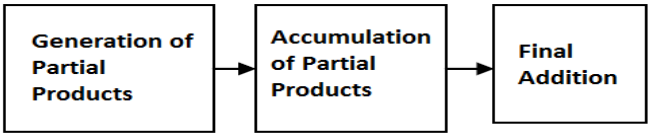

Fig.1. Block diagram of wallace-tree multiplier [22]

1.4.2 Modified Wallace-Tree Multiplier

To reduce the complexity of the reduction tree, Waters and Swartzlander presented a modification in the Traditional Wallace (TW) multiplier [23]. In this the partial products are readjusted in a reverse pyramid style. It makes it easy to analyze the tree for efficient reduction. The number of stages for RCW multiplier remains the same as that of TW multiplier. RCW tries to reduce the partial product tree using only full adders and half adders are used only where they are necessary to satisfy the number of rows in a stage. This approach enables RCW multiplier to reduce the area of the reduction process. However, a much larger final adder as compared to TW multiplier is used in RCW multiplier[23]. The size of the final adder for an N-bit RCW multiplier can be computed by: Final Adder (RCW) = $2N-2$.

1.4.3 Wallace-Booth Multiplier

Booth recoding algorithm and compressor adders are used in Booth Recoded Wallace Tree Multiplier for its realization [24]. In this architecture, two major things with differences are that Booth Recoding algorithm is introduced to generate and reduce the number of the partial products of multiplier, whereas, 3:2, 4:2, and 5:2 compressor structures are introduced to reduce the number of partial product addition stages. Critical delay path is minimized by replacing the XOR blocks with multiplexer blocks in these compressors. Using Carry Select Adder final two rows are summed to produce the final result. This architecture has the advantage of higher speed and lower area [25].

1.5 DADDA MULTIPLIER

The summation proceeds in a more regular way but slower manner in array multiplication scheme. Using array scheme, at each stage of the summation only one row of bits in the matrix is eliminated [12]. The partial products are generated by using array of AND gates in a parallel multiplier. The main concern is about the summation of the partial products, and the time taken to perform this summation determines the maximum speed at which a multiplier can operate. The Dadda scheme minimizes the number of adder stages required to perform the summation of partial products. This is achieved by using half and full adders. The task is to reduce the number of rows in the matrix number of bits at each summation stage [12]**.** Dadda multiplier is a refinement of the parallel multipliers presented by Wallace[30].

1.6 BRAUN MULTIPLIER

Braun multiplier is a basic parallel multiplier. It is also called as carry save array multiplier. It is easy to design one and there is no need of logic registers. The structure consists of adders arranged in the iterative way and array of AND gates. This can be called as non – additive multipliers [5]. In the internal structure, two main things are that each products can be generated in parallel with the AND gates, and each partial

product can be added with the sum of partial product which has previously produced by using the row of adders.

The carry out will be shifted one bit to the left or right and then it will be added to the sum which is generated by the first adder and the newly generated partial product. Because the sum and carry signals are both in the critical path, in the multiplier array a full adder with balanced carry and sum delays are desirable. The main disadvantages of Braun multiplier are two; (i) the number of components required in building blocks of Braun Multiplier increases quadratically with the number of bits; (ii) at the last stage of full adders, the potential susceptibility to Glitching problems is due to exploitation of the Ripple Carry Adder (RCA) [32].

II. PHASE- 1 (SERIES-WISE COMPARISON)

2.1 COMPARISION SUMMARY OF VEDIC MULTIPLIERS

After having analysis of table 3, table 4, table 6 and table 7 got the comparison idea which shows that Vedic multiplier using HCA (han carlson adder) is the best multiplier among the vedic series of multipliers. In addition to this, table 5 is also showing a better vedic multiplier using skalansky adder.

Table 3. [4] Vedic Urdhava multiplier

Multipliers	Delay (ns)	No. of Slices
8-bit Urdhava	12.588	133 out of 46560
8-bit Urdhava with mux based adder	9.130	107 out of 46560
16-bit Urdhava	20.338	742 out of 46560
16-bit Urdhava with mux based adder	16.994	594 out of 46560

Table 4. vedic multipliers [4]

Multipliers	%age improvement in speed	%age improvement in area
8-bit	27.47%	19%
16-bit	16.44%	20%

Table 5. delay comparison [11]

Methode	Logic (ns)	Route (ns)	Toatl Delay(ns)
Vedic mul. With 8-bit RCA	15.717	10.136	25.853
Vedic mul. With 8-bit Skalansky adder	14.963	9.351	24.314

Table 6. Different vedic multipliers [21]

No. of Bits	Adder	Delay (ns)	Slices	LUTs
6-bit	RCA	15.527	7	12
	KSA	15.407	7	12
	HCA	14.052	8	14
12-bit	RCA	24.963	14	24
	KSA	18.497	25	43
	HCA	17.070	22	39
24-bit	RCA	44.355	28	48
	KSA	27.921	79	139
	HCA	20.196	59	104

Table 7. Comparison of various vedic multipliers [20]

COMPARISON OF COMBINATIONAL DELAY WITH DIFFERENT ADDER ARCHITECTURE

Device Name	Multiplier	Adder Type and Combinational delay in ns		
		RCA	CLA	KS
Spartan II Xc2S200pq 208-5	4 bit	23.439	24.508	10.334
	8 bit	42.479	42.820	40.176
	16 bit	80.688	78.665	69.648
Virtex2 Xc2V250F G256-5	4 bit	12.964	14.553	13.281
	8 bit	24.442	24.677	22.090
	16 bit	45.227	41.447	37.184
Virtex5 Xc5VLX22 0-2ff1760	4 bit	06.712	06.828	07.718
	8 bit	12.214	10.864	05.640
	16 bit	19.201	20.478	18.995

2.2 COMPARISION SUMMARY OF BOOTH MULTIPLIERS

It is obvious from theory that the modified booth multipliers with high radix are more efficient than conventional booth multipliers. It is obvious from table 9 & 10 that increase in area and delay is not that much high as the efficiency. If we compare delay

and area both simultaneously then booth with Radix 4 is the best multipliers among the booth series of multipliers.

Table 8. Comparison between booth radix2 & radix 4 [6]

Device Utilization Summary	Radix 2	Radix 4
Number of Slices	397	71
Number of 4 input LUTs	184	100
Number of bonded INPUT	16	16
Number of bonded OUTPUT	16	16
Macro Statistics		
# Latches	24	12
8-bit latch	24	12
# Xors	71	23
1-bit xor2	64	21
8-bit xor2	7	2
Timing Summary		
Minimum period:	5.454ns	4.750ns
Minimum input arrival time before clock	7.936ns	4.014ns
Maximum output required time after clock:	6.216ns	6.205ns

Table 9. Area comparison [28]

S.N0	BOOTH MULTIPLIER	NO. OF CELLS	TOTAL AREA
1	RADIX 2	9426	27569
2	RADIX 4	1690	3959
3	RADIX8	2280	4359
4	RADIX16	14672	32973
5	RADIX32	25703	57091

Table 10. Delay comparison [28]

S.NO	BOOTH MULTIPLIER	TIME DELAY(ns)
1	RADIX 2	23.382
2	RADIX 4	13.993
3	RADIX8	14.810
4	RADIX16	26.581
5	RADIX32	27.729

2.3 COMPARISION SUMMARY OF WALLACE-TREE MULTIPLIERS

From tables, it is clear that the modified booth wallace is better than the conventional wallace tree multiplier. But booth-recoded wallace tree multiplier is the most optimized multiplier. Modified wallace multiplier is better than the traditional wallace multiplier in terms of complexity and delay.

Table 11. Complexity of reduction[25]

Input Size (N)	8	16	24	32	64
Stages (S)	4	6	7	8	10
Wallace					
Full adders	38	200	488	96	3850
Half adders	15	52	100	156	430
Tatal gates	402	2008	4801	8778	36388
Modified Wallace					
Full adders	39	201	490	907	3853
Half adders	3	9	16	23	53
Tatal gates	363	1845	4474	8263	34889

Table 12. Comparison number of transistor and latency[24]

Wallace Multiplier Architectures	Number of Transistors (N)	Latency (L)
Conventional Wallace Multiplier	2998	27
Efficient High speed Wallace Tree Multiplier	2748	15

Table 13. comparison of area, delay and power for wallace-tree multiplier of different sizes [23]

Size	Delay (ns)				Area (μm^2)				Power (mW)			
	TW	RCW	Dadda	PW	TW	RCW	Dadda	PW	TW	RCW	Dadda	PW
8	2.81	2.64	2.64	2.66	3392	3346	3262	3148	1.92	2.04	1.94	1.96
16	4.13	3.65	3.80	3.75	14847	14372	14242	13876	11.09	11.41	11.22	11.20
24	4.64	4.58	4.62	4.44	34337	33479	33323	32352	27.69	28.24	28.32	28.04
32	14.83	14.62	14.82	14.62	61526	59271	59086	58375	51.85	52.74	53.11	52.48
64	22.88	21.57	21.98	21.62	246842	238843	238597	237553	216.50	219.17	222.64	218.92

Table 14. Delay comparison [25]

Type	Width (bit)	Delay (ns)
Wallace	8	7.168
Vedic	16	13.452
Booth radix 8	32	12.081
Booth radix 8	32	11.564
FPGA(XC6vlx75tl)	32	11.238
Booth radix 8	32	9.536

2.4 COMPARISION OF DADDA & BRAUN MULTIPLIERS

Table 16 and table 17 show that partitioned dadda multiplier is better than the regular multiplier if implemented in higher order bits. For example, 64x64 partitioned multiplier is better in terms of area and delay. But it is not power efficient multiplier. Braun multiplier using Kogge stone adder (KSA) is more fast than that by using ripple carry adder (RCA).

Table .15. Delay for braun multipliers [32]

S.No.	FPGA	Braun Multiplier using R C A (ns)	Braun Multiplier using K S A (ns)
1	Spartan 2(xc2s 15-6cs144)	16.019	15.929
2	Spartan 2(xc2s 200-5fg 56)	18.446	18.346
3	Spartan 2E(xc2s 100e-7ft256)	14.251	14.161

Table 16. Regular Dadda mul. With CLA [10]

Multiplier NxN	**Area(um^2)**	**Delay(ns)**	**Power(uW)**
8x8	8.428	3.40	6.32
16x61	29.169	4.71	33.09
32x32	105.237	5.92	210.50
64x64	397.146	7.54	925.92

Table 17. Partitioned Dadda mul. With CLA [10]

Multiplier NxN	**Area(um^2)**	**Delay(ns)**	**Power(uW)**
8x8	8.957	3.51	6.85
16x16	30.241	4.61	35.22
32x32	107.362	5.47	218.76
64x64	386.629	6.94	952.59

III. PHASE- 2 (INTER-SERIES COMPARISON)

On the basis of comparison summaries, it is obvious that there are six better multipliers out of total 23 multipliers discussed here. Those are listed as: among vedic series (i) vedic multiplier using han carlson adder is the best, (ii) vedic multiplier using sklansky adder is also a better one. Among the booth series multipliers (i) booth radix 4 than other radix multipliers. Among wallace-tree multipliers (i) wallace-booth and (ii) modified wallace multipliers are better. Now there is a need of inter-series comparison for the analysis of relative performance of these multipliers. Now here are some tables to do this comparison.

Table 18. 24x24 multipliers delay, area and power comparison [8],

Parameter	Array Multiplier	Booth Mul. Radix 4	Vedic Multiplier	Wallace tree Multiplier
Delay(ns)	19.1	6.35	8.0	10.6
Area	96582	80196	71946	103847
Power dissipation(uW)	18.692	38.529	20.305	32.932
AD(10^{-3})	1.845	0.509	0.575	1.101
PD(10^{-12})	357.0172	244.66	162.44	349.08
Power Ddnsity(10^{-7})	1.935	4.804	2.822	3.171

Table 19. Comparison of 12x12 multipliers [2]

TYPE	DELAY (ns)	AREA	POWER (uW)	Area-Delay product(10^3)
ARRAY MUL	7.106	807	23.98	5.734
BOOTH radix 2	4.89	1175	38.264	5.745
BOOTH radix 4	4.669	772	26.22	3.604
WALLACE	4.860	956	29.209	4.646
BOOTH WALLACE	4.665	859	28.830	4.007

Table 20. Comparison 24x24 multipliers [9]

Parameter	Array Multiplier	Booth Mul. Radix 4	Vedic Multiplier	Wallace tree Multiplier
Delay(ns)	18.8	11.3	12.5	13.7
Area	87707	46982	55999	70962
Power dissipation(uW)	15.878	12.386	10.299	15.035
AD(10^{-3})	1.649	0.531	0.700	0.972
PD(10^{-12})	298.5064	139.9618	128.7375	205.9795
Power Ddnsity(10^{-7})	1.810	2.636	1.840	2.119

The generalized comparison of various multipliers is shown in the table no. 21.. The lowest delay is of booth-wallace multiplier. The least areas are of modified booth and vedic multipliers. And the lowest power is of vedic multiplier. However modified booth is also a power efficient multiplier.

Table 21. Comparison table for conclusion

TYPE	DELAY	AREA	POWER
ARRAY MUL	highest	large	highest
MODIFIED BOOTH	medium	less	low
VEDIC MUL	medium	less	lowest
WALLACE	high	large	high
BOOTH WALLACE	low	largest	high

IV. CONCLUSION

From the analysis of all multipliers, it is obvious that modified booth radix4 and modified wallace-booth multipliers are the best multipliers. In some cases, vedic and dadda mutipliers(in terms of speed) are also superior to these but by overall analysis booth and wallace tree can be easily modified to obtain a best optimised multiplier. This comparison helps us to select a suitable multiplier for a particular task or application. For future research, a better multiplier can be proposed by modifying these multipliers and a better comparison of various multipliers with the proposed modified multiplier can be done by taking all multipliers simultaneously in one table. That will give more accurate comparison in percentage values.

REFERENCES

[1] Savita Nair & Ajit Saraf, "A Review Paper on Comparison of Multipliers based on Performance Parameters," International Conference on Advances in Science and Technology (ICAST-2014).

[2] Jasbir Kaur, Sumit K, "Analysis and Comparison of Different Multiplier," International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-816975-78. Volume: 4 Issue: 6. June 2016.

[3] K.Harika, "Analysis of Different Multiplication Algorithms & FPGA Implementation," IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), Volume 4, Issue 2, Ver. I (Mar-Apr. 2014), PP 29-35, e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197.

[4] Saji. M. Antony, "Design of High Speed Vedic Multiplier using Multiplexer based Adder," International Conference on Control, Communication & Computing India (ICCC) 119-21 November 2015.

[5] Mr. M. Prakash, "SIMULATION AND COMPARATIVE ANALYSIS OF DIFFERENT TYPES OF MULTIPLIERS", IJAICT Volume 1, Issue 7, November 2014.

[6] Sukhmeet Kaur, "Implementation of Modified Booth Algorithm (Radix 4) and its Comparison with Booth Algorithm (Radix-2)", Advance in Electronic and Electric Engineering. ISSN 2231-1297, Volume 3, Number 6 (2013), pp. 683-690.

[7] Kavita and Jasbir K., "Design and Implementation of an Efficient Modified Booth Multiplier using VHDL," International Journal of Advances in Engineering Sciences Vol.3 (3), July, 2013 e-ISSN: 2231-0347 Print-ISSN: 2231-2013, ICETEM 2013.

[8] Bhavesh S. and Ruchika M., "COMPARISON OF SINGLE PRECISION FLOATING POINT MULTIPLIER USING DIFFERENT MULTIPLIER ALGORITHMS," IRF-International Conference Organized by Institute of Research and Journals in association with: COE, Bharti Vidyapeeth, Deemed University, Pune.

[9] Bhavesh Sharma , Amit Bakshi, "Comparison of 24X24 Bit Multipliers for Various Performance Parameters," 1st International Conference on Advent Trends in Engineering, Science and Technology "ICATEST 2015", 08 March 2015.

[10] B. Ramkumar, V. Sreedeep and Harish M Kittur, "A Design Technique for Faster Dadda Multiplier", Integrated Circuit Design Laboratories, VIT University, Vellore, India.

[11] Kavitha Priya N, and Karthikeyan KV, "Analysis of Mac Unit Using Vedic Multiplier and Sklansky Adder", Research Journal of Pharmaceutical, Biological and Chemical Sciences, ISSN: 0975-8585, May – June 2016.

[12] Addanki Purna Ramesh, "Implementation of Dadda and Array Multiplier Architectures Using Tanner Tool," IJCSET, ISSN : 2229-3345 Vol. 2 No. 2.

[13] Parul Agrawal, Rahul Sinha, "Comparative Analysis and FPGA Implementation of Vedic Multiplier for various Bit Lengths using Different Adders," IJIRCCE (An ISO 3297: 2007 Certified Organization), Vol. 3, Issue 10, October 2015, ISSN(Online): 2320-9801, ISSN (Print): 2320-9798.

[14] C. Saranya, Dr. M. Balasubramaniam, Dr. R. Naveen, "Performance Analysis of Various Multipliers for Digital Signal Processing Applications," IJARCCE, ISO 3297:2007 Certified, Vol. 5, Issue 9, September 2016. ISSN (Online) 2278-1021, ISSN (Print) 2319 5940.

[15] Kavita, Umesh Goyal, "Performance Analysis of Various Vedic Techniques for Multiplication," International Journal of Engineering Trends and Technology-Volume4 Issue3- 2013, ISSN: 2231-5381.

[16] Vishal Galphat1, Nitin Lonbale, "The High Speed Multiplier by using Prefix Adder with MUX and Vedic Multiplication," International Journal of Science and Research (IJSR), ISSN (Online): 2319-7064, Volume 5 Issue 1, January 2016.

[17] Sudeep.M.C, "Design and FPGA Implementation of High Speed Vedic Multiplier," International Journal of Computer Applications (0975 – 8887), Volume 90 – No 16, March 2014.

[18] Mahadevappa Mugalihal, Dr. T. C. Thanuja, "Design and Implementation of an Efficient 32×32 Bit Vedic Multiplier Using Kogge Stone Adder," International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 6, June 2016 ISSN: 2277 128X.

[19] Sakshi Rajput, Priya Sharma, " High Speed and Reduced Power –Radix-2 Booth Multiplier," IJCEM International Journal of Computational Engineering & Management, Vol. 16 Issue 2, March 2013. ISSN (Online): 2230-7893.

[20] N.G.Nirmal, Dr. D.T.Ingole, "Novel Delay Efficient Approach for Vedic Multiplier with Generic Adder Module," IJERA, ISSN: 2248-9622. Vol. 3, Issue 3, May-Jun 2013, pp.1394-1396.

[21] Gijin V George and Anoop Thomas, "High Performance Vedic Multiplier Using Han-Carlson Adder," IJERT, ISSN: 2278-0181, Vol. 3 Issue 3, March – 2014.

[22] Swathi A.C, Yuvraj T, "A Proposed Wallace Tree Multiplier Using Full Adder and Half Adder," IJIREEICE, Vol. 4, Issue 5, May 2016, ISSN (Online) 2321 – 2004, ISSN (Print) 2321 – 5526.

[23] Shahzad Asif and Yinan Kong, "Low-Area Wallace Multiplier," Hindawi Publishing Corporation, VLSI Design, Volume 2014, Article ID 343960, http://dx.doi.org/10.1155/2014/343960

[24] Hussain, R. K. Sah, "Performance Comparison of Wallace Multiplier Architectures," International Journal of Innovative Research in Science, Engineering and Technology, (An ISO 3297: 2007 Certified Organization. Vol. 4, Issue 1, January 2015. ISSN(Online): 2319 – 8753. ISSN (Print) :2347 – 6710.

[25] Himanshu Bansal, K. G. Sharma, "Wallace Tree Multiplier Designs: A Performance Comparison Review," Innovative Systems Design and Engineering, ISSN 2222-1727 (Paper) ISSN 2222-2871,Vol.5, No.5, 2014.

[26] Elisardo Antelo and Paolo Montuschi, "Improved 64-bit Radix-16 Booth Multiplier Based on Partial Product Array Height Reduction," IEEE Transactions on Circuits and Systems I: Regular Papers (Volume: 64, Issue: 2, Feb. 2017).

[27] Minu Thomas, "Design and Simulation of Radix-8 Booth Encoder Multiplier for Signed and Unsigned Numbers," IJIRST–International Journal for Innovative Research in Science & Technology| Vol. 1, Issue 1, June 2014| ISSN(online): 2349-6010.

[28] JasbirKaur, Sumit Kumar, "PERFORMANCE COMPARISON OF HIGHER RADIX BOOTH MULTIPLIER USING 45nm TECHNOLOGY," IJIRSET (An ISO 3297: 2007 Certified Organization), Vol. 5, Issue 1, Januray 2016, ISSN(Online): 2319-8753, ISSN (Print): 2347-6710.

[29] Hyunpil Kim, Sangook, "Radix 16 booth multiplier using noval weighted 2-stage booth algorithm," IEICE Electronics express, vol. 11, no. 13, June 2014.

[30] S.Shenbagam, V.Vandhana, "PERFORMANCE OF DADDA MULTIPLIER USING CARRY SELECT ADDER," ICSTM-16, May 2016, ISBN: 978-81-932074-8-2.

[31] Mohammad Faizan Khan Q., Ram Racksha Tripathi," Binary Multiplier Using Modified Radix-4 Booth Algorithm", International Journal of Emerging Technology and Advanced Engineering, (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 6, Issue 4, April 2016.

[32] Ms. Madhu Thakur, Prof. Javed Ashraf," Design of Braun Multiplier with Kogge Stone Adder & It's Implementation on FPGA", International Journal of Scientific & Engineering Research, Volume 3, Issue 10, October-2012 1, ISSN 2229-5518.