iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** V **Month of publication:** May 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Design of High speed Vedic MAC Unit using Urdhva Tiryakbhyam sutra & comparison with Conventional Architecture

Arpita A. Koli[1], S.B.Kulkarni[2]
[1]M.Tech- VLSI Deign and Embedded Systems, [2]Associate Prof & H.O.D)
K.L.E. Dr.M.S .Sheshgiri College of Engineering & Technology, Belgavi, India

Abstract— A major trend in the semiconductor design industry is to continually produce increasingly faster processors. For high speed Processors the main processing bottlenecks is the multiply and Accumulate unit (MAC). The speed of MAC depends greatly on the multiplier. The speed of multiplier greatly depends on the number of multiplication and adder units. In this paper we have proposed an improved time and area efficient MAC unit which employs Vedic multiplier. The Vedic mathematics going to reduce the number of adder and multiplier as compare to the conventional multipliers. So we propose high-speed, efficient area MAC adopting Vedic multiplication sutra based architecture for 2bit, 4bit, 8 bit, 16 bit, 32 bit and 64bit size and compared with present conventional architectures. The proposed MAC unit is coded in Verilog, synthesized and simulated using Xilinx ISE10.1.
Index Terms—MAC, Vedic Multiplier, Ripple Carry (RC), Adder, Carry save Adder (CSA)

I. INTRODUCTION

A design of high speed 64 bit Vedic Multiplier-and-Accumulator (MAC) unit is implemented using high performance Vedic multiplier and compares its delay with present MAC units. The key component of MAC unit is Multiplier that multiplies two n-bit numbers X and Y and gives a product 2n bits wide. This is added to or subtracted from the contents of the accumulator in the add/sub unit. The result is saved in the accumulator. The MAC unit is designed using Vedic multiplier and Ripple carry adder hence, compared the performance of MAC unit with Braun, Array, Wallace multiplier and carry save adder. The MAC inputs are obtained from the memory location and given to the multiplier block. A system's performance is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. Furthermore, it is generally the most area consuming. However, area and speed are usually conflicting constraints so that improving speed results mostly in larger areas. As a result, whole spectrums of multipliers with different area-speed constraints have been designed with fully parallel. Hence, with the suitable choice of the type of the multiplier the performance of the MAC unit can be made better. Hence, in the construction of the MAC unit multiplier plays a vital role so, we have selected those multipliers which exhibit better performance than the previous one implemented in the MAC units. In this paper, we proposed a MAC unit consisting of adder and accumulator in a same block. By this proposed method by building both adder and the accumulator in the same block the delay can be decreased and other better performance can be seen. When the input is given to the multiplier it starts computing value for the given 64 bit input and hence the output will be 128 bits. The multiplier output is given as the input to adder and accumulator block shown below in figure 1. The output adder and accumulator block is 129 bit i.e. one bit is for the carry (128bits+ 1 bit). Then, the output is fed back to the same adder and accumulator block. The figure 1 shows the new architecture of MAC unit.

II. MAC OPERATION

As we know, MAC unit mainly consist of multiplier, adder and accumulator. In this paper, we proposed a MAC unit consisting of adder and accumulator in a same block. By this proposed method by building both adder and the accumulator in the same block the delay can be decreased and other better performance can be seen. This will be useful in 64 bit digital signal processor. The

input which is being fed from the memory location is 64 bit. When the input is given to the multiplier it starts computing value for the given 64 bit input and hence the output will be 128 bits. The multiplier output is given as the input to adder and accumulator block shown below in figure 1. The function of the MAC unit is given by the function of the MAC unit is given by the following equation [1]:

$$F=\Sigma Pi\ Qi \qquad (1)$$

The output adder and accumulator block is 129 bit i.e. one bit is for the carry (128bits+ 1 bit). Then, the output is fed back to the same adder and accumulator block. The figure 1 shows the new architecture of MAC unit.

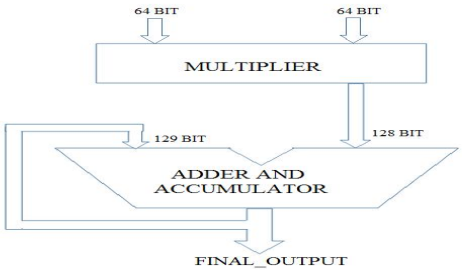


Figure 1: MAC Block diagram

III. VEDIC MULTIPLIER

The hardware architecture of 2X2, 4x4, 8x8, 16x16, 32x32 and 64x64 bit Vedic multiplier module are displayed in the below sections. Here, "Urdhva-Tiryakbhyam" (Vertically and Crosswise) sutra is used to propose such architecture for the multiplication of two binary numbers. The beauty of Vedic multiplier is that here partial product generation and additions are done concurrently. Hence, it is well adapted to parallel processing. The feature makes it more attractive for binary multiplications. This in turn reduces delay, which is the primary motivation behind this work.

A. Vedic Multiplier for 2x2 bit Module

The method is explained below for two, 2 bit numbers A and B where A = a1a0 and B = b1b0 as shown in Fig. 2. Firstly, the least significant bits are multiplied which gives the least significant bit of the final product (vertical). Then, the LSB of the multiplicand is multiplied with the next higher bit of the multiplier and added with, the product of LSB of multiplier and next higher bit of the multiplicand (crosswise). The sum gives second bit of the final product and the carry is added with the partial product obtained by multiplying the most significant bits to give the sum and carry. The sum is the third Corresponding bit and carry becomes the fourth bit Of the final product [3]. Figure 2 shows the block diagram of 2x2 bit Vedic Multiplier which is further used for the implementation of the 4x4 bit Vedic multiplier and further 8x8 Vedic multiplier, 16x16 bit, 32x32bit and 64x64bit Vedic multiplier is implemented.

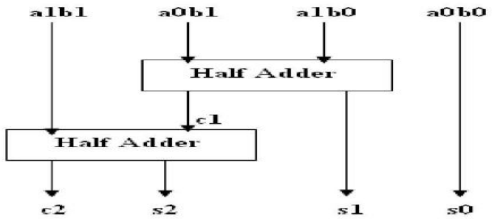


Figure 2[10]:Block Diagram of 2x2 bit Vedic

Figure 2: Block Diagram of 2x2 bit Vedic Multiplier

B. Vedic Multiplier for 4x4 bit Module

The proposed Vedic multiplier can be used to reduce delay. Early literature speaks about Vedic multipliers based on array multiplier structures. On the other hand, we proposed a new architecture, which is efficient in terms of speed. The arrangements of RC Adders shown in Fig. 3, helps us to reduce delay. Interestingly, 4x4 Vedic multiplier module is implemented easily by using four 2x2 Vedic multiplier modules. The outputs of 2x2 bit multipliers are added accordingly to obtain the final product. Here total two 6 bit and one 4 bit Ripple-Carry Adders are required as shown in Fig.3.

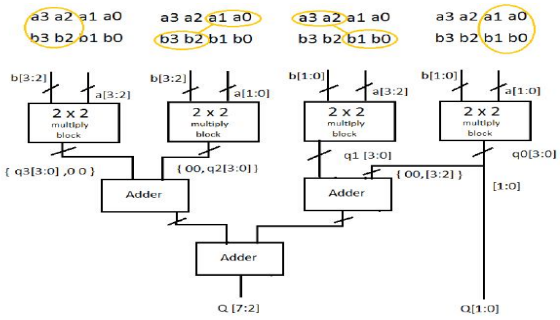


Figure 3: Block Diagram of 4x4 bit Vedic Multiplier.

C. Vedic Multiplier for 8x8 bit Module

The 8x8 bit Vedic multiplier module as shown in the block diagram in Fig. 3 can be easily implemented by using four 4x4 bit Vedic multiplier modules as discussed in the previous section Let's analyze 8x8 multiplications, say A= A7 A6 A5 A4 A3 A2 A1 A0 and B= B7 B6 B5B4 B3 B2 B1B0. The output line for the multiplication result will be of 16 bits as – S15 S14 S13 S12 S11 S10 S9 S8 S7 S6 S5 S4 S3 S2 S1 S0. Let's divide A and B into two parts, say the 8 bit multiplicand A can be decomposed into pair of 4 bits AH-AL. Similarly multiplicand B can be decomposed into BH-BL. The 16 bit product can be written as: Using the fundamental of Vedic multiplication, taking four bits at a time and using 4 bit multiplier block as discussed we can perform the multiplication. The outputs of 4x4 bit multipliers are added accordingly to obtain the final product. Here total two 12 bit and one 8 bit Ripple-Carry Adders are required as shown in Fig.4.

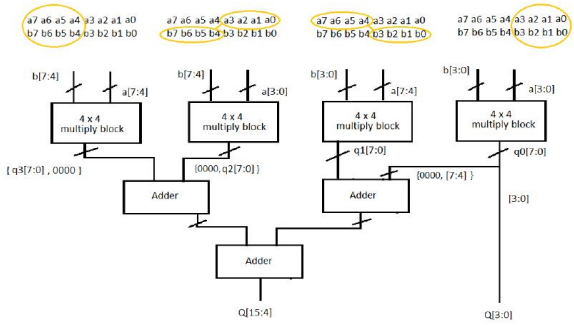


Figure 4: Block Diagram of 8x8 bit Vedic Multiplier.

D. Vedic Multiplier for 16x16 bit Module

16x16 Vedic multiplier module is implemented easily by using four 8x8 Vedic multiplier modules. The outputs of 8x8 bit multipliers are added accordingly to obtain the final product. Here total two 24 bit and one 16 bit Ripple-Carry Adders are required as shown in Fig.5.

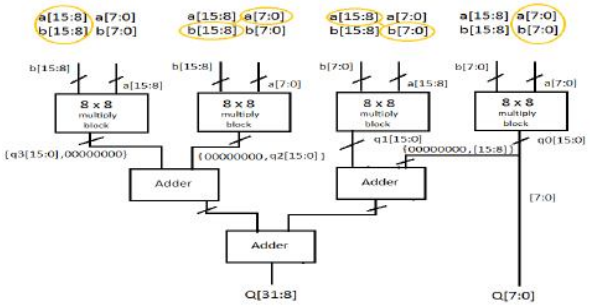


Figure 5: Block Diagram of 16x16 bit Vedic Multiplier.

E. Vedic Multiplier for 32x32 bit Module

32x32 Vedic multiplier module is implemented easily by using four 16x16 Vedic multiplier modules. The outputs of 16x16 bit multipliers are added accordingly to obtain the final product. Here total two 48 bit and one 32 bit Ripple-Carry Adders are required as shown in Fig.6.

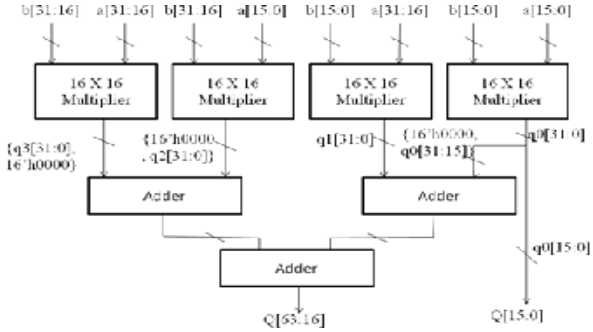


Figure 6: Block Diagram of 32x32 bit Vedic Multiplier.

F. Vedic Multiplier for 64x64 bit Module

64x64 Vedic multiplier module is implemented easily by using four 32x32 Vedic multiplier modules. The outputs of 32x32 bit multipliers are added accordingly to obtain the final product. Here total two 95 bit and one 64 bit Ripple-Carry Adders are required as shown in Fig.7.

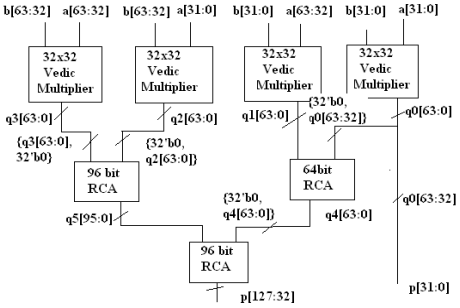


Figure 7: Block Diagram of 64x64 bit Vedic Multiplier.

IV. ARRAY MULTIPLIER

A. Step A: n ×n Partial product generation using array of AND gates by multiplying x0.y0, x0.y1, …, up to xn−1. yn−1 in parallel at the same time. n = 4 in the example of multiplying 0b1011 with 0b0l01.

B. Step B: Use adders to add the partial products at the n-levels. Note that each level m partial product x xm.y0, xm.y1, …, up to x1. yn–1 is shifted to the left one position to the left to account for the differing place values of the bits in the second input.

C. Step C: Generating final result using two-bit operand adders

D. Designing Array Multiplier

Total Number of logic units in n-bit × m bit Array Multiplier
n ×m two-input ANDs and (m – 1) units of n-bit adders

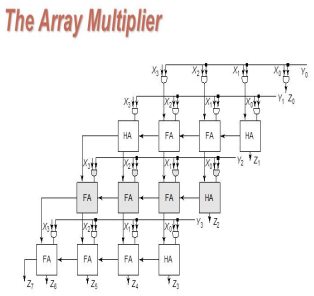


Figure 8: Block Diagram of Array Multiplier

V. BRAUN MULTIPLIER

It is a simple parallel multiplier generally called as carry save array multiplier. It has been restricted to perform signed bits. The structure consists of array of AND gates and adders arranged in the iterative manner and no need of logic registers. This can be called as non – addictive multipliers.

Architecture:

An n*n bit Braun multiplier [9] & [10] is constructed with n (n-1) adders and n2 AND gates as shown in the fig.1,

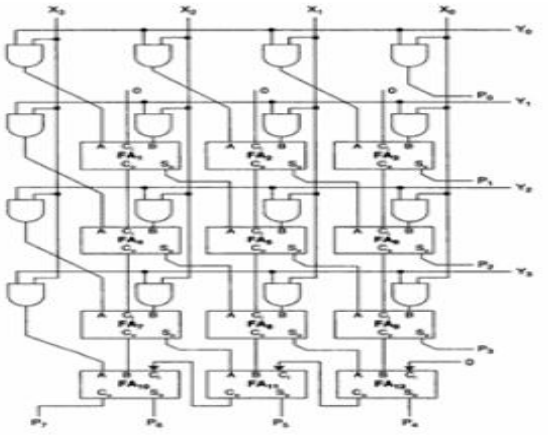


Figure 9: Block Diagram of Braun Multiplier

VI. WALLACE MULTIPLIER

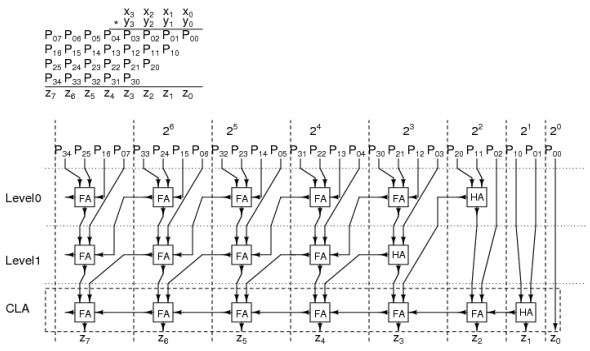


Figure 10: Block Diagram of Wallace Multiplier

VII. RIPPLE CARRY ADDER

A full adder is a combinational circuit that performs the arithmetic sum of three input bits: augends Ai, addend Bi and carry in Cin from the previous adder. Its results contain the sum Si and the carry out, Cout to the next stage.

The Boolean equations of a full adder are given by:

Sout= ABC + AB'C' + A'B'C + BA'C'

= (AB'+BA') C +AB+A'B') C'

Sout= A⊕B⊕C

Cout= AB + AC + BC

Cout= AB + C (A⊕B)

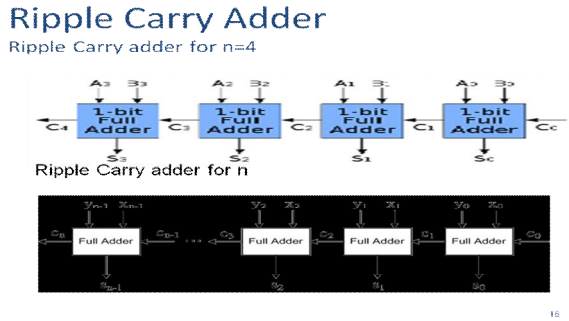


Figure 11: Block Diagram of Ripple Carry Adder

VIII. CARRY SAVE ADDER

The carry-save unit consists of n full adders each of which computes a single sum and carry bit based solely on the corresponding bits of the three input numbers. Given the three n - bit numbers a, b and c it produces a partial sum PS and a shift-carry SC.

PSi = ai^bi^ci (5)

SCi = (ai^bi) | (ai^ci) | (bi^ci)

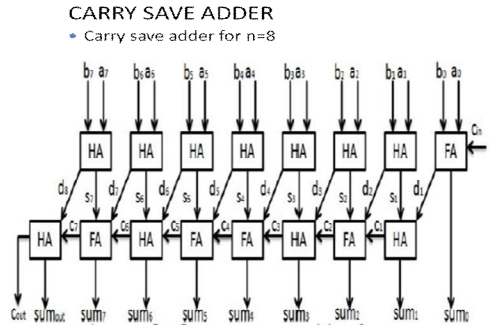


Figure 12: Block Diagram of Carry Save Adder

IX. RESULTS

The design is done using Verilog-HDL by using tool Xilinx ISE 10.1i and target family Spartan 3E,Device- XC3S100,speed -5,package: FG320.

TABLE I. COMBINATIONAL DELAY OF PROPOSED VEDIC MAC UNIT

N-bit MAC unit	Logic Delay (ns)	Route Delay (ns)	Total Delay (ns)
64bit	300.416	123.361	423.778
32bit	151.488	67.182	218.670
16bit	77.024	33.863	110.887
8bit	42.119	18.153	60.272
4bit	20.930	4.946	25.876
2bit	8.837	3.816	12.653

TABLE II: AREA OF PROPOSED VEDIC MAC UNIT

N-bit MAC Unit	Number of Logic cell usage
64bit	13557
32bit	3772
16bit	1098
8bit	312
4bit	89
2bit	21

TABLE III: COMPARISON OF AREA

Multiplier unit	2x2-bit	4x4-bit	8x8-bit
Vedic multiplier	4-M 2-A	4-M 12-A	4-M 32-A
Array multiplier	4-M 2-A	15-M 8-A	64-M 53-A
Braun multiplier	4-M 2-A	15-M 12-A	64-M 56-A
Wallace multiplier	4-M 2-A	26-M 16-A	

No .of Additions- A &
Multiplications-M in multiplier

In the Table 6, 'M' stands for multiplications used in the respective width of multipliers and 'A' stands for additions used in the respective width of multipliers. From this Table 6 it is clear that how the Vedic mathematics going to reduce the number of adder and multiplier as compare to the conventional multipliers.

TABLE IV: COMPARISON OF DEVICE UTILIZATION

S.No	MAC Unit	No. of Slices	No. of 4 input LUTs	No. of IOs	No. of bonded IOBs
1.	Vedic 4x4 MAC Unit	27 out of 960 2%	49 out of 1920 2%	18	17 out of 66 25%
2.	Array4x4 MAC Unit	27out of 960 2%	48 out of 1920 2%	19	18 out of 66 25%
3.	Braun 4x4 MAC Unit	25 out of 960 2%	43 out of 1920 2%	18	17 out of 66 25%
4.	Wallace 4x4 MAC Unit	32 out of 960 3%	57 out of 1920 2%	32	21 out of 66 31%

TABLE V: COMPARISON OF COMBINATIONAL DELAY WITH VARIOUS MULTIPLIER AND MAC UNIT

S.No	Multipliers	Logic Delay	Route Delay	Total Delay
1.	Vedic4x4	8.866	4.173	13.039
2.	Array4x4	9.171	4.389	13.560
3.	Braun4x4	7.947	3.724	11.671
4.	Wallace4x4	9.171	4.388	13.559
S.No	Adders			
1.	RCA8X8	9.035	3.676	12.711
2.	CSA8X8	10.122	5.345	15.467
S.No	MAC Unit			
1.	Vedic4x4 MAC Unit	20.930	4.946	25.876
2.	Braun4x4 MAC Unit	21.176	7.952	29.128
3.	Array4x4 MAC Unit	23.926	9.310	32.448
4.	Wallace 4x4 MAC Unit	21.176	8.039	29.215

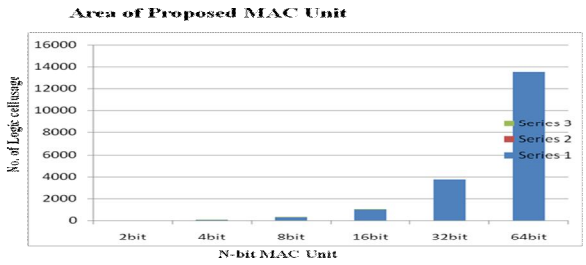


Figure 13: Area of Proposed MAC Unit

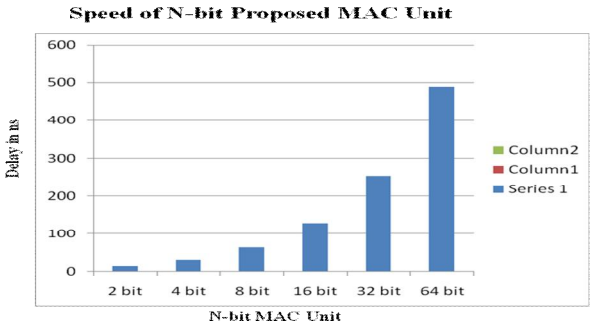


Figure 14: Speed of N-bit Proposed MAC Unit

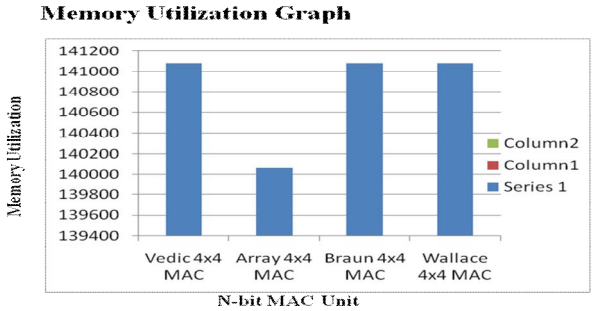


Figure 15: Memory Utilization Graph

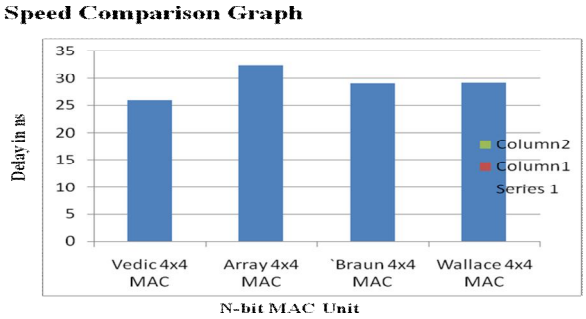


Figure 16: Speed Comparison Graph

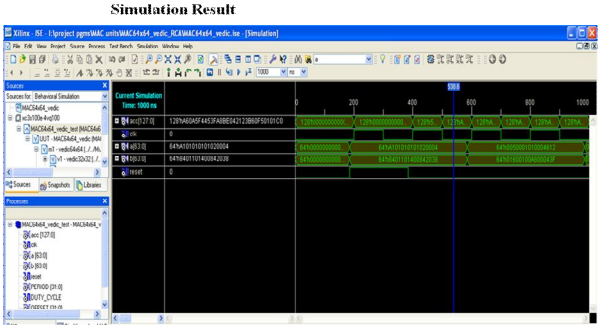


Figure 17: Simulation Result

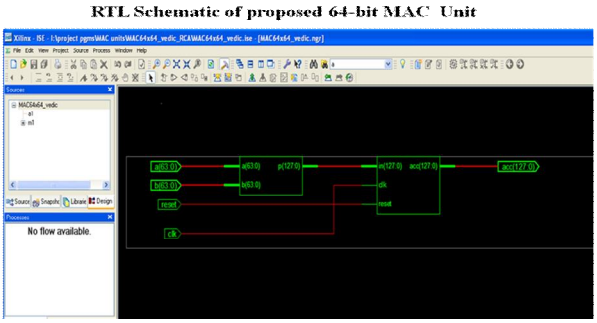


Figure 18: RTL Schematic of Proposed 64-bit MAC Unit

REFERENCES

[1] Vaijyanath Kunchigi1, Linganagouda Kulkarni2, Subhash Kulkarni3"32 Bit MAC Unit Using Vedic Multiplier" International Journal of Scientific and Research Publications, Volume 3, Issue 2, February 2013 1[ISSN 2250-3153]

[2] Ranjan Kumar Barik1, Sourav Kumar Dwibedi2, ShasankaSekhar Rout3, SatyaRanjan Sahu4" An Improved and Efficient MAC Unit and its Implementation in FPGA" International Journal of Review in Electronics & Communication Engineering (IJRECE) Volume 2 - Issue 2 April 2014

[3] Mr. Sumit C. Katkar, Prof. Pragati KeneProf. Shubhangini Ugale" DESIGN OF EFFICIENT 64 BIT MAC UNIT USING VEDIC MULTIPLIER FOR DSP APPLICATION – A REVIEW" IJAICT Volume 1, Issue 9, January 2015

[4]Shaik.Masthan Sharif1 ,D.Y.V.Prasad2" Design of Optimized 64 Bit MAC Unit for DSP Applications" International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 456- 460 (2014) Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh.[ISSN 2278-3091]

[5] Y.Narasimha Rao, GSVP Raju, Penmetsa V KrishnaRaja," Design and Performance Evaluation of High Speed MAC Unit with Parallel Pipeline Technology" International Journal of Computer Applications (0975 – 8887) Volume 106 – No.4, November 2014

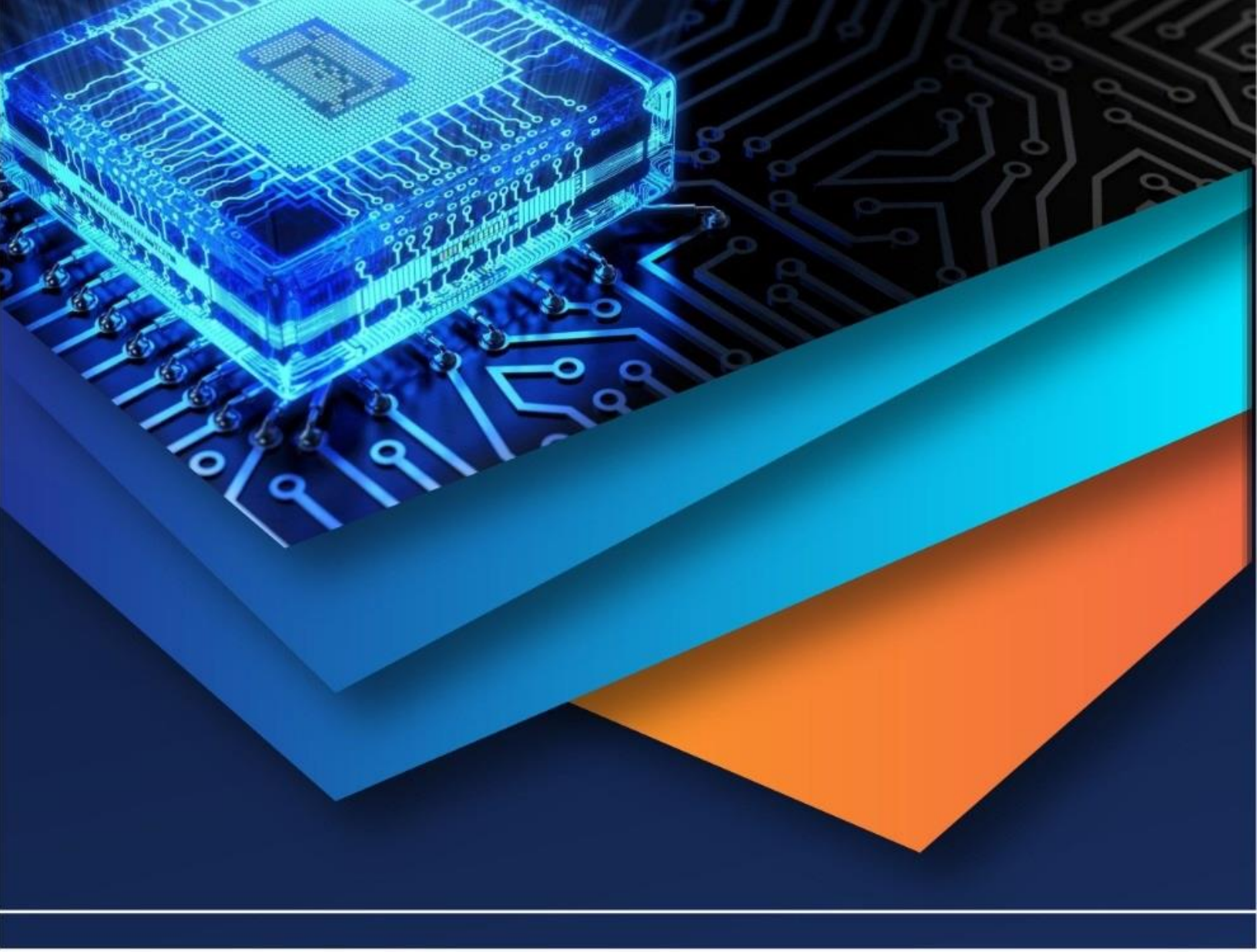
[6] P.jagadeesh"Design of High Performance 64 bit MAC Unit" 2013 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013].

[7] W.J. Townsend, E.E. Swartzlander Jr., and J.A. Abraham, "A Comparison of Dadda and Wallace Multiplier Delays, "Proc. SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, pp. 552-560, 2003.

[8] M.H. Rais, M.H. Al Mijalli, "Braun's multipliers: Spartan-3AN based design and implementation", J. Comput. Sci., vo. 7, no. (11), pp.1629-1632, 2011.

[9] C.S. Wallace, "A Suggestion for a Fast Multiplier," IEEE Trans.Electronic Computers, vol. 13, no. 1, pp. 14-17, Feb. 1964.

[10] Avatar singh, S.Srinivasan "Digital Signal Processing" Text Book India Edition.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY