International Journal of Electrical Engineering and Technology (IJEET)
Volume 12, Issue 5, May 2021, pp. 179-188, Article ID: IJEET_12_05_017
Available online at https://iaeme.com/Home/issue/IJEET?Volume=12&Issue=5
ISSN Print: 0976-6545 and ISSN Online: 0976-6553
DOI: 10.34218/IJEET.12.5.2021.017

© **IAEME** Publication **Scopus** Indexed

DESIGN OF LOW POWER MAC USING YAVADUNAM SUTRA

Tushar Deora, Tanya Singhania, Rajeev Pankaj Nelapati*
Department of Communication Engineering, School of Electronics Engineering,
Vellore Institute of Technology, VIT University, Vellore, Tamil Nadu, India
*Corresponding Author

ABSTRACT

In our day to day circumstances we come across various mathematical challenges. Vedic mathematics can solve any algorithm in a less sophisticated way. Vedic mathematics is an old approach to fathom issues in a rapid manner. This paper proposes design of area efficient and low power 64 bit MAC unit which is made up of multiplier architecture based on Yavadunam sutra of Vedic mathematics and high performance Adder. The proposed MAC unit also offers critical enhancement in speed. The proposed architecture is designed using the principles of Yavadunam Sutra. The Verilog codes for the proposed Yavadunam multipliers were simulated using Mentor Graphic Tool (ModelSim SE 10.5). The proposed architecture of MAC is synthesized using the Synopsys Design Compiler with SAED 32 nm technology. The results show that the reduction in area and power is because of the proposed 32 bit multiplier implemented in MAC and consequently its efficiency increases.

Key words: vedic mathematics, MAC, Verilog, Vedic Multiplier; Yavadunam sutra

Cite this Article: Tushar Deora, Tanya Singhania, Rajeev Pankaj Nelapati, Design of Low Power MAC using Yavadunam Sutra, *International Journal of Electrical Engineering and Technology (IJEET)*, 12(5), 2021, pp. 179-188.
https://iaeme.com/Home/issue/IJEET?Volume=12&Issue=5

1. INTRODUCTION

We are living in time where there is always a need of fast computing digital signals processors which can compute multiplication and addition in fast and effective manner. To meet the requirement of fast and efficient Digital signal processor, which can perform real time Digital signal processing like FFT, convolution, filters, addition and multiplication or ALU (Arithmetic Logic Unit) of Microprocessor, for such purpose we need MAC unit which is able to multiply two numbers and add to accumulator. As MAC unit includes multiplication and addition, hence if we can improve our multiplier and also work on Adder we can meet with our desired output of low power and high efficiency MAC unit.

In the proposed work, we worked on multiplier and adder which is used in MAC unit. Multiplier is improved musing Vedic mathematics called Yavadunam sutra and compared with

MAC unit which has Array multiplier, to see which is more efficient and less time consuming and as expected multiplier with Yavadunam sutra algorithm of Vedic mathematics was more efficient. When compared to advanced mathematics, this Vedic mathematics makes the calculation basic and simple. With the help of Vedic multiplier and Ripple Adder, MAC is implemented which has less area and power consumption. The verilog coding is used for the proposed Yavadaunam multipliers which were simulated using Mentor Graphic Tool (ModelSim SE 10.5). Architectures were synthesized using the Synopsys Design.

2. RELATED WORK

Sharmaet al. [1] presents the design of high speed multiplier and squaring architectures based upon ancient Indian Vedic Mathematics sutras, all the partial products are adjusted using concatenation operation and are added using single carry save adder instead of two adders at different stages. Wang et al [2] proposed a multiplier circuit based on the Add and Shift algorithm. Although this proposed array multiplier is easy to design, it is very slow because of its long critical path and it requires less area. Arish et al [3] in his paper proposed a finest algorithm for binary multiplication in terms of area and delay. But for higher order bits the area increases with increase in number of bits. Kunchigi et al. [4] proposed low power square and cube architectures using Vedic sutras. The Ekadhikena Purvena Sutra is used for squaring, is limited to number which ends with digit 5 only. The other method Dwandwa Yoga or Duplex is used in two different senses. The first one is by squaring and the second one is by cross multiplication. Multiplications are very expensive and slow the overall operation. The performance of many computational problems is often dominated by the speed at which a multiplication operation can be executed. In general multipliers are classified into the Serial and the Parallel multiplier. The parallel multipliers are preferred over the serial multipliers due to the fact that the parallel multipliers don’t make use of combinational circuit and does not have feedback unit [5].The parallel multipliers are classified into array and tree multipliers. Tree multipliers are more beneficial than the array multipliers. Wallace Tree multipliers expend less power and are of high speed, despite the fact that they have abnormal structure [6], [8], [12], [11]. After all the literature survey on multipliers, we did our research on MAC unit. In many digital signal processing applications MAC unit is an inevitable component. This unit is mainly used as it reduces the load of CPU by working independently [7]. In this paper, The work presents a systematic design methodology for fast and area efficient digital multiplier based on Vedic mathematics and then a MAC unit has been made which uses this multiplier [9].As we know MAC unit supports large number of digital signal processing applications within limited number of cycles as per application demand. Like some of the filters can adjust a delay during execution but the algorithms such as orthogonal transforms etc.demands accurate speed execution / measurement which sometimes can be beyond the processor capability [10].

3. VEDIC MULTIPLIER – YAVADUNAM

This segment conveys the principles and algorithm of Yavadunam sutra followed by some examples. We can multiply any no. using Yavadunam Sutra whether it is in decimal or binary. Let’s take example of two decimal numbers using the Yavadunam sutra.

Take an example of 104 and 112.

So X=104 and Y =112

The important observations are:

- Take the Nearest base value of 10^N which is 100 in this example.
- Find the deficiency by subtracting the number from its base value.

So,

$D_1 = X - 10^N = 104\text{-}100= 4$ $D_2 = Y - 10^N = 112\text{-}100=12$

- Now take LHS = $(X+Y) - 10^N$ + Carry
- Thus, LHS= (104 + 112) -100 + 0 =116
- And RHS is taken by multiplying both the deficiency $D_1{*}D_2$. Thus, RHS = 4*12 = 48
- Now RHS= LHS&RHS

Which is 11648 in this case.

Thus, Answer of multiplying 104 and 112 is 11648. This method is proved successful only for the numbers close to base values. Hence this multiplier has the objective for the achievement of effective area and power. The following section conveys the proposed Vedic multiplier using Yavadunam principles.

4. BINARY VEDIC MULTIPLIER

As we have done multiplication in Yavadunam sutra we can also do it in binary but for binary inputs, base value will be in power of 2 (i.e. 2^N), where N is no. of bits in the input. So using this method we can multiply any number of any range in quickly. Here in this case, deficiency is computed by taking two's compliment of two input taken in binary. Based on the input values there are three modes of operation. Let's take two binary numbers be X and Y. If both X and Y is greater than 2^{N-1} then it is Mode 1. If both X and Y are less than 2^{N-1} then it is considered Mode 2. If one of the inputs is greater and one of the inputs is less (i.e. $X>2^{N-1}$ and $Y<2^{N-1}$ or $X<2^{N-1}$ and $Y> 2^{N-1}$) then it is considered as mixed called Mode3. Hence, there are three modes – mode1, mode2 and mode3. The Algorithm for Yavadunam multiplier if inputs are binary contains N bit multiplier and Algorithm is as follows.

4.1. Algorithm

Let's take two binary digits be X and Y.

Step 1: Find the deficiency considered as D1and D2 which is computed by taking the two's complement of X and Y respectively.

Step 2: Now we have to multiply both theDeficiencies D1and D2Using the N bit multiplier.

Step 3: The RHS of product XY is derived by taking the least N bits of product of deficiencies D1and D2.

Step 4: MSB N bits of the multiplier is taken from the RHS and it is further added deficiencies that we derived in Step 1.

Step 5: The LHS of the product is computed by adding the deficiencies D1 and D2. Depending on the value of the input theadder output is taken as such or two's complement.

Step 6: The carry of the RHS is added. The sign of theadder changes based on the given inputs.

If both the inputs are greater than 2^{N-1},

Then LHS = $[2^N – (D1+D2)]$ + carry bits from RHS

If both the inputs are lesser than 2N-1,

then LHS = (D1+D2) - carry bits from RHS.

If both the inputs are mixed,

then LHS = $[2^N – (D1+D2)]$ + carry bits from RHS.

The above proposed algorithm is implemented in the proposed multiplier architecture. Our proposed multiplier is 32 bit and it works efficiently based on the above algorithm. The implemented 32 bit multiplier architecture is shown in the Figure 1.

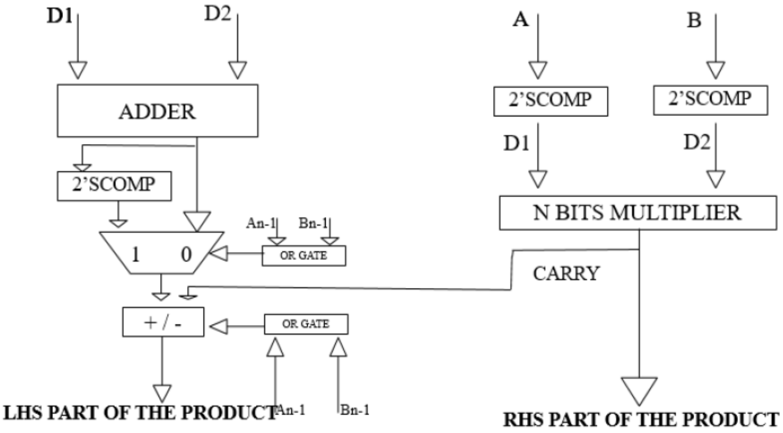


Figure 1 Architecture of Binary Multiplier

Let's take example of multiplication of two binary numbers.

Say, X= 1001 and Y=0100

The important observations are:-

- Let's take N bit multiplier be 4.
- Base value is 2^N here it is 10000.
- If 2^N is 10000 then 2^{N-1} is 1000.
- Here in this case X > 2^{N-1} and Y< 2^{N-1}, hence it is mixed Mode or Mode 3.
- Now the deficiencies of X and Y are found using two's complement so D1=111 and D2 =1100.
- Now we have to add both the deficiencies D1 + D2 =10011
- And multiplication of both the deficiencies be D1 *D2=1010100.
- LHS= 2^N – (D1 + D2) + carry = 10000 –(10011) +101=10
- RHS is simply multiplication of deficiencies D1*D2=1010100, in this case 101 becomes carry which is added in LHS.
- Now we have to concatenate LHS and RHS to get the result 100100.

Thus answer of two binary no X= 1001 and Y=0100 when multiplied using Yavadunam sutra is 10010.

5. RIPPLE CARRY ADDER

In the proposed paper we have used Ripple carry adder to add the number to the accumulator. Ripple carry adder is simple logic circuit which is basically made up of full adder and has carry and sum as its output. It is called ripple carry adder because each carry is getting rippled into next stage.

These types of adder are made especially for Digital Signal Processing and microprocessor. If the working of ripple carry adder is seen properly, so we can see that it has to wait for its previous stage full adder carry-in. and n^th^ full adder has to wait unit (n-1) complete their operation.

Ripple-carry adders are quite simple and easily expandable to any desired width. However, they are rather slow, because carries may propagate across the full width of the adder. This happens, for example, when the two 8-bit numbers 10101011 and 01010101 are added. Because each Full Adder A requires some time to generate its carry output, cascading *k* such units together implies *k* times as much signal delay in the worst case. A variety of fast adders can be designed that require logarithmic, rather than linear, time. In other words, the delay of such fast adders grows as the logarithm of *k*.

Ripple carry adder is made of full adder and the schematic and the truth table of full adder is shown in the

Figure 2 and Figure 3 respectively.

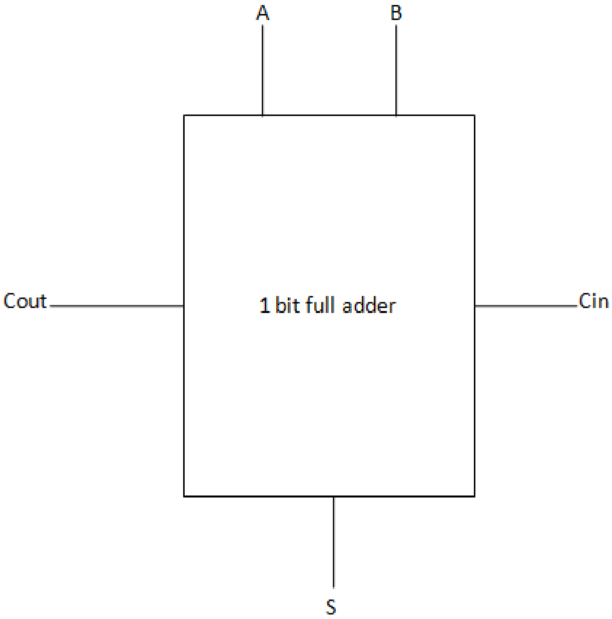
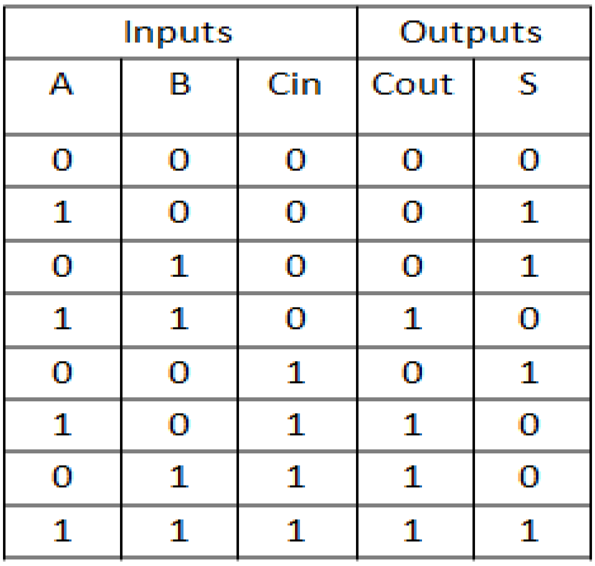


Figure 2 Schematic of Full Adder



Inputs			Outputs	
A	B	Cin	Cout	S
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
1	1	0	1	0
0	0	1	0	1
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1

Figure 3 Truth Table of 1 bit Full Adder

4. PROPOSED MAC UNIT

Accumulate (MAC) Unit. MAC Unit is a major module found in pretty much every processor accessible today. As compared to Booth multiplier, the proposed multiplier or multiplier made using Yavadunam sutra is fast and efficient, which makes MAC unit faster. So MAC unit contains Arithmetic modules.

MAC unit multiplies two numbers, which is multiplied using multiplier, which is given as input and adds using adder or accumulates using accumulator in its previous value. To understand MAC unit let’s take an example of two no. be X and Y.

Let X= 2 and Y=4 and Accumulator (A) =0. Initially two number will be multiplied so 2*4 =8 and it will be added to Accumulator which becomes A=0+8=8. Now let’s say we take another input X=4 and Y=5, so they will be multiplied 4*5 =20 and will be added to previous accumulator value, A becomes A= 8+20=28. If again inputs are, let’s say X=3 and Y=2, it will be multiplied with multiplier to get 3*2=6, so A=6+28=34, process repeats like this.

The architecture of Multiplier is such in a way that it consists Vedic multiplier followed by ripple carry adder and D flip flop is shown in figure 4.

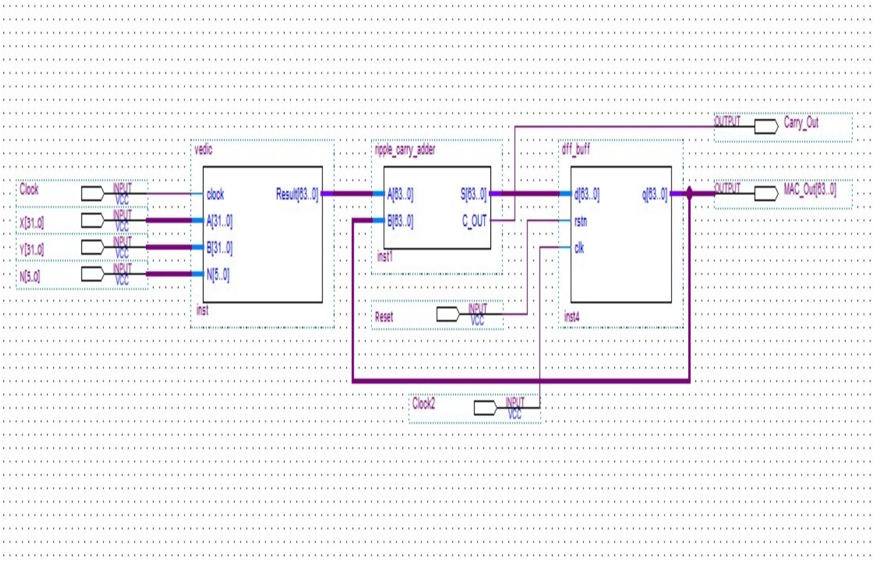


Figure 4 Proposed architecture of MAC UNIT

7. RESULTS AND DEICUSSION

The Verilog codes for the proposed MAC unit using Yavadunam multiplier was simulated using Mentor Graphics tool (ModelSim SE 10.5). And the simulated output waveform of proposed MAC unit with Yavadunam multipliers generated which is shown in the figure 5

The proposed architecturesare synthesized using the Synopsys Design Compiler with SAED 32nm technology. The area, delay, power of the proposed multiplier architecture is shown in table 1. The power, area and delay of the proposed MAC unit is compared with the conventional MAC unit.

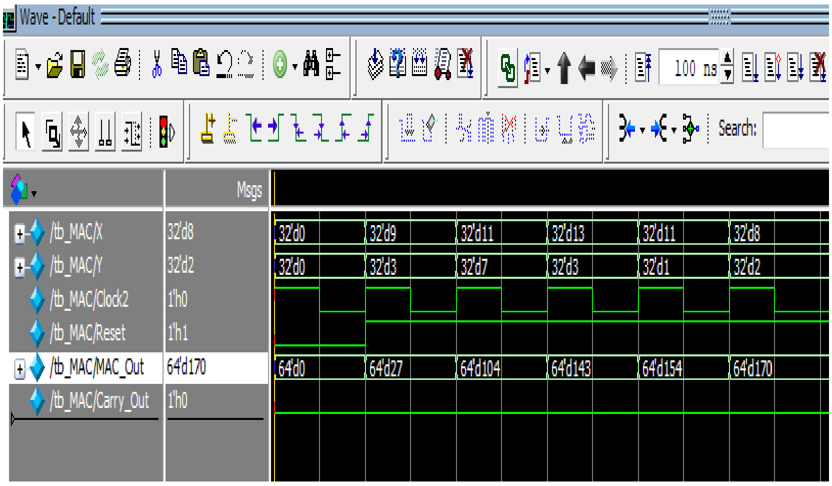


Figure 5 Simulation output of proposed MAC unit

Table 1 Experimental Results of Area, Power and delay

	MAC with proposed multiplier	**Conventional MAC**
Area	9998.787377	14949.512523
Power	2.7521e+03 uW	3.2266e+03 uW
Delay	4.83 ns	3.85 ns

As shown in table one we can clearly see that Area of MAC with Yavadunam multiplier is 9998.787377 units and area of MAC without Yavadunam multiplier is 14949.512523 units.

And the power of MAC with Yavadunam multiplier is 2.7521e+03 Uw and power of MAC without Yavadunam multiplier is 3.2266e+03 uW. And the delay of MAC with Yavadunam multiplier is 4.83 nanoseconds and delay of MAC without Yavadunam sutra is 3.85 nano seconds.

Table2 shows the results of internal power, switching power, leakage power and total power for the 64 bit proposed MAC and conventional MAC.

Table 2. The detailed power comparison of the proposed MAC and Conventional MAC.

64 bit	**32nm Technology**			
	Internal Power (uW)	**Switching Power (uW)**	**Leakage Power (uW)**	**Total Power (uW)**
MAC without Yavadunam multiplier	1.2459 e+03	689.9293	1.2908 e+09	3.2266e+03
MAC with Yavadunam multiplier	1.2591 e+03	454.6496	1.0383 e+09	2.7521e+03

The following section shows the graphical analysis of Area, power and delay for the MAC with Yavadunam multiplier and MAC without Yavadunam multiplier.

The graphical representation of the detailed power analysis of both MAC units for 32 nm technology is shown in figure 6.

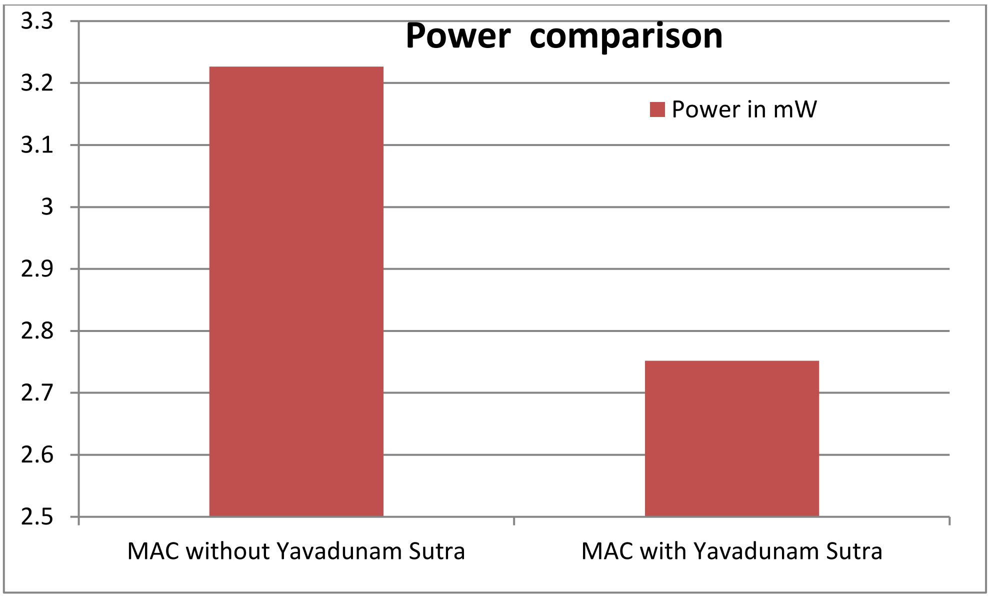


Figure 6 Power Comparison between the two MAC

The graphical representation of the detailed area analysis of both MAC units for 32 nm technology is shown in figure 7.

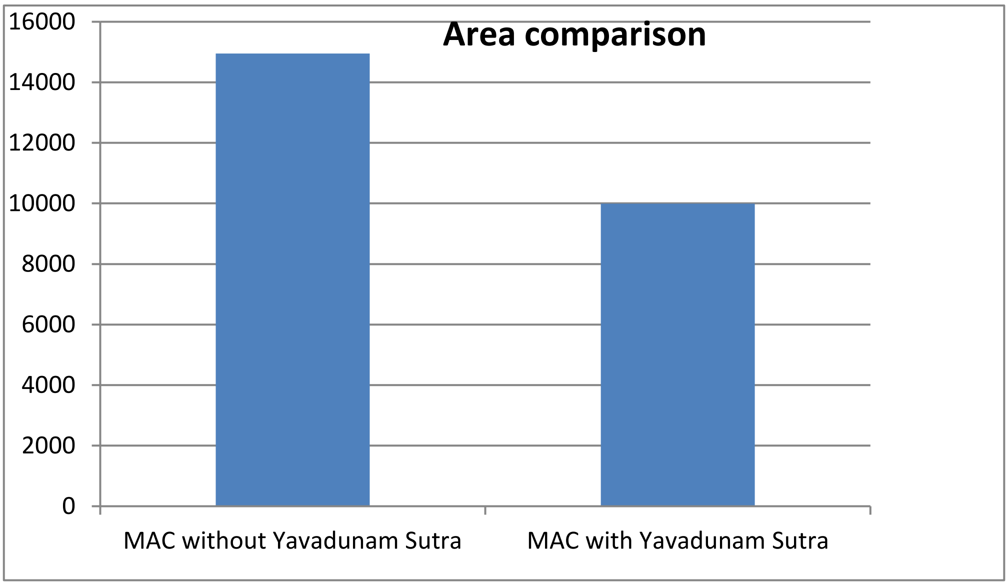


Figure 7 Area Comparison between the two MAC

The graphical representation of the detailed delay analysis of both MAC units for 32 nm technology is shown in figure 8.

The performance analysis report of 32 bit Yavadunam multiplier MAC unit synthesized in 32 nm technology reveals that the area of the proposed Yavadunam multiplier based MAC unit is 33.116% more efficient than conventional multiplier based MAC unit. Similarly, the power of the proposed Yavadunam multiplier based MAC unit is 14.705% more efficient than conventional multiplier based MAC unit. But, our proposed MAC unit is 25.454% less efficient than conventional MAC unit in terms of delay which could be improved in future.

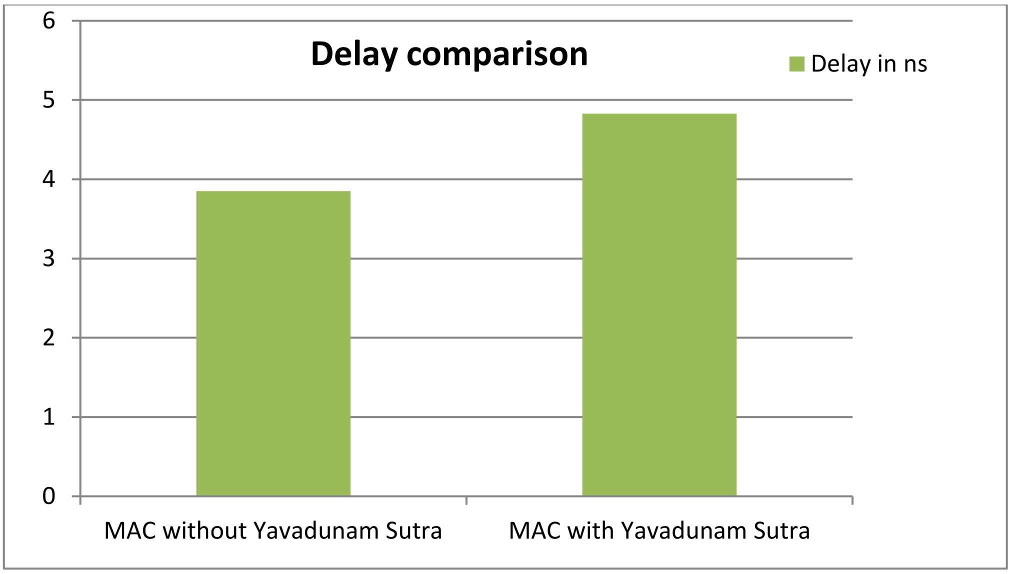


Figure 8 Delay Comparison between the two MAC

Hence, we see that our proposed MAC architecture is more area and power efficient when compared to conventional MAC structure, and the proposed MAC unit can be used in various DSP applications for better and fast results.

8. CONCLUSION

The problem of designing an efficient MAC unit is resolved. A complete unique architecture is designed which would work for any set of inputs or say any deficiencies whether positive, negative or mixed. The proposed architecture is precise and gives accurate end result for any type of inputs. As compared with the prevailing techniques, during literature survey, the speed of the proposed system is improved. The speed of the proposed MAC unit can further be increased if an efficient adder like prefix parallel adder is used instead of ripple carry adder. This research primarily focused on multiplier and thus, 64 bit Kogge Stone adder part is a scope of improvement, also delay can also be improved in the proposed MAC unit.

REFERENCES

[1] Sharma, R., Kaur, M., & Singh, G. (2015). Design and FPGA implementation optimized 32-bit Vedic multiplier and squarearchitectures. In *International conference on industrial instrumentation and control* (ICIC), May 28-30.

[2] Wang J S, Kuo C N and Yang T H 2004 Low Power fixed Width array multipliers. In *:Proceedings of international sadhana (2019) 44:197 Page 9 of 10 197 symposium on low power electronics and designpp. 307-312.*

[3] Arish S and Sharma R K 2015 An efficient binary multiplier design for high speed application using Karratsubaalgorithm and Urdhava-Tiryagbhyam algorithm. In*: Proceedings of the 2015 global conference on communication technologies.*

[4] Kuchingi, V., Kulkarni, l., & Kulkarni, S. (2014). Low power squareand cubearchitectures using Vedic sutras. In *Fifth international conference on signals and image processing.*

[5] PriyankaBrahmaiah V, Dharma Teja L and Dr. Y.PadmaSai 2013 Study On Comparison Of Various Multipliers International Journal Of Elctronics and Communication Engineering And Technology 4132-142.

[6] Abhilash R, Raju IBK, Chary G and Dubey S 2015 Area-Power Efficient Vedic Multiplier Using Compressors International Conference On Electrical, Electronics, Signals, Comm. Optim.1-5.

[7] P.Jagadeesh, S.Ravi, Dr. Kittur Harish Mallikarjun, “Design of High Performance 64 Bit MAC Unit”, International Conference on Circuits, Power and Computing Technologies (ICCPCT), PP 782-786,

[8] Harish Kumar Ch 2013 Implementation and Analysis of Power, Areaand delay of Array, Urdhva, Nikilam Vedic Multipliers Int. J. of Scientific and Research Publications 3 1-5.

[9] Manoranjan Pradhan, rutuparna Panda, Sushanta Kumar Sahu,”Speed Comparison of 16x16 Vedic Multipliers”, International Journal of Computer Applications, Volume 21-No6, May 2011.

[10] HardwareMultiply/Accumulate (MAC) Unit, Motorola

[11] Vidyashankari P and Lokesha B 2015 Design and Implementation of Squareand CubeArchitectures using Vedic Sutras on FPGA International Journal of Emerging Technology in Computer Scienceand Electronics (IJETCSE) 14 377-381.

[12] SiridharaA L, Mahendra V and Ravinder T 2017 and Electronics (IJETCSE) 14 377-38.