DESIGN OF HIGH SPEED APPROXIMATE MULTIPLIER USING ADDER COMPRESSORS

[1]G.Siva Prasad, [2]L.Viharika, [3]M.Divya, [4]R.Nagaraju, [5]K.Seetaramaraju
[1, 2, 3,4U] G Scholar, [5]Assistant Professor
[1]Electronics and Communication Engineering
1Godavari Institute of Engineering and Technology, Rajahmundry, Andhra Pradesh

Abstract: In this modern era, many of the digital systems are error resilient which allows us to take the advantage of approximate computations. This makes the use of replacement of exact computing units by their counterparts. Approximate computing can also decrease the complexity at the designing levels with an increase in performance and power efficiency. Adders and multipliers are the basic buildings blocks of many digital applications. These blocks can be approximated in several ways. Research works are on the rise at many levels on approximate computing. Approximation at designing level is more advantageous as the modifications at this level much easier than the preceding levels. A method of designing an approximate multiplier with a novel structure introduced in 16-bit adder compressor is proposed. The 16-bit adder compressor (AC) is designed with 8-2 adder compressors in general. The 8-2 adder compressor is designed with 7-2 and 3-2 adder compressors and half adders. The existing and proposed multiplier is designed using Xilinx 14.7 in the frontend. The speed of proposed multiplier 55.44% increase compare to Existing Multiplier.

Index Terms **– Approximate multipliers, 16-Bit AC, AC's.**

I. INTRODUCTION

In the field of multimedia signal processing, there is an acceptance of the small magnitude of errors. For this area, there is no particular demand for exact processing so there is a place for approximate value. To make the most of computational error tolerance various techniques are accessible. These are of three types. (1) Insistent voltage scaling, (2) Truncation of bit-width, (3) Use of imprecise building blocks. The main necessity for approximate computing was to decrease the hardware implementation as well as aim to diminish the power consumption. The main components involved in multimedia processing are adder as well as a multiplier. These two components have a huge role in the processing of media. Multiplication is nothing but the repeated addition of partial products. Generally, these adders are utilized to achieve the multiplier partial product. In order to diminish the multipliers hardware complexity, truncation is widely used to design fixed width multiplier. In this method broken array multipliers are introduced in order to implement the partial product as well as truncation for least significant bits, which can also degrade the power consumption. In the proposed method for faster processing and at the same time to diminish the error at the output sum, a novel structure in the 4-2 compressor is used.

A compressor is implemented using two full adders in general. The existing technique employed 4-2 compressors that are utilized for partial product reduction involved in 8x8multiplier. There are certain disadvantages concerned in this type of compressor since it delivers non-zero outputs in the place of zero-valued inputs that will impact the corresponding mean relative error. To overcome the disadvantages involved in the multiplier, different full adders that contain a fixed number of transistors are brought to bear. In current years it is witnessed that there is high demand for low power consuming electronics for better battery life as well as it is common practice for every system which requires less complexity, faster processing, as well as the reliable hardware design.

The main aspiration of this is to conquer the disadvantages that will lead to a better approximation of results. In static segment multiplier (SSM) there are m-bits (i.e., m-bit segment) taken from each n-bit operand. An m-bit segment can start only from one of two or three fixed bit positions subject to the paramount one bit is fixed for a positive number. This is used for high accuracy. Mainly for handheld devices, it is a necessary to design an architecture that has low power consumption since it is powered by batteries. So raise in the complexity of the circuit may engender a requirement for high power which will lead to disastrous implementation. The basic challenge is to design the reliable computation circuit that has not only low power consumption but also should implement in the least area.

II. LITERATURE SURVEY

A detailed presentation of the existing design work that is required for the proposed multiplier implementation. It covers the necessity of minimization of power, since power efficient implementation is a very challenging task. At the same time, to achieve the high computational speed, existing work covers the minimization of power consumption by diminishing the switched capacitance that confers one by one as follows:

In case of the sequential multiplier as well as the combinational multiplier performance, the multiplication mainly contains the generation of partial product in addition to diminishment in the partial product. Subsequently, it follows the carry propagation addition. Considering the combinational case due to realization involves less area.

The Author **Karthikeya Bharadwaj et.al** reported in his work multiplier design by utilizing the Wallace tree for the purpose of minimizing the power in addition to the area. The author employs the idea of 'carry-in prediction' in order to diminish the critical path. He also stated by utilizing the Wallace tree we easily optimize the design for the area as well as power. His simulation result achieves the accuracy of 99.86% to 99.967%. Later on, he was executed on the applications in real time on benchmark images. Finally, he observed and concluded that there is a huge diminishment in the power. At the same time, there is a huge reduction in the area need to implement a multiplier. The author also noticed that there is no loss of image quality.

The Author **Weiqiang Liu et.al** delivered in his work about the realization of approximate computing by utilizing the Radix-4 booth multiplier that has more prominence in the application of image processing. The main goal of his approximate computation is to achieve low power in addition to the high performance. He designed the booth multiplier by making the use of radix-4 modified booth encoding commonly known as MBE algorithm. The author designed approximate partial product by utilizing the approximate Wallace tree. He has finally concluded that his 16-bit radix-4 booth multiplier has a better approximation as well as more precise than the existing booth multiplier. The author achieved all these key merits with less propagation delay in addition to minimization of power consumption.

The Author **Shiksha Bathla** represented in his work about the high-speed 4-bit multiplier with low power consumption. He has proposed the four-bit multiplier by utilizing FinFET technology. Shiksha Bathla has done four-bit multiplier work by means of HSPICE tool. Since when compared to metal oxide semiconductor technology (MOSFET) FinFET utilize less power. Furthermore, he was successfully delivered optimization delay term in FinFET based multiplier by mean of sleep mode in addition to the tri-mode technique.

K. Benarji Srinivas et.al. delivered in his work about the multiplier that consumes less power in addition to the high-speed performance. One of the key characteristic merit of his work is diminishing the switching activities in order to diminish the power dissipation, since row and column multiplier diminishes the switching actions. The author utilized the tri-state buffer in order to be in command of gating element. Furthermore, with the utilization of tri-statebuffer, there is a wide diminishment in signal propagation delay. The author also stated that multiplier that he has contributed, consumes less power of about 20% as well as row & column bypass multiplier well suited for some major applications like filtering.

Suryasnata Tripathy et.al. proposed low power multiplier architecture by utilizing the key concept of Vedic mathematics and this architecture is implemented by mean of 45 Nanometer technology in order to achieve faster computing performance. The author implemented the four bit as well as the eight-bit multiplier by using sutra of urdhvatiryakbhyam.

Callaway et.al.(2006) represented in his work about the reduction in transition action for a digital multiplier. The author further demonstrated the switching activity that has been involved in a partial product with a diminishment in the hardware structure. For high-speed multiplier in common practice can be called as the multiplier that uses superfluous binary adder tree. There are mainly two steps involved in the fixed point multiplication. The first step involves the generation of partial product. Subsequently follows the accumulation of partial product. In order to achieve the faster multiplication, it is necessary to follow the standard ways. First, one produces the less number of partial products by pick up the pace in accumulation in subsequent steps.

D. Jakuline Moni et.al. has made some significant points in his paper about the design of low power booth multiplier that can perform sixteen-bit multiplication operation or else two eight-bit multiplication operation. Furthermore, the product obtained at the output was truncated in order to diminish the power consumption. Subsequently, it will raise the speed of processing in compensation need to surrender the precision of the output. Since this multiplier perform at high speed because it omitting the switching operation of the ineffective range of values.

Koyel Dey et.al. represent the unique procedure for designing the high performance eight-bit multiplier by utilizing the Vedic mathematics. The author implemented the key procedure by adopting the sixteen nanometer technology. Later on the author compared the existing multiplier performance with the multiple channel complementary metal oxide semiconductors (McCMOS) technology in addition to 65 nanometer technology. Furthermore, the author carried the key analysis by mean of T-spice simulation. Finally, he was concluded that multiplier by mean of 16 nanometer technology consume less power when judge against on other technologies.

Jinpeng Xing in his work presented high performance multiplier based FPGA. The author adopted about the new coding algorithm that helped in designing the reliable 24x24 bit multiplier with low power dissipation. Moreover, it not only reduces the computational steps required to generate partial product but also diminishes the number of adders required for the multiplier. The author successfully delivered the 24x24 bit multiplier based on FPGA with the diminishment of power consumption about 8.4 % when compared to existing algorithm.

III. EXISTING WALLACE TREE MULTIPLIER

The growing demand for high speed signal processing, raise the necessity to design the digital system with enhancement in the performance. Some of the digital applications like multimedia processing, it is not essential of exact precision. So therefore approximate processing is acceptable that will further not only decrease the system complexity but also diminishes the implementation area. In this work advanced multiplier is redesigned with petite modifications in order to diminish the system complexity.

In this approach, multiplier partial product is changed to initiate the unreliable probability. In this advance approximation design is done by employing two variant multiplier of 16-bit. Furthermore, the synthesis outcome expose that this two multiplier attain saving of power by seventy-two and thirty-eight percentages respectively. At the same time, it is showing the better performance when we make comparison with the present offered multipliers. Furthermore, it shows the superior precision having the mean relative error is 7.5% and 0.021% for this advanced approximate multiplier. The computation steps involved in the generation of partial product mainly utilize the computational gates like half adder and full adder.

The execution of multiplier mainly consist of three key steps they are partial product generation, diminish tree of partial product and consequently, the vector integrate summation to generate the ending product from the output of sum as well as rows that are produced from the reduction tree. In consequent procedure consumes additional power. So therefore in this scenario, approximation is implemented in the reduction tree with the intention to reduce computational steps needed to process the product is shown in Fig.1.

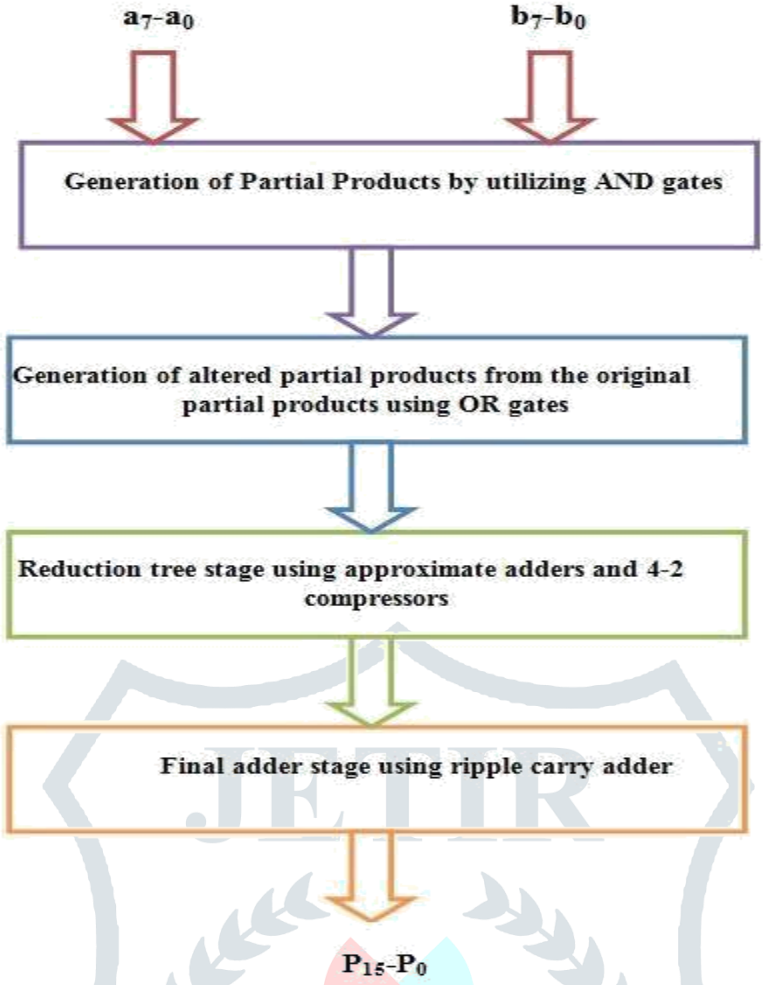

Fig. 1 block diagram of existing multiplier

Let us consider the 8-bit unsigned multiplier to demonstrate the approximate multiplier proposed work. Let the operands of 8-bit unsigned multiplier be = $\sum^{7}_{=0}$ 2 and = $\sum^{7}_{=0}$ 2 then the partial product is given by .

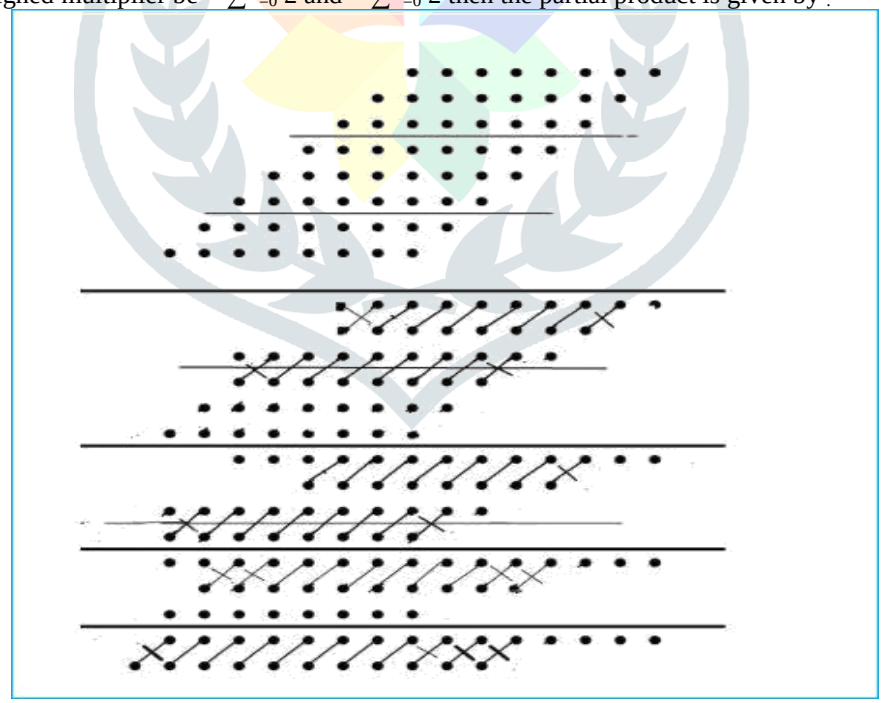

Fig. 2 Structure of conventional multiplier

Obtained partial products are then combined to get "propagate" and "generate" signals pm, n and gm, n. Conversion of partial products into propagate and generate signals is shown in fig. 3. Partial product that has been generated in further process consequently, we will calculate the corresponding probability. For the generated signal the probability statistics is depicted in table 1.

Table 1 Generated signal Probability info

M	Probability of the generate elements being				P_{err}
	All zero	One 1	Two 1's	Three 1's And more	
2	0.8789	0.1172	0.0039	-	0.00390
3	0.8240	0.1648	0.0110	0.00024	0.01124
4	0.7725	0.2060	0.0206	0.00093	0.02153

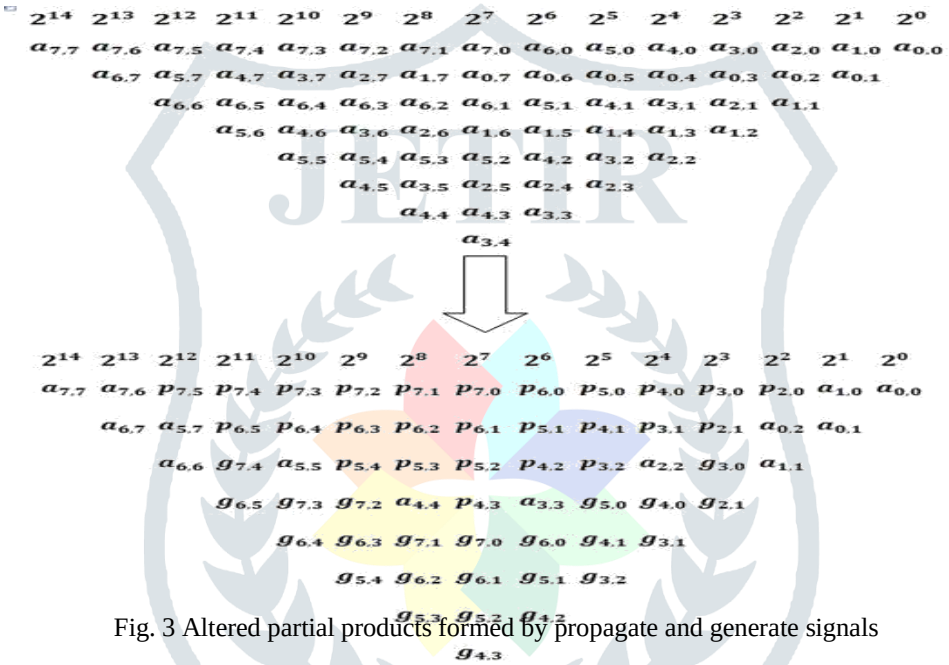

Fig. 3 Altered partial products formed by propagate and generate signals

Considering the probability theory point of view, for the generation of partial product , has the probability 0.25 of being one. In the multiplication procedure the column which contain greater than three partial products are united to structure the propagation as well as generate signals are given in figure 4.1. Subsequently they are modified to form the , , as well as if we observe the column three with weight 2^3&column number eleven with the weight 2^{11} in between these columns are replaced with the partial products , (,). the main operation involved in the implementation of this partial product are as follows

$$P_{m,n} = a_{m,n} + a_{n,m} \quad (1)$$

$$G_{m,n} = a_{m,n}.\ a_{n,m} \quad (2)$$

IV. ADDER COMPRESSORS

Compressors by far have been considered as the most efficient building blocks of a high speed multiplier. It provides an advantage of accumulation of partial products at an expense of least possible power dissipation. Rather than entirely summoning partial products with the help of CSA/Ripple adder tree, a structure of compressors would complete the same task in much lesser time and also will simultaneously eradicate the problems of large power consumption and optimization of the area. This addition of partial products when done using conventional method of implementing Full Adders and Half Adders cannot account as much to lessening of delay associated with the critical path as when counter or compressors are used. The reason for the apparent preference of compressors over counters is the advantages it provides in terms of power, number of transistors used and the delay associated with the critical path (comprising of XOR's mainly). The compressor design implemented in this paper prefers both MUX's and XOR's.

The internal structure of the 3-2 adder compressor is presented in Fig. 4-a. The maximum delay is given by two XOR gates. The final sum S of the 3-2 adder compressor is given in (3). The 3-2 adder compressor can also be used as a full-adder (i.e. mux-based full-adder) when the input C is used as a carry input.

$$S=Sum+2Carry \quad (3)$$

The internal structure of the 7-2 adder compressor is presented in Fig. 4-b. The maximum delay is given by ten XOR gates. The final sum S of the 7-2 adder compressor is given in (4).

$$S=Sum+2(Cout1+Cout2+Carry) \quad (4)$$

In this paper 8-2 adder design using 3-2 and 7-2. The internal structure of the 8-2 adder compressor is presented in Fig.6 . The final sum S of the 8-2 adder compressor is given in (5)

S=Sum+2(Cout0+Cout1+Cout2+Cout3+Cout4+Carry) (5)

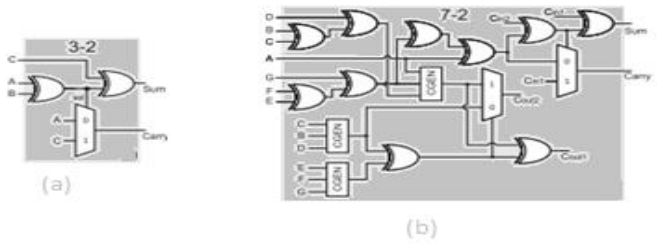

Fig. 4 Adder compressors internal structures: (a) 3-2; (b) 7-2

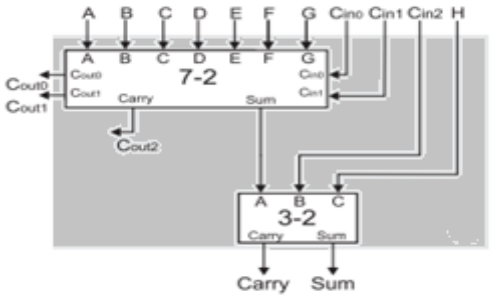

Fig. 5 The structure of 8-2 adder compressor with Combination of 7-2 and 3-2 adder compressors

V. HIGH SPEED ADDERS

For any multiplication algorithm contains three steps but in this summation of partial products is an important step to generate the final result. The performance of the multiplier depends on how fast partial products get added to obtain the final result. Many researchers can +work in this area to achieve fast adders. The fundamental adder architecture is a Ripple Carry Adder and further develops number of adders such as Carry look ahead adder, Carry select adder, Carry save adder and Carry skip adder etc. In this ripple carry adder is well known for its regular structure and maximum delay because each step waits for the carry from the previous step. Carry look ahead adder have a minimum delay but area associated with these adders are maximum. Carry skip adder gives the more performance than ripple carry adder but it's consisting of extra hardware circuitry to skip the carry generated. Carry save adder gives the further addition by reducing addition there are number of three into two. The major drawback carry save adder consumes larger area. Further carry select adder uses the two ripple carry adders and it doesn't wait for previous stage to execute. The carry select adder with higher bits exhibits excellent area and speed trade off compare with other adder architectures. Many modifications can be dined in carry save adder for sacrificing its speed for area. To implement a *W*-bit 3-2, and 7-2 adder compressors it is needed a recombination of partial *Carry* and *Sum* terms. To make the recombination of *Carry* and *Sum* it is used a cascade of half-adder and full-adders circuits in a Ripple Carry form, as presented in the example of the fig. 4, for an 8-bit 4-2 compressor. In this work we use a more efficient Carry Look Ahead adder (CLA) to perform the recombination of the partial results. We have used a pipeline stage between the line of the compressors and the adder line used to recombine the partial results (highlighted in figure 5).The design of 16-Bit high adder using different 8-2 adder compressors is shown in Fig. 6. The main objective in this section we can design high speed adders using four 8-2 adder compressors and already explained in section 4.

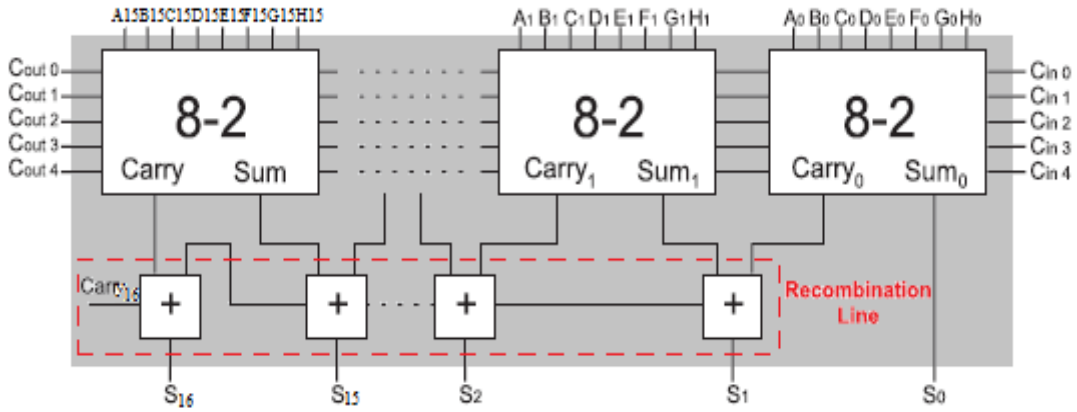

Fig. 6 16- bit adder using 8-2 ACs

VI. PROPOSED APPROXIMATE MULTIPLIER

The Proposed Multiplier is shown in Fig. 7. In this architecture, we can use above section 16-bit adder using 8-2 ACs at partial production and final sum stage after complete the process of OR gate operation. Implementation above multiplier develops the Verilog code and then simulation we can use Xilinx 14.7.

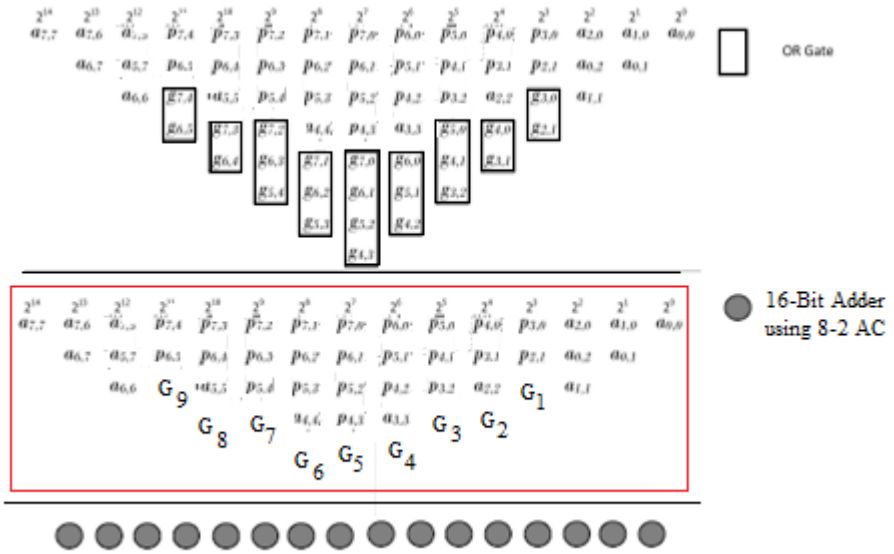

Fig. 7 The Proposed approximate Multiplier with 16-Bit adder using 8-2 ACs

VII. RESULTS AND DISCUSSION

The design was synthesized on Xilinx ISE and the functional verification of Approximate multiplier was done on Xilinx ISIM. The targeted device is of Spartan-3 of Spartan family. The grade speed of the design is set to -5. The following section contains the results obtained by synthesizing the design in Xilinx ISE 14.7.

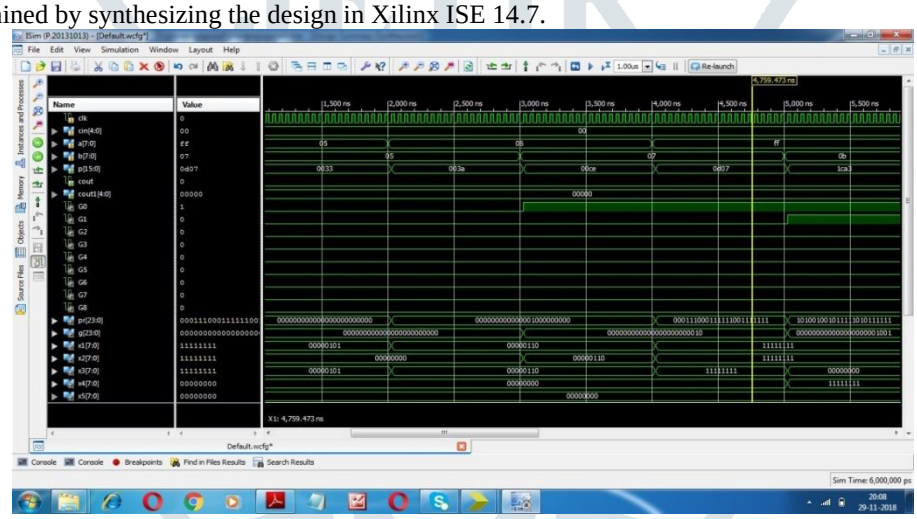

Fig. 8 simulation output1 of proposed method

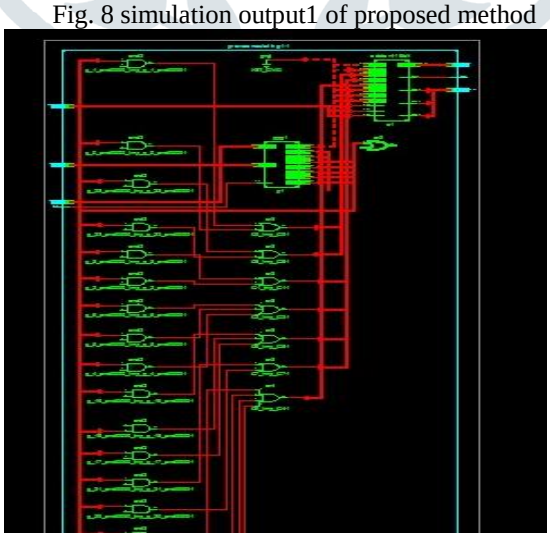

Fig. 9 RTL of proposed method

The design was synthesized on Xilinx ISE and the functional verification of approximate existing and proposed multiplier was done on Xilinx ISIM. The targeted device is of Spartan-3e of Spartan family. The grade speed of the design is set to -5. The following section contains the results obtained by synthesizing the design in Xilinx ISE 14.7. The comparison result is shown in Table 2 and resultant comparison plot is shown in Fig 10.

	AREA			DELAY (ns)
	SLICES	LUTS	FFS	
Existing Multiplier	43	85	63	16.52 ns
Proposed Multiplier	41	60	64	7.36 ns

Table 2 Comparison of Existing and Proposed Approximate Multiplier in terms of Area and delay

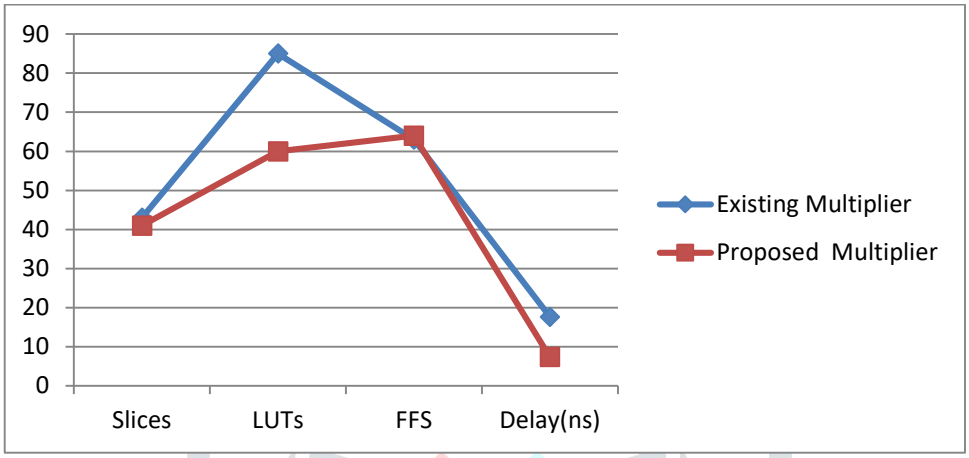

Fig. 10 Performance Analysis Of Existing And Proposed Multiplier

VIII. CONCLUSION

Proposed multiplier increases the Speed so that it can multiply and used in image processing applications. It is highly efficient, 55.44 % delay reduced compared to that of previous works. Adaptive voltage level at ground reduces the power consumption by lifting-up the ground potential whenever required and decreasing the voltage through transistors whenever needed. So this multiplier architecture can be used for high speed applications like data mining, and also the cases where the error occurrence is not a major concern.

REFERENCES

[1] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 32, no.1, pp.124–137, Jan.2013.

[2] A.Momeni, J. Han, P.Montuschi, and F.Lombardi, "Design and analysis of approximate compressors for multiplication," *IEEE Trans.Comput.*, vol. 64, no. 4, pp. 984–994, Apr. 2015.

[3] S. Narayanamoorthy, H.A. Moghaddam, Z. Liu, T. Park, and N.S. Kim, "Energy-efficient approximate multiplication for digital signal processing and classification applications," *IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst.*, vol. 23, no. 6, pp. 1180–1184, Jun. 2015.

[4] G. Zervakis, K. Tsoumanis, S. Xydis, D. Soudris, and K. Pekmestzi, "Design-efficient approximate multiplication circuits through partial product perforation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 10, pp. 3105–3117, Oct. 2016.

[5] C. Liu, J. Han, and F. Lombardi, "A low-power, high-performance approximate multiplier with configurable partial error recovery," in *Proc.Conf. Exhibit. (DATE)*, 2014, pp. 1–4.

[6] C.H. Lin and C. Lin, "High accuracy approximate multiplier with error correction," in *Proc. IEEE 31st Int. Conf. Comput. Design*, Sep. 2013, pp. 33–38.

[7] Suganthi Venkatachalam and Seok-bum KO, senior member IEEE. "Design of power and area efficient approximate multipliers" IEEE transactions on VLSI systems.2017

[8] MateshwarSingh, Surya deo Choudhary, Asutoshkr.singh, "Design and simulation of half adder circuit using AVL technique based on CMOS technology" IRJET august 2017

[9] Bishwarup mukherjee, Biplab Roy, Arindam biswas, Aniruddha ghoshal, "design of low power 4X4 multiplier based on 5t half adder, 8t full adder, 2t and gate IEEE-2015.