AN OPTIMIZED CB-UT MULTIPLIER FOR EFFICIENT DESIGN OF THE AM OPERATOR

Modalavalasa Hari Krishna[1], Gudla Sateesh Kumar[2], Laxmi Vandana[3]
[*1]M.Tech Scholar, [2]Professor & HOD, [3] Asst. Professor,
Dept. of ECE, Aditya Institute of Technology and Management,
Tekkali, Andhra Pradesh, India

ABSTRACT

The multiplication and accumulation are the vital operations involved in almost all the Digital Signal Processing applications. Add-Multiply (AM) operator or Multiply-Accumulator (MAC) units are generally employed in all high performance digital signal processors (DSP) and controllers. The performance of AM operator mainly depends on the speed of multiplier. A lot of research has been contributed in this area and the conventional multipliers were modified to provide good speed performance but needs to be improved further along with area optimization. Urdhva-Tiryakbhyam Multiplier (UTM) architecture is adopted from ancient Indian mathematics "Vedas' and can generate the partial products and sums in one step, which reduces the carry propagation from LSB to MSB. UTM can be used to implement high performance AM operators but results in larger silicon areas. This increased area can be minimized by using the modified compressor based design of UTM. In this work, the carry-look-ahead (CLA) adder is adopted instead of parallel adders for high speed of accumulation. So, the Compressor-Based-Urdhva-Tiryakbhyam (CB-UT) multiplier with CLA results in both area and performance optimization of Add-Multiply operator. The functionality of this architecture is evaluated by comparing with the Modified Booth (MB) multiplier based AM operator in terms of performance parameters like propagation delay, power consumption and silicon-area. The design is implemented and verified using Xilinx Spartan-3E FPGA and ISE Simulator..

KEYWORDS: *AM operator, CB-UT multiplier, Urdhva-Tiryakbhyam multiplier*

I. INTRODUCTION

Digital Signal Processing (DSP) systems enhance the performance of the modern consumer electronics by providing custom accelerators for domains of multimedia, communications etc. Typical DSP are employed in applications with a large number of arithmetic operations as their implementation is based on computationally intensive kernels, such as Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), Finite Impulse Response (FIR), convolution and filters. As expected, the performance of DSP systems is mostly affected by decisions on their design regarding the allocation and the architecture of arithmetic units such as Multiply-Accumulator (MAC) unit. Recent research activities in the field of arithmetic optimization of DSP systems [16], [11] have shown that the design of arithmetic units can lead to significant performance improvements by combining the operations which share data. Based on the observation of widely used DSP algorithms, the addition can often be subsequent to a multiplication. The Multiply-Accumulator (MAC) or Multiply-Add (MAD) or Add-Multiply (AM) units were introduced to address more efficient implementations of DSP systems compared to the conventional architectures, which use only primitive resources [9]. a lot of research have been contributed in this area and several architectures have been proposed to optimize the performance of the MAC units in terms of primary design constraints like area occupation, critical path delay or power consumption [10]–[12]. As noted in [13], MAC components can increase the flexibility of DSP systems' data path synthesis as a large set of arithmetic operations can be efficiently mapped onto them. The MAC unit contains adders and multipliers and the speed of MAC unit greatly depends on its multiplier's performance. This in turn raises the demand for design of multipliers with high speed performance, at the same time maintaining low area and moderate power dissipation [2]. The straight forward design of the MAC unit, by first allocating an adder and then driving its output to the input of a multiplier, increases significantly both area and critical path delay of the circuit.

Over the past few decades, several new novel architectures are developed for multipliers to improve its performance in terms of area, power and delay. Booth's [4] and modified Booth's algorithm based multipliers are highly used in modern Very Large Scale Integrated (VLSI) design but have their own set of disadvantages. In these multiplier algorithms, the multiplication process, involves several transitional operations before calculating the final result. The intermediate stages of these algorithms include several additions, subtractions and comparisons to compute the product, which reduce the speed of operation exponentially with the total number of bits present in the multiplier and the multiplicand [5]. Since the speed is major design constraint, multipliers with such type of architectures are not good approach to design MAC units, since it involves several time consuming intermediate operations. In order to address the disadvantages of the above mentioned existing methods [6] in terms of speed of operation, explored a new approach for multiplier architecture based on ancient Vedic mathematics. Vedic mathematics is an Indian ancient and eminent approach which provides a simple foundation to solve many mathematical challenges faced in the present scenario[2]. Vedic mathematics was existed in ancient India and re-discovered by a popular indian mathematician, Sri Bharati Krishna Tirthaji [7]. Tirthaji divided the Vedic mathematics into 16 simple sutras (formulae). These Sutras deals with many basic calculations such as Arithmetic, Analytical, Algebra, Geometry, Trigonometry, etc. The simplicity in the Vedic mathematics sutras shows a way for its application in several prominent domains of engineering and technology. These sutras are widely useful in Signal Processing, Control Engineering, communication systems and VLSI [14].

One of the highlights in Vedic mathematics approach is that the calculation of all partial products required for multiplication, are obtained well in development with fixed hardware architecture, before that the actual operations of multiplication begin. The generated intermediate partial products are added based on the Vedic mathematics algorithm to obtain the final product. This results in a very high speed approach to achieve multiplication operation [15].

In this work, a novel method has been proposed to further enhance in the speed of a AM operation by designing it using Urdhva -Tiryakbhyam (UT) Vedic Multiplier. The Vedic multiplier has been optimized by replacing the existing full adders and half adders of the multiplier with compressors Based (CB) adders. Compressors are logic circuits which are skilled of adding more than 3 bits at a time as divergent to a full adder and also can be designed with a lesser gate count and higher speed, in comparison with an equivalent full adder circuit and these are existed in several variants [15]. The Carry Look Ahead (CLA) adder has been adopted instead of parallel adders for high speed of accumulation during MAC operation. So, the Compressor Based Urdhva -Tiryakbhyam (CB-UT) multiplier with CLA provides both area and performance optimization of AM architecture.

Here, the performance of the proposed CB-UT multiplier based AM operator by comparing it with the existing Modified Booth multiplier based AM operator in terms of Area, power consumption and speed performance.

The rest of the paper is organized as follows: In section II, the motivation and technical background issues of present existing Modified Booth multiplier based AM operator are discussed. The proposed CB-UT multiplier based AM operator is presented in section III along with its design methodology. Section IV shows the performance evaluation of prosed method by the comparison with existing AM operator and section V concludes the work.

II. MOTIVATION AND AM IMPLEMENTATION

2.1. Motivation

In this paper, the AM unit is implemented by focusing on its speed performance as its main design constraint. The conventional architecture of the AM operator requires that its input are first driven to an adder and then the input and the sum are driven to a multiplier in order to get the final result. An optimized design of the AM operator is based on the performance of the multiply operational speed. The multiplier design can be optimized for speed or area using existing architectures but not both at once. However, the optimization of speed results in increased area and vice-versa, in many existing multiplier architectures. To address this problem, the multiplier can be designed using UTM architecture and then the area can be minimized by design the internal adder blocks using the compressor based design. Another drawback of existing AM operator design is, due to the use of a conventional adder is that it inserts a significant delay in the critical path of the AM operator. As there are carry signals to be propagated inside the n-bit adder, the critical path depends on the bit-width of the inputs. In order to reduce this delay, a Carry-Look-Ahead (CLA) adder has been used in this work, however, it results in increased area occupation and power dissipation. This adder architecture can also be optimized for area and power by implementing it using compressor based design. As a result, significant area savings are observed along with a very high speed performance. In this work, a new technique has been proposed combining the advantages of Vedic UTM, CLA adder and compressor based architecture for optimized AM operator.

2.2. Review of the Modified Booth Form

Modified Booth (MB) is a prevalent form used in multiplication operations in many existing algorithms [1]. Modified Booth form provides faster results than conventional booth form by halving the total number of partial products comparing to any other radix-2 representation. MB form is a redundant signed-digit radix-4 encoding technique. Let us consider the multiplication of two binary numbers in 2's complement form such as X and Y with each number consisting of n=2k bits. The multiplicand can be represented in MB form as:

$$Y = \langle y_{n-1}y_{n-2}\dots y_1 y_0 \rangle_{2's} = -y_{2k-1}.2^{2k-1} + \sum_{i=0}^{2k-2} y_i.2^i \quad (1)$$

$$= \langle y_{n-1}^{MB} y_{n-2}^{MB} \dots y_1^{MB} y_0^{MB} \rangle_{MB} = \sum_{j=0}^{k-1} y_j^{MB}.2^{2j} \quad (2)$$

$$y_j^{MB} = -2y_{2j+1} + y_{2j} + y_{2j-1} \quad (3)$$

Digits correspond to the three consecutive bits , , and with one bit overlapped and initially considering that .Table 1 shows how they are formed using the MB encoding technique. Each digit is represented by three types of signal bits such as one-bit, two-bit and sign-bit. The sign bit shows whether the digit is negative ($S_j = 0$) or positive ($S_j = 0$). Signal one-bit shows if the absolute value of a digit is equal to 1 ($one = 1$) or not ($one = 0$). Signal two-bit shows if the absolute value of a digit is equal to 2 ($two = 1$) or not ($two = 0$). These three bits are used to calculate the MB digits by the following relation:

$$y_j^{MB} = (-1)^{N_j}.[one_j + 2.two_j] \quad (4)$$

TABLE 1. Modified Booth Algorithm encoding

Binary			y_j^{MB}	MB encoding			Input carry
y_{2j+1}	y_{2j}	y_{2j-1}		*sign* S_j	$\times 1$ one_j	$\times 2$ two_j	C_{IN_j}
0	0	0	0	0	0	0	0
0	0	1	+1	0	1	0	0
0	1	0	+1	0	1	0	0
0	1	1	+2	0	0	1	0
1	0	0	-2	1	0	1	1
1	0	1	-1	1	1	0	1
1	1	0	-1	1	1	0	1
1	1	1	0	1	0	0	0

The encoded bit equations can be generated from the values in above truth table using K-maps. The one-bit can be obtained from xor operation and this is used to generate Two-bit with one extra and operation. The sign bit can directly obtained from

$$y_{2j+1}. \quad (5)$$

$$one_j = y_{2j-1} \oplus y_{2j} \tag{6}$$

$$two_j = (y_{2j-1} \oplus y_{2j}).\overline{one_j} \tag{7}$$

$$S_j = y_{2j+1} \tag{8}$$

2.3. Urdhva Tiryakbhyam Sutra

Vedic Mathematics is the ancient methodology of Indian mathematics which has a unique technique of calculations). Basically the ancient Indian Vedic mathematics is divided into 16 different sutras to perform many mathematical calculations. Among these 16 sutras Urdhva-Tiryakbhyam is the most preferable and efficient algorithm to perform multiplication of integers as well as binary numbers. The term "Urdhva Tiryakbhyam” originated from 2 Sanskrit words Urdhva and Tiryakbhyam. “Urdhva” means “vertically” and “Tiryakbhyam” means “crosswise” respectively. Let us consider the two 8 bit numbers X (X7-X0) and Y (Y7-Y0), where index 0 to 7 represent bits from the Least Significant Bit (LSB) to the Most Significant Bit (MSB). Number P (P15-P0) represent each bit of the multiplied result [1]. The terms P0 to P15 are calculated by adding partial products, which are calculated by using the logical AND operation between X and Y bits. The bits obtained from equations (1) to (15) are concatenated to single 16-bit binary number produce the final product of multiplication. The carry bits generated at MSB bit are ignored since they are unnecessary [12].

2.4. Compressor based adder

A compressor based adder is a combinational circuit which is used to add more than three binary numbers at a time. Here, uses 4:2 compressors and 7:2 compressor as novel building blocks for designing high level compressors. Compressor based architectures can easily swap the combination of some full adders and half adders; thereby, design of high performance processors is possible. The corresponding circuits of compressors are deliberated below.

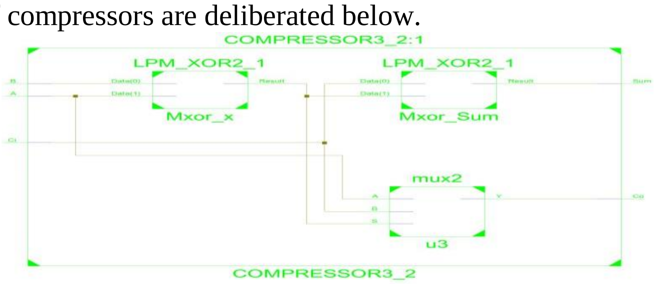

Fig.1. 3:2 compressor based adder

2.5. Compressor based Urdhva Tiryakbhyam Multiplier

The Urdhva Tiryakbhyam method based multiplication requires several full adders and half adders to add partial products which lead to a large propagation delay. So, to address this problem, a novel structure has been proposed, by combining the compressor architecture and utilized the same in the existing Urdhva-Tiryakbhyam based architecture in order to reduce the unnecessary partial products addition. The basic architecture for this design was shown in figure 2. From the figure 1, the compressor based UTM requires only 12 parallel stages whereas the conventional Urdhva-tiryakbhyam multiplier uses 15 parallel stages. This design results in the great improvement with respect to high speed and low area multiplier design and also many of the stages have been reduced.

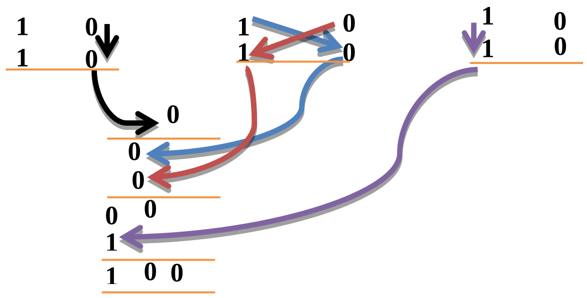

Fig.2. Basic Vedic multiplication process

III. DESIGN METHODOLOGY

In this work, the top level AM operator module is designed by following top-down design methodology. In second level, the AM operator is divided into two sub-modules such as multiplier and accumulator. Then these modules are again divided into lower level sub-modules until reach the gate level modules. The Vedic multiplier is implemented by considering a 2x2 multiplier as its basic multiplication unit which consists of 3 AND gates and 2 half adder modules. This 2x2 multiplier module has been designed using compressor based adders to reduce the area is show below figure 3. This 2x2 CBUT multiplier is used for higher level 4x4 CBUT multiplier design. This bottom-top implementation process is repeated up to designing a 16x16 CBUT multiplier using 8x8 multiplier modules.

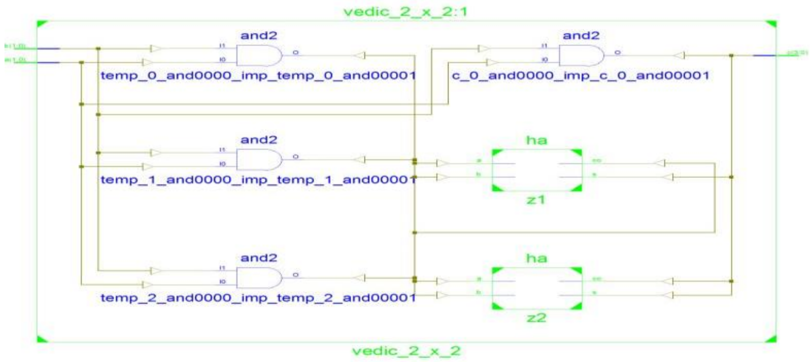

Fig.3. RTL schematic diagram of CB-UT multiplier based AM operator

Consider, "a" and "b" are two numbers to be multiplied and "p" is the product. Figure 3 illustrates the steps to multiply two 2-bit binary numbers. Converting the multiplication process to a hardware equivalent, the architecture have 3 AND gates which will act as 2-bit multipliers and two half adders to add the products to get the final product. 4x4 bit multipliers are designed using 4 such 2x2 multipliers and 3 adders as shown in the design with proper instantiating of the 2x2 multipliers and adders. Here, the adder modules are of compressor type to improve the performance. This bottom-top implementation is carried out until integrate 16x16 multiplier using four 8x8 multipliers and 3 compressor based adder modules as shown in below figure 4[3].

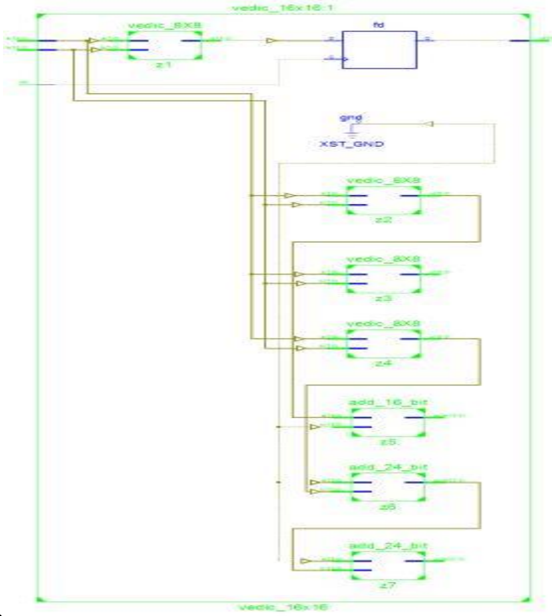

Fig.4. RTL schematic diagram of 16x16 CB-UT multiplier

This 16x16 optimized CB-UT Vedic multiplier is used for design of high performance AM Operator by connecting it with a accumulator module as shown in below figure 5. The AM operator consists of a CB-UT multiplier and an accumulator. The accumulator composed of an adder and register to store the result for one clock period. An extra flip-flop is used for storing the carry of adder for further addition. This extra flip-flop is used to store MSB in order to avoid overflow effect.

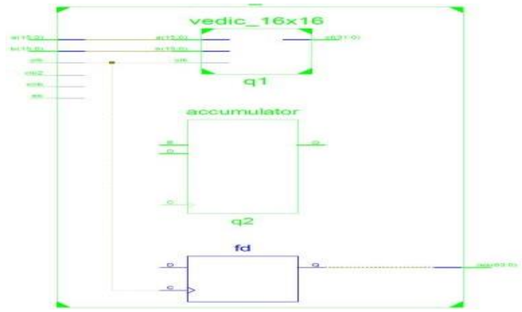

Fig.5. RTL schematic diagram of CB-UT multiplier based AM operator

IV. RESULTS

To evaluate the performance of the proposed CB-UT based AM Operator a comparison has been carried out in between two architectures of AM Operator such as AM Operator using compressor based Urdhva Tiryakbhyam multiplier and AM Operator using MB Multiplier. These two architectures were designed and implemented on a XILINX Spartan-3E-XC3S500E FPGA by using Verilog HDL as the RTL programming language by using Xilinx Project Navigator ISE 13.2. The Spartan-3E used for the experiments has a speed grade of 5 and package FG320. The simulated and synthesized results have been tabulated in Table 1.

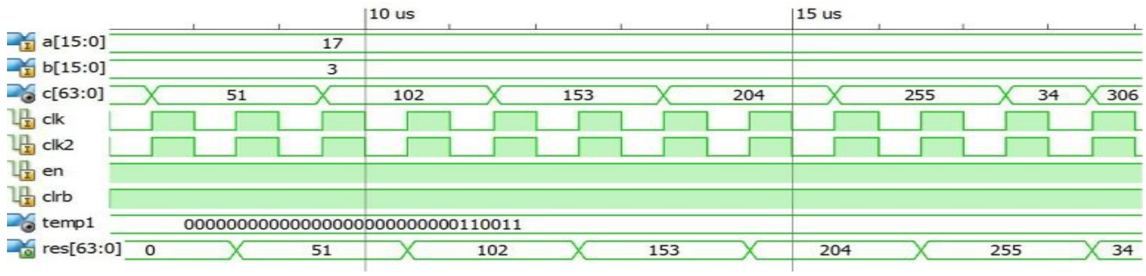

Fig.6. Simulation timing waveform diagram of AM operator

The simulation results are used to check the correctness of functionality of circuit. 'a' and 'b' are the inputs 'c' is the output. Two global clocks 'clk' and 'clk2' are used for synchronization of sub-modules of the AM operator. 'en' is the enable signal to activate the circuit and 'clrb' is the clear signal in negative logic. As shown in figure 6 let consider, 'a' as 17(0000000000010001) , 'b' as 3 (0000000000000011), 'en' as '1' and assign 'clrb' with '0' and again set to '1'. 'clk' and 'clk2' are assigned with two clock signals. Then the AM operation can be observed with accumulated operation. The synthesis summery reports of AM operators using modified Booth multiplier and CB-UT multiplier were shown in figure and figure respectively, and their Technology map schematic diagrams were also shown in figure and figure respectively.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	825	4656	17%
Number of Slice Flip Flops	233	9312	2%
Number of 4 input LUTs	1526	9312	16%
Number of bonded IOBs	100	232	43%
Number of GCLKs	2	24	8%

Fig.7. synthesis summery reports of AM operators using modified Booth multiplier

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	498	4656	10%
Number of Slice Flip Flops	278	9312	2%
Number of 4 input LUTs	941	9312	10%
Number of bonded IOBs	100	232	43%
Number of GCLKs	2	24	8%

Fig.8. Synthesis summery reports of AM operators using CB-UT multiplier

The area optimization of the CB-UT multiplier based AM operator can be evaluated from the comparison of above utilization summery reports. MB multiplier based AM operator requires 1526 number of 4-input LUTs whereas the CB-UT 941 number of LUTs which reduce the area by 6% of total available LUTs. The number of slices also reduced from 825 to 498 which results in reduction by 7% of total available slices.

Fig.9.Technology schematic diagram of modified booth multiplier based AM operator

The technology schematic diagrams represent the placement and routing of LUTs which are used for designed circuits. It also provides the information regarding the input-output buffers and global clocks along with input-output ports.

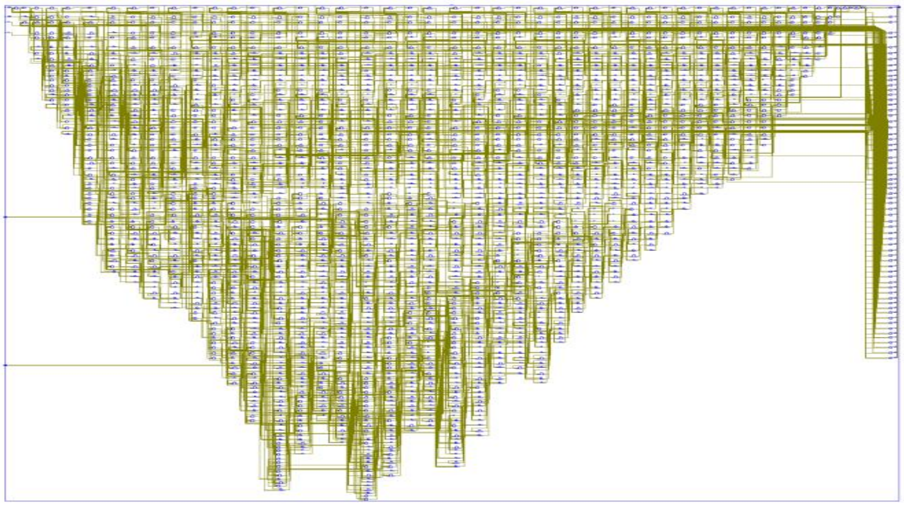

Fig.10. Technology schematic diagram of CB-UT multiplier based AM operator

The area reduction can directly observed from the above technology schematic maps of AM operators using MB multiplier and CB-UT multiplier.

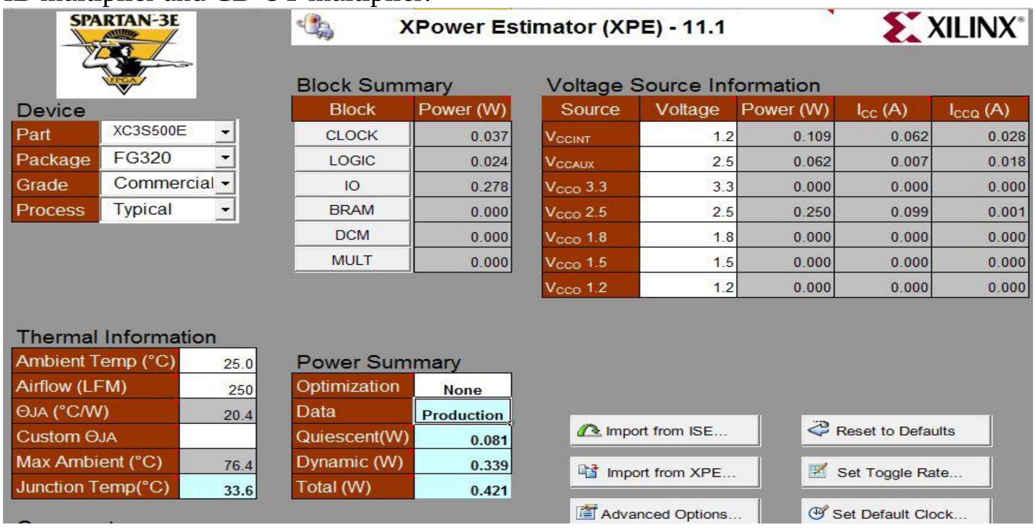

Fig.11. power analysis using XPower Estimator.

The static power can directly obtained from synthesis results but Xilinx software can't provide dynamic power information. For this dynamic power analysis the Xilinx provides plugin support such as XPower Estimator (XPE). XPower Estimator-11.1 is Microsoft Excel spread book, which provides detailed power analysis by using mapping report file generated during the synthesis using Xilinx synthesizer. The dynamic power is 339mW and static power is 81mW. The total power is 421mW whereas the total power consumption for MB multiplier based AM operator is 437 mW.

Table 2. Performance evaluation table

Parameter	MB_AM	CBUT-AM
AREA(slices)	825	498
AREA(LUTs)	1526	941
Delay (ns)	62.76	16.71
Power (mw)	437	421
power * Delay	2742.612	703.491

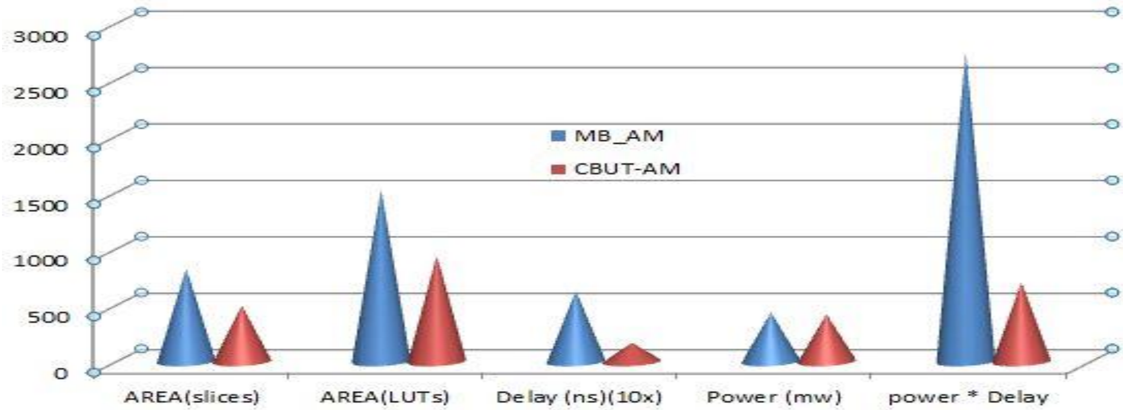

Fig.12. Performance evaluation comparison of MB multiplier based AM operator and CB-UT based AM operator

The above comparison table shows that the CB-UT multiplier based architecture provides optimization for AM operator design in terms of area, power and speed performance. The combinational delay can be efficiently reduced from 62.76 ns to 16.71 ns by replacing MB multiplier with CB-UT multiplier. The power delay product of proposed architecture also compared against that of MB multiplier based AM operator. The power delay product improved by 74.3%.

V. CONCLUSION

This work focuses on optimizing the design of Add Multiply operator. The proposed structured architecture for the CB-UT multiplier optimizes the performance of multiplier both in terms of speed and area. The CLA adder using compressor based adder provides the accumulation in one system clock. Hence, the compressor based UTM and CLA adder integrated to optimize the speed and area of the AM operator. The performance is evaluated by comparing with MB based AM operator using Xilinx ISE in terms of area, power and delay. This efficient architecture of AM operator reduces the combinational delay from 62.76 ns to 16.71 ns and exhibits better speed performance along with efficient reduction in area and power.

VI. FUTURE SCOPE

The proposed AM operator architecture can be utilized in high performance DSP processors and Micro controllers. Even though the proposed architecture provides better speed, area and power performance, still it has to be optimized in terms of customized placement and routing for full-custom VLSI design. The proposed CB-UT multiplier architecture has to be synchronized with the system clock control by inserting delay control and synchronization mechanism between input and output.

REFERENCES

[1] Kostas Tsoumanis, Sotiris Xydis, Constantinos Efstathiou, Nikos Moschopoulos, and Kiamal Pekmestzi "An Optimized Modified Booth Recoder for Efficient Design of the Add-Multiply perator", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: REGULAR PAPERS, VOL. 61, NO. 4, APRIL 2014

[2] N.Rajasekhar, Dr.T.Shanmuganantham "A Modified Novel Compressor based Urdhwa Tiryakbhyam Multiplier." On International Conference on Computer Communication and Informatics (ICCCI -2014), Jan. 03 – 05, 2014, Coimbatore, INDIA, 2014.

[3] Harish Babu N ; Satish Reddy N ; Bhumarapu Devendra ; Jayakrishanan P," Pipelined architecture for vedic multiplier", IEEE International Conference on Advances in Electrical Engineering (ICAEE), Jan.01,2016, Vellore,2016.

[4] A.D.Booth, "A Signed Binary Multiplication Technique," J. mech. Andappl. Math, vol 4, no.2, pp. 236-240, Oxford University Press, 1951.

[5] M. Ramalatha, K. Deena Dayalan, S. Deborah Priya, “High Speed Energy Efficient ALU Design using Vedic Multiplication Techniques,” Advances in Computational Tools for Engineering Applications, 2009, IEEE Proc., pp 600-603.
[6] Jagadguru Swami Sri Bharati Krisna Tirthaji Maharaja, “Vedic Mathematics: Sixteen Simple Mathematical Formulae from the Veda,” pp. 5-45, MotilalBanarasidas Publishers, Delhi, 2009.
[7] Tiwari, Honey Durga, et al., "Multiplier design based on ancient IndianVedic Mathematics," Int. SoC Design Conf., 2008, vol. 2. IEEE Proc.,pp. II-65 - II-68.
[8] Sushma R. Huddar Sudhir Rao , Kalpana M and Surabhi Mohan ,” Novel High Speed Vedic Mathematics Multiplier using Compressors” IEEE Conf.VLSI, pp.465-469, 2013
[9] S.Nikolaidis, E.Karaolis, and E.D.Kyriakis-Bitzaros, “Estimationof signal transition activity in FIR filters implemented by a MAC architecture,” IEEE Trans. Computer-Aided Des. Integrated Circuits Systems, vol. 19, no. 1, pp. 164–169, Jan. 2000.
[10] O.Kwon, K.Nowka and E.E.Swartzlander, “A16-bit by 16-bit MAC design using fast 5:3 compressor cells,” J. VLSI Signal Process Systems, vol. 31, no. 2, pp. 77–89, Jun. 2002.
[11] E. E. Swartzlander and H. H. M. Saleh, “FFT implementation with fused floating-pointoperations,” IEEE Trans. Comput., vol. 61, no. 2, pp. 284–288, Feb. 2012
[12] Y.-H. Seo and D.-W. Kim, “A new VLSI architecture of parallel multiplier-accumulator based on Radix-2 modified Booth algorithm,” IEEE Trans. Very Large Scale Integrated. (VLSI) Syst., vol. 18, no. 2, pp. 201–208, Feb. 2010.
[13] A. Peymandoust and G. de Micheli, “Using symbolic algebra in algorithmic level DSP synthesis,” in Proc. Design Automation Conf., Las Vegas, NV, 2001, pp. 277–282.
[14] Hsiao, Shen-Fu, Ming-Roun Jiang, and Jia-SienYeh, "Design of high speed low-power 3-2 counter and 4-2 compressor for fast multipliers”. IEEE Electronics Letters, vol. 34, no.4, pp. 341-343, Feb. 1998.
[15]. D. Radhakrishnan and A. P. Preethy, "Low power CMOS pass logic 4-2compressor for high-speed multiplication," Circuits and Systems, Proc.43rd IEEE Midwest Symp, vol. 3, pp. 1296-1298, 2000.
[16] A.Amaricai, M.Vladutiu andO.Boncalo,“Design issues and implementations for floating-point divide-add fused,” IEEE Trans. Circuits Syst.II–Exp. Briefs, vol.57,no.4,pp.295–299,Apr.2010.

AUTHOR PROFILE

M. Hari Krishna is presently pursuing his M.Tech in VLSI system design in Electronics and Communication Engineering Department, AITAM, Tekkali. His areas of interest are Low Power VLSI system design and digital filter optimization. He has attended for one international conference and two national level workshops. The author may be reached at mhkrishnamtech@gmail.com.

G. Sateesh Kumar is presently working as Professor and Head of the Department in Electronics and Communication Engineering Department, AITAM, Tekkali. He completed his Ph.D from Andhra University in the Dept. of Electronics and Communication Engineering. He has 14 experience years in teaching and research. He published more than 18 research papers in National/ International Journals and Conferences. He is a life member of ISTE.

Laxmi Vandana is presently working as Assistant Professor in Electronics and Communication Engineering Department, AITAM, Tekkali. She completed her M.Tech from JNT University in the Dept. of Electronics and Communication Engineering. He has 10 experience years in teaching and research. She published 4 research papers in National/ International Journals and Conferences.