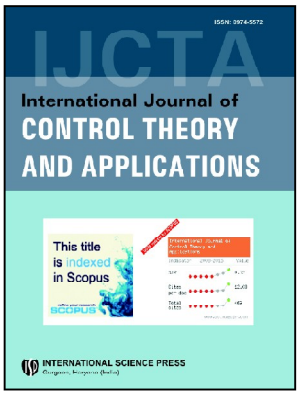

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 9 • Number 47 • 2016

FPGA Implementation of Multiplication and Accumulation Unit using Vedic Multiplier and Parallel Prefix adders in SPARTAN 3E

Pavan Ambati, Mrudula Singamsetti, K. Sai Priyanjali and O. Nageswaramma

Abstract: In this paper proposes the design of the MAC unit with high-speed Vedic Multiplier utilizing the techniques of Urdhva Tiryakbhyam[1] along with high-speed adders i.e. Ladner-Fischer, Han-Carlson, Brent-Kung, Kogge Stone adders etc., these adders are added to improve the performance of any multiplier. Mainly DSP processor depends on the multiplier as it is one of the key hardware blocks in general processors as well as in digital system designs. The basic operations involved in Digital systems are addition and multiplication. Depending on the performance of adders, speed and accuracy of a digital system are designed. The main purpose of adders is to compute additions of partial products very fast during multiplication operation. In the conventional paper, the Vedic multiplier has implemented by using ripple carry adders.But in tree adders, all the carry bits are executed in parallel at the costly in terms of area and power. The main advantage of carry tree reduces the states of logic levels (N) by pertaining to generate the carriers in parallel. The parallel prefix adders possess high speed due to the complexity O (log2N) and the delay was very low [3] [4]. Simulation and Synthesis report were made by using XilinxISE14.2 version.

Key Words: Adders, Multiply Accumulate Unit, Vedic multiplier, Verilog, Spartan 3E

1. INTRODUCTION

Multiply and accumulation unit is playing a key role in many of the Digital signal processing / Image processing /Audio processing systems such as Convolution, Filtering, and Inner products power dissipation is identified as one of the critical parameters to produce high speed and low power dissipation in real time applications. Many of the DSP modules are accompanied by repetitive applications of multiplication and addition operations. MAC unit figures out the multiplication of numbers and then adds that result value to an accumulator which stores the result. An adder is one of the basic block modules in the MAC unit. There are different types of adders like Half adder, Full adders, Carry looks ahead adder, Ripple carry adders, etc., but if the width of the adder increases propagation delay passing the carry to the next stages also increases. Hence in current technology parallel prefix adders are efficient with respect to area and propagation delay for high-speed computations and larger widths. In the Parallel Prefix adders, prefix means the execution of the operands depends on the initial inputs and parallel means execution of the operands in parallel. The overall module is segmented into smaller modules can be figure out in parallel. Then all bits of the sum will execute the process concurrently. There are different parallel prefix adders J. Sklansky conditional adder, Ladner-Fisher adder, Brent-Kung adder, Han-Carlson adder, S. Knowles, Kogge-Stone adders.These Parallel Prefix adder has a small fan-out from each prefix cell but has a large critical

path multiplier, the combinational path delay for various adders hardware design is evaluated[3][4]. The multiplier is also one of the basic building blocks in our MAC unit as well as in many FFT's and DFT's.With the advancement in technology, many researchers have tried to design multipliers with less area, low power consumption high speed. Hence moved to the efficient multiplying algorithm's which can be implemented in the chips. Conventional Multipliers are classified into various categories like a serial multiplier, Array multipliers, and Serial-parallel multiplier. A serial multiplier which is simple in construction i.e., hardware with least possible amount of chip area and the disadvantage is the input of the next stages of the design is dependent on previous stages output which leads to delay consumption. A parallel multiplier which carries high-speed mathematical computations and the drawback is it occupies a larger area. In our MAC architecture, we are using Vedic multiplier which is fast, efficient and easy to understand. The Vedic mathematics rediscover from the Veda. He invented 16 Vedic Sutras deals with the mathematics related to arithmetic, algebra and geometry [1].The Vedic multiplier is taken from the sutras. Vedic means Knowledge, Ganitha sutras means mathematics. Joining together called as knowledge of mathematics. In olden days this multiplication is used only for decimal number system, but now it is also applicable to binary number system which is going to be compatible with digital hardware. Vedic mathematics is an ancient technique which helps in simplifying multiplying computations. The subject was revised largely due to the efforts of Jagadguru Swami Bharathi Krishna Tirthaji of Govardhan Peeth, Puri Jagannath (1884-1960). In our MAC structure we are using Urdhva Tiryakbyham Vedic multiplier [1].

2. PARALLEL PREFIX ADDERS

This is a parallel prefix taken a concept from the carry look-ahead adder. Parallel prefix adders are classified into different adders they are Han-Carlson adder, Ladner-Fischer adder, Lynch-Swartz lander Brent-Kung adder, and Kogge–Stone adder [3][4][5].

Kogge-Stone adder

The Kogge–Stone adder has a reduces fan-out at the individual stage, which improves the achievement for CMOS process nodes and occupies more area to implement. However, the wiring is a heavy problem for Kogge–Stone adders. 4 bit and 16 bit Kogge stone adders diagram which are shown in the given fig 1 and fig 2. Each

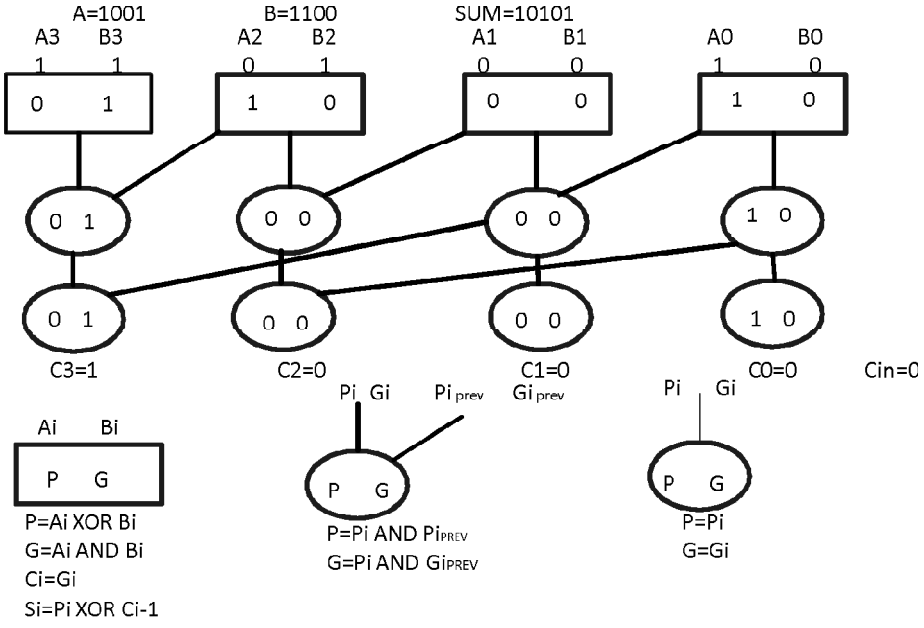

Figure 1: 4 Bit Kogge Stone Adder Block Diagram

stage consists of propagating and generate blocks, the final result of generating bits are generated in the last stage which is exerted the first propagate after the input to generates sum bits. If the number of bits increases level by level power and delay of each stage also increases. Caries generated are used as a carry inputs for a ripple carry adder which generates final sum bits. Increasing sparsity (generating carry bits) leads to the reduction in a number of computations and reduces the routing distance. Table I represents the Report analysis of Kogge-Stone Adder for 4,8,16 bits.

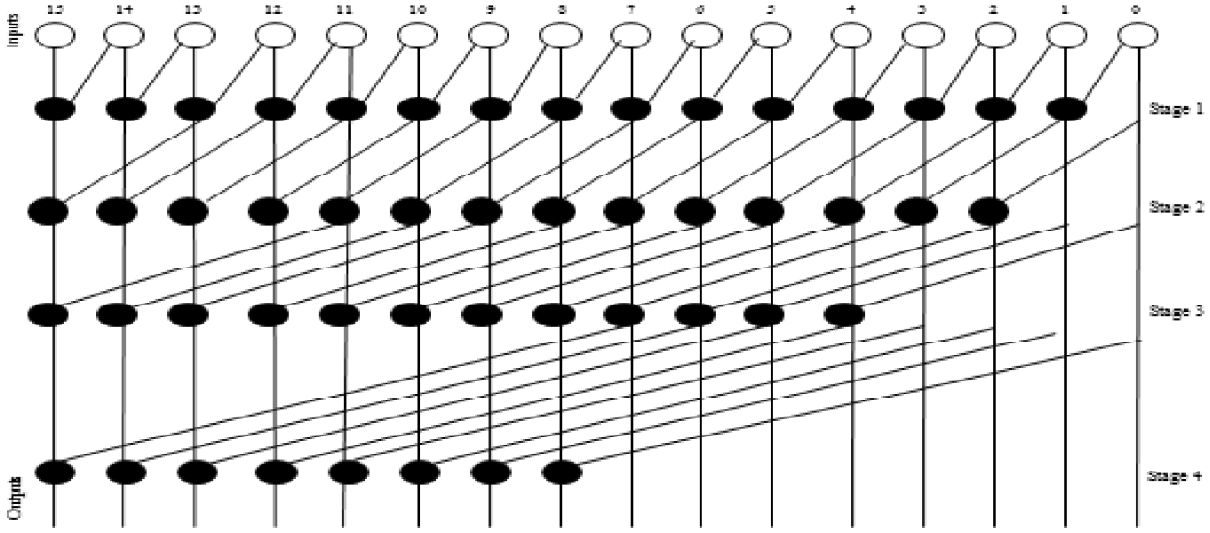

Figure 2: 16 Bit Kogge Stone Adder Block Diagram

Table I
Report Analysis for Kogge Stone Adder

Report Analysis	*4BIT*	*8BIT*	*16BIT*
Path Delay(ns)	8.112	10.21	13.19
No of slices	5	19	52
No of LUTS	9	32	90
No of Bonded IOB'S	14	25	50

Brent-Kung adder

Brent-Kung adder is a very notable as parallel prefix logarithmic adder architecture which is shown in fig 3 shows Brent-Kung adder and number of stages are more but intermediate stage connections are less than compare with other adders. For Brent-Kung Adder requires low wiring, cost and gate level depth is 0 (log2 (n)). Hence the power is lower. Table II shows the report analysis of Brent-Kung adder for 4,8,16 bits.

Table II
Report Analysis for Brent-Kung Adder

Report Analysis	*4BIT*	*8BIT*	*16BIT*
Path Delay(ns)	7.898	10.964	17.728
No of Slices	4	9	19
No of LUT'S	7	17	34
No of Bonded IOB'S	14	25	50

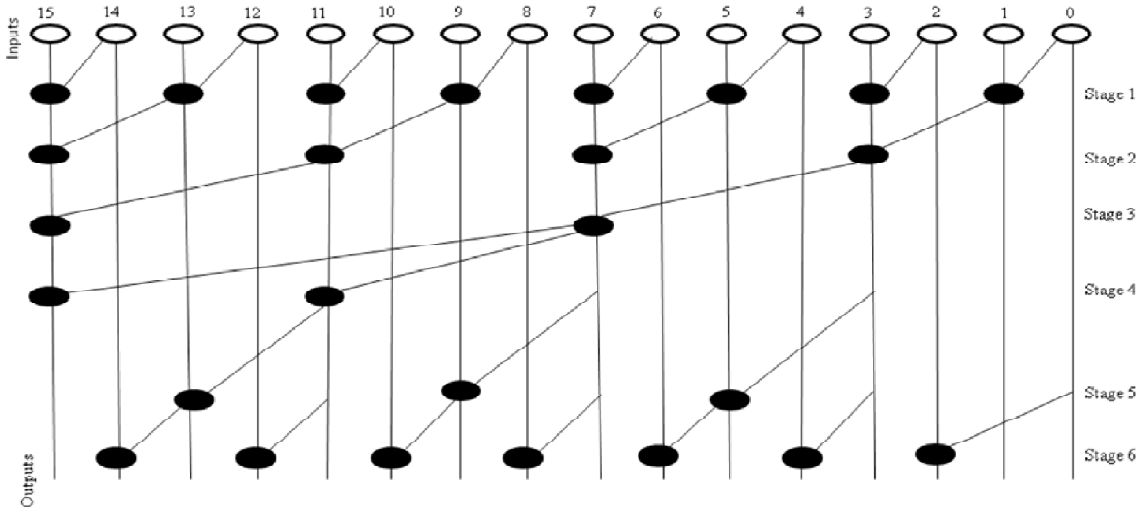

Figure 3: 16 Bit Brent-Kung Adder Block Diagram

Ladner-Fischer adder

Ladner-Fischer adder 16-bit architecture was shown in fig 5. The time prescribed to set up carry signals in Ladner-Fischer adder is O (log n). The Ladner-Fischer adder concept was advanced by R. Ladner and M. Fischer. The better achievements of a Ladner-Fischer adder have Low depth High fan-out nodes and the disadvantage is a large area. Table III shows the report analysis of Ladner-Fischer adder for 4, 8, 16bits.

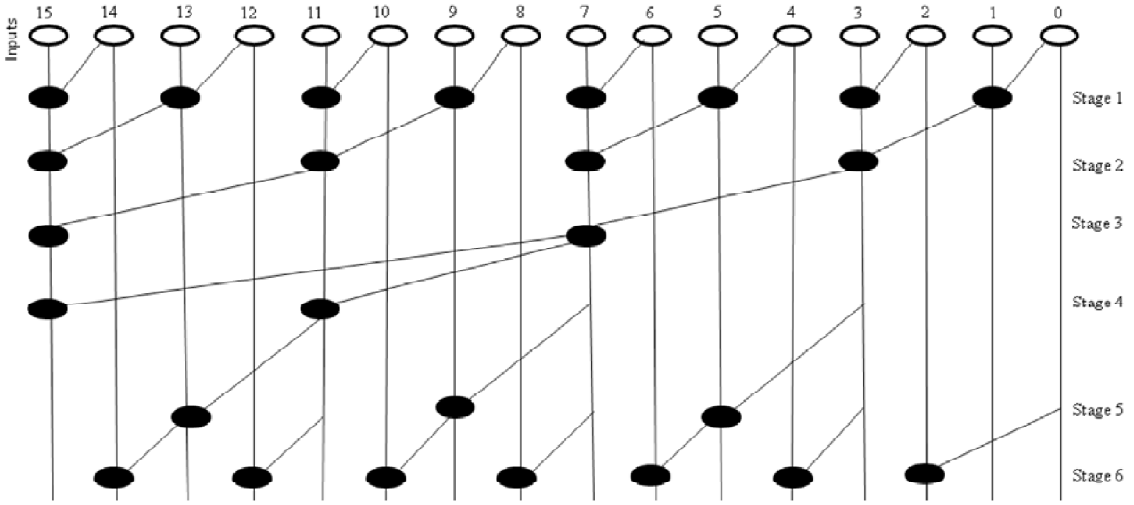

Figure 5:16 Bit Lander Fischer Adder Block Diagram

Table III
Report Analysis for Lander Fischer Adder

Report Analysis	*4BIT*	*8BIT*	*16BIT*
Path Delay(ns)	7.898	11.074	13.822
No of Slices	4	11	25
No of LUT'S	7	19	45
No of Bonded IOB'S	14	25	50

3. VEDIC MULTIPLIER

2Bit Vedic Multiplier: The main principle of Vedic mathematics is to reduce the calculations which are based on Urdhva Tiryakbhyam which is one of the basic 16 sutras. These Vedic sutras are helpful for complex multiplication numbers (n*n) in the decimal number system. The multiplication is purely based on vertical and crosswise multiplication [9] [10].2 bit Vedic multiplier algorithmic approach is shown in fig 6. The Algorithm for Urdhva Tiryabhyam is as follows.

Algorithm

1) Multiply the number a by b.
2) Multiply the LSB bits of the given numbers this gives the LSB digit answer.
3) Multiply LSB (a_0) of the top number with the second bit MSB of the bottom number (b_1) and vice versa with the MSB of the a_1 with the LSB of b_0 and add them together
4) Add step 2 and 3
5) Multiply MSB bits of the given numbers and move one place to the left and added with step 4 which gives us the multiplication result.

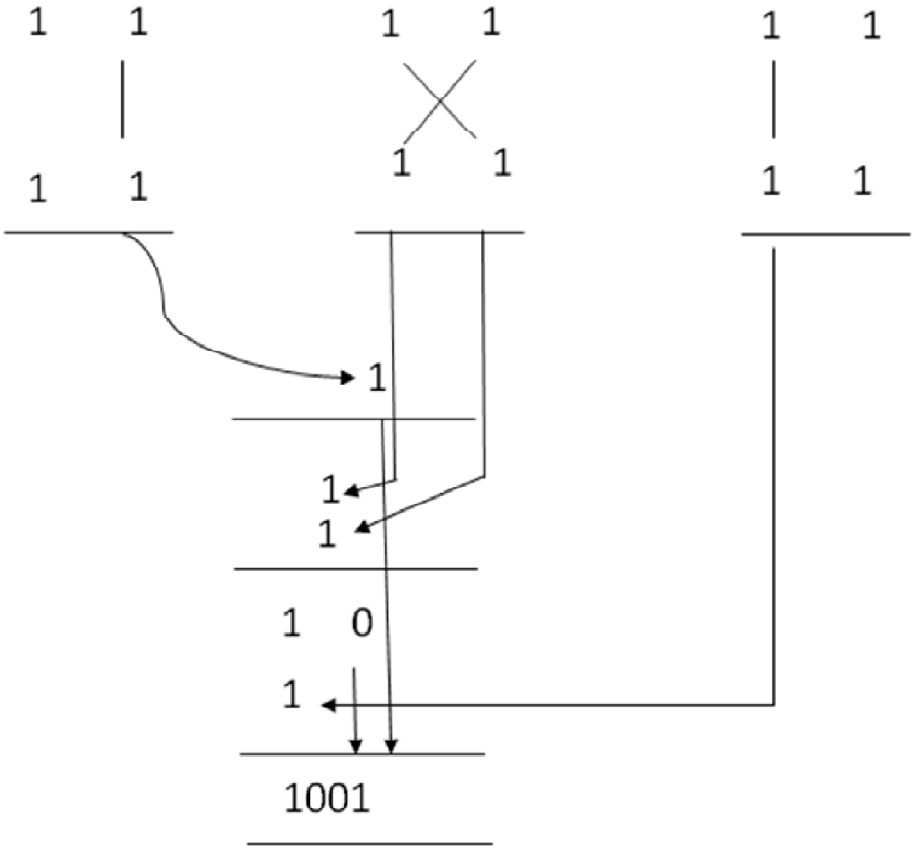

Figure 6: 2 Bit Vedic Multiplier Algorithmic Approach

Block diagram of 2-bit Vedic multiplier which is shown in fig 7,based on the Vedic Algorithmic approach multiplication can be done to (n*n) bit numbers with less number of calculations. In multiplication approach

along with computation complexity delay, speed and area are also one of the major criteria. If we use ripple carry adders in the place of adders Delay is one of the major problems .In order to increase speed, we are using parallel prefix adders as a major component in our module because execution is done parallel. Hence in the place of ripple carry adders we are using parallel prefix adders (Brent-Kung adder, Han-Carlson adder, Ladner-Fischer, Kogge–Stone adder) in the Vedic multiplier and the comparison results which is shown in the given table IV. Block Diagram, RTL Schematic and output waveforms for 16-bit Vedic multiplier which is shown in fig 8, fig 9 and fig. 10.

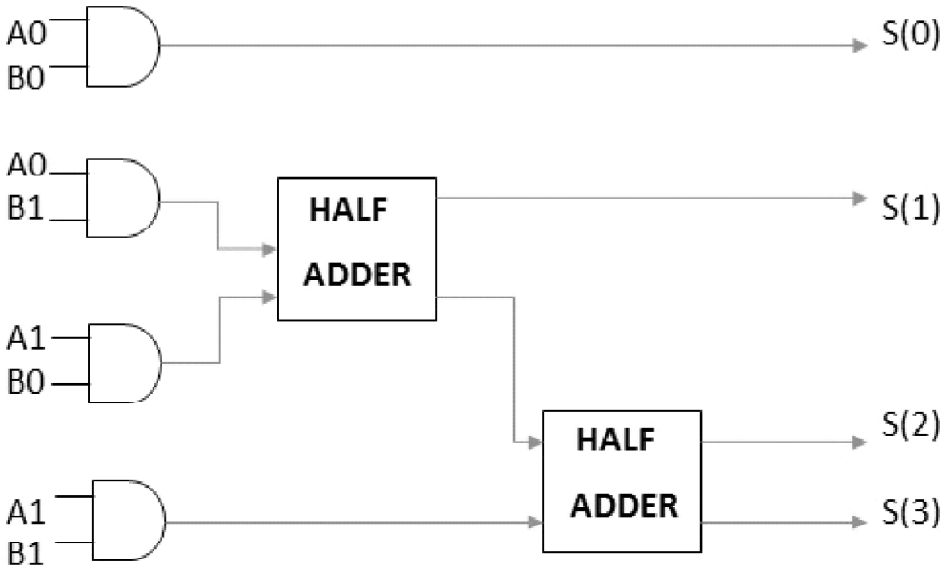

Figure 7: 2 Bit Vedic Multiplier Block diagram

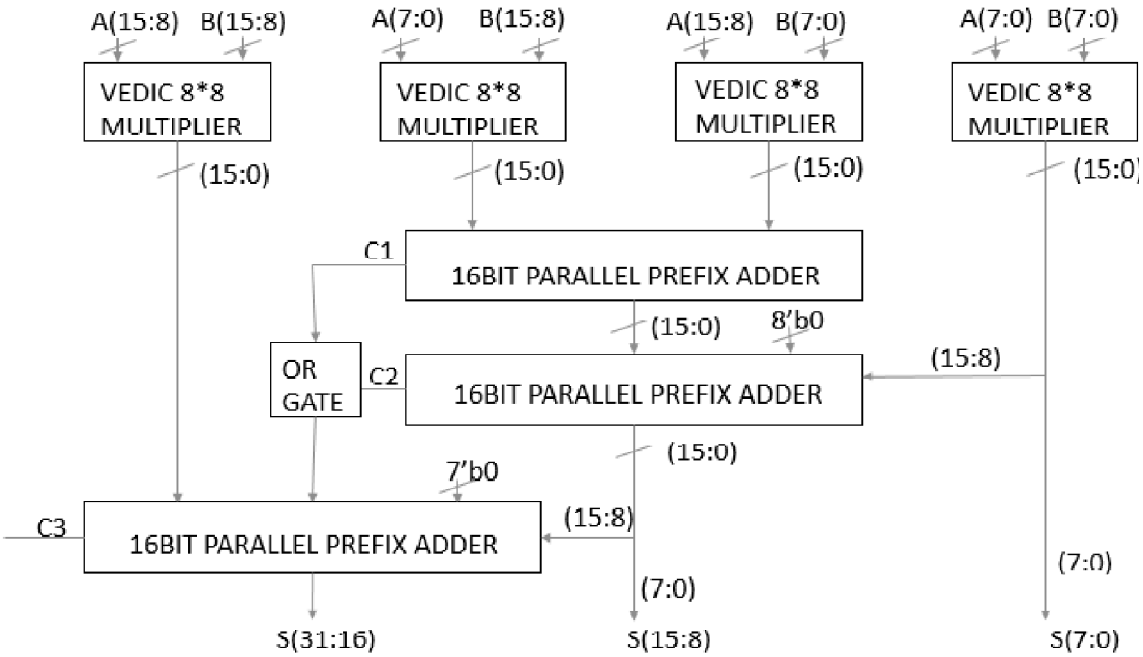

Figure 8: 16 Bit Vedic Multiplier Block diagram

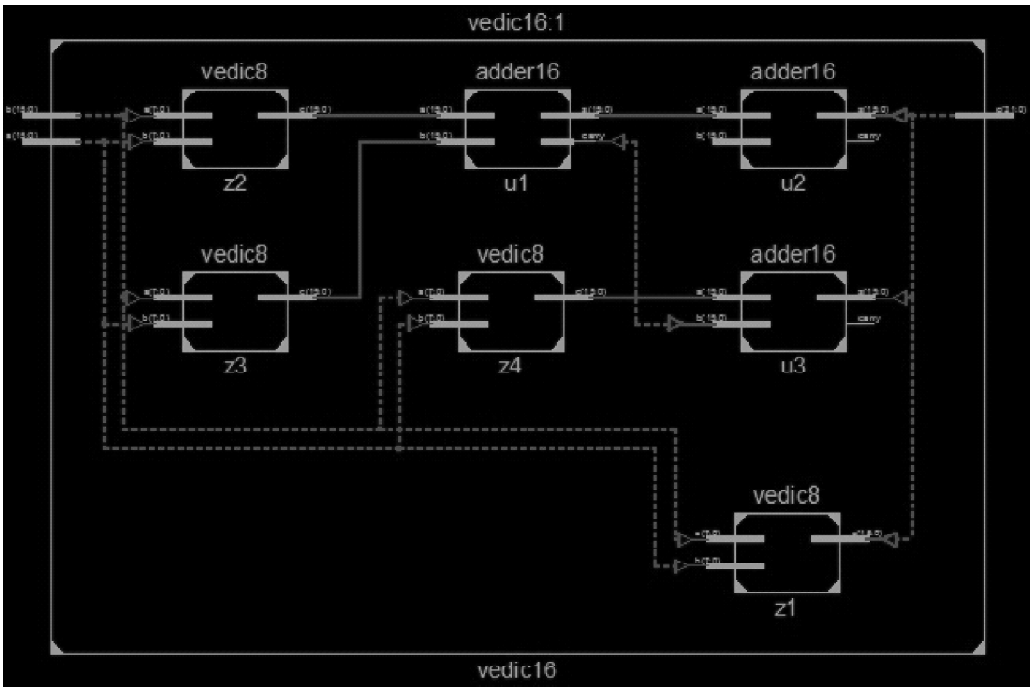

Figure 9: RTL Schematic for 16-bit Vedic Multiplier

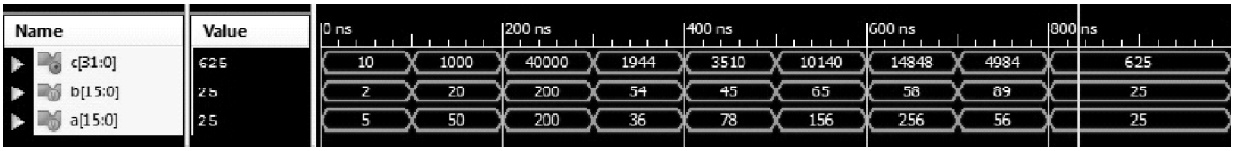

Figure 10: Output Waveforms for 16-bit Vedic Multiplier

4. CONVENTIONAL MAC UNIT

The Multiplication and accumulation unit contains the basic multiplier and basic adder, which performs multiplication operations, adder performs arithmetic operations and an Accumulator which store the result [6] [8]. The output is feedback to the adder, for every clock cycles adds the multiplication result value to the accumulator value. By using conventional adders like parallel adders and conventional multipliers propagation delay is more [6] [8] .Hence we are going for the new Architecture which was implemented by using Parallel prefix adders and Vedic multiplier.

Table IV
Vedic Multiplier by using Parallel Prefix Adders

Multiplication Unit	*Kogge Stone Adder*			*Brent Kung adder*			*Ladnerfischer adder*		
NO.OF BITS USED	4	8	16	4	8	16	4	8	16
COMBINATIONAL PATH DELAY(ns)	11.764	21.809	34.236	12.59	21.593	36.14	12.59	21.79	34.381
NO. OF SLICES	21	134	673	19	101	452	19	101	467
NO. OF.4 I/P'S LUT'S	37	238	1182	34	176	788	34	176	813
NO. OF IOB'S	16	32	64	16	32	64	16	32	64

Proposed MAC Unit: In the conventional MAC unit propagation delay is caused by the adders present in the multiplication unit as well feedback path addition from the accumulation unit. Hence the multiplication unit is replaced by the Vedic multiplier and adders is replaced by parallel prefix adders. The proposed 16 bit MAC unit which is shown in fig. 11.

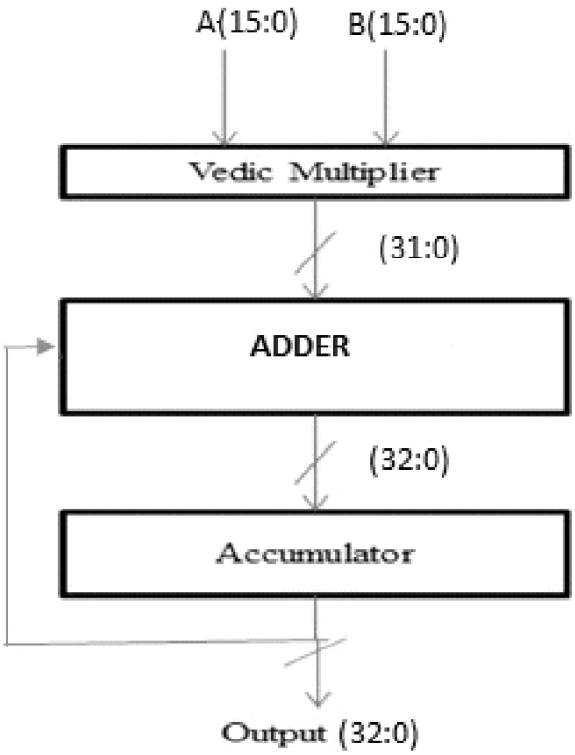

Figure 11: 16Bit proposed MAC unit

5. SIMULATION RESULTS

The output waveforms, RTL Schematic of the 16 bit MAC unit is shown in the fig. 12, fig. 13.

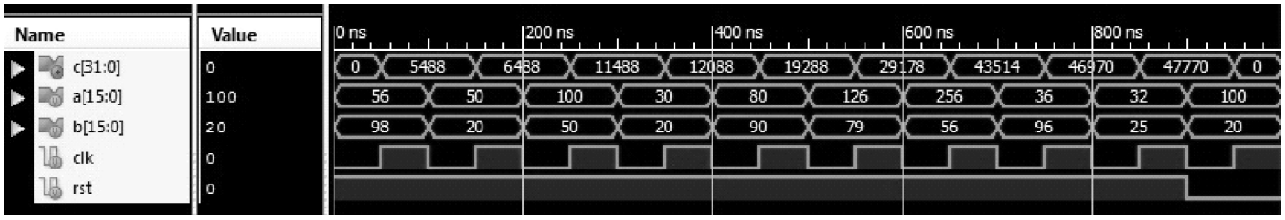

Figure 12: Outputs of Proposed MAC

6. EXPERIMENTAL RESULTS

The proposed MAC Unit consists of Vedic multiplying units with parallel prefix adders and the conventional MAC unit is implemented by using conventional adders like ripple carry adder with array multiplication has synthesized in Xilinx 13.4 by using the Verilog code and the results have been verified by chip scope pro. The architecture is implemented in FPGA kit by using SPARTAN 3-E shown in fig 14. Maximum period, Maximum Period, Maximum Frequency, Delay, LUTS, Slices report, which is shown in the given Table V. Fig 15, 16, 17

shows the detailed graph for maximum frequency, Number of LUT'S and Number of slices respectively. There is a tradeoff between speed and area.

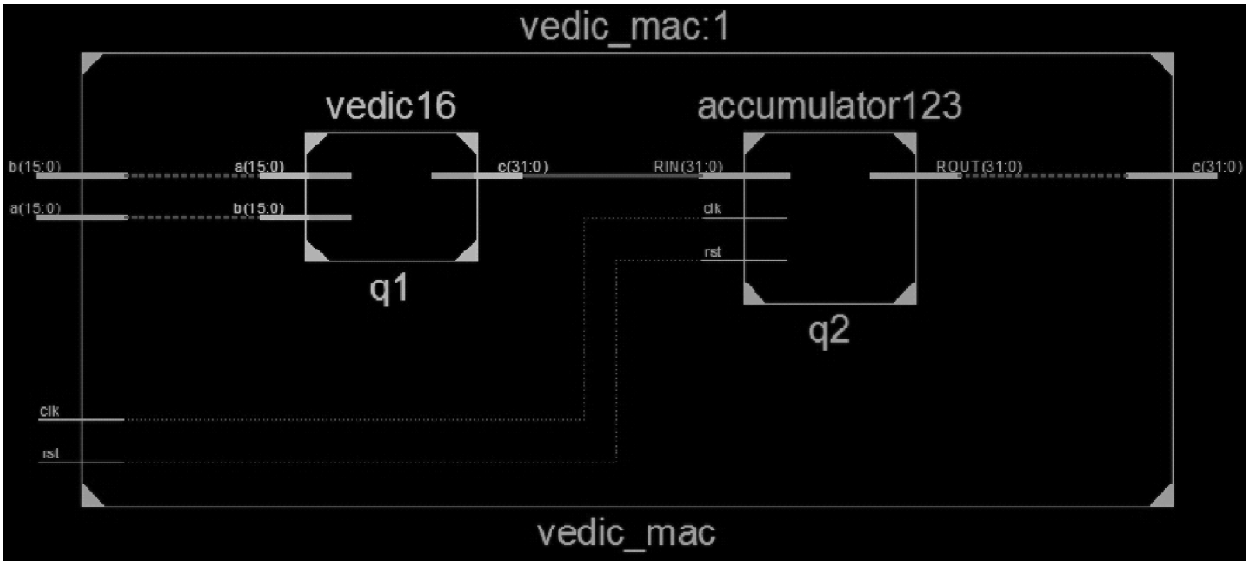

Figure 13: RTL Schematic of MAC Unit

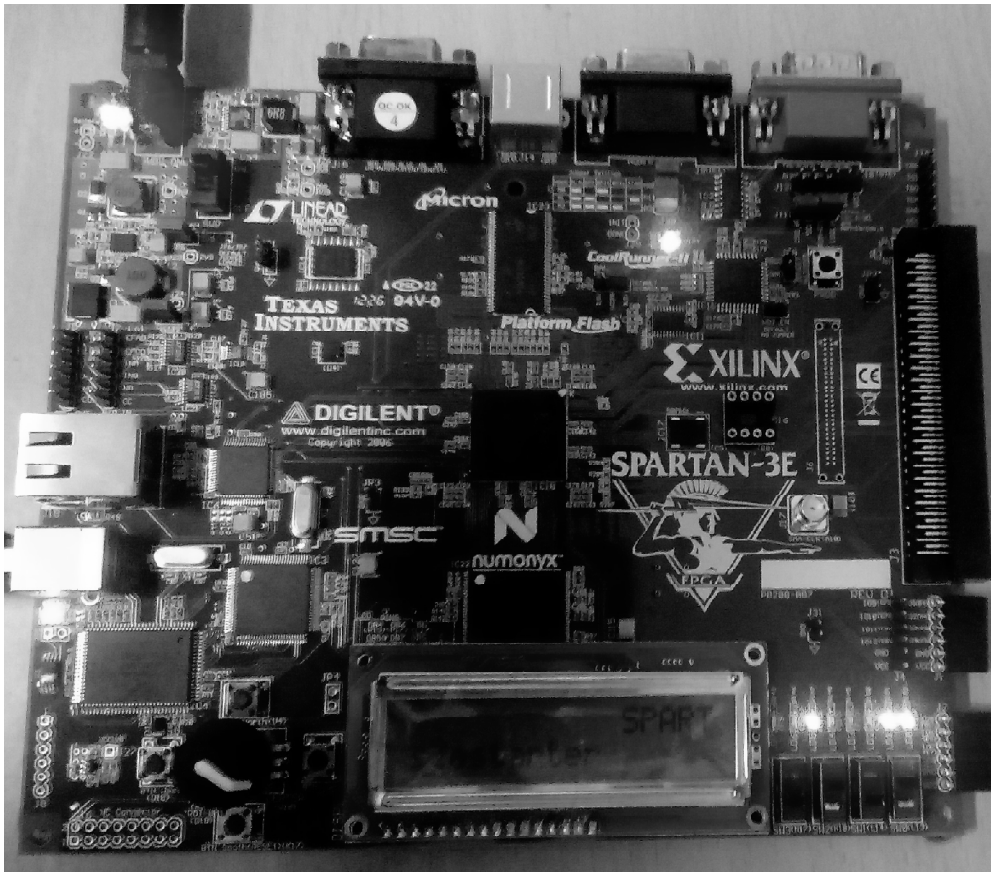

Figure 14: Multiplication and Accumulation Unit implemented in Spartan 3E

Table V
Report Analysis for the proposed MAC Unit of 4bit, 8bit, 16bit

MAC UNIT	*4BIT*	*8BIT*	*16BIT*
MINI PERIOD	4.065	5.352	6.388
MAXFREQUENCY	246.027	186.86	156.553
PATH DELAY(ns)	12.615	20.027	37.603
NO OF SLICES	33	135	511
NO OF SLICE FF's	8	16	32
NO OF 4IP'S LUTS	60	243	905
NO OF IOB'S	19	35	66
GENERAL CLK	1	1	1

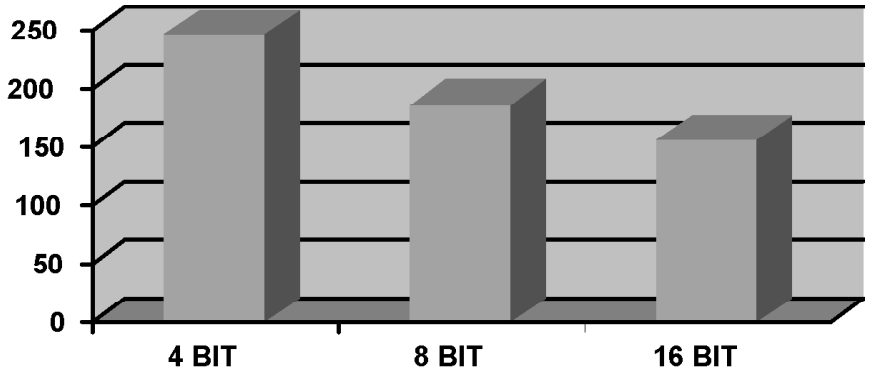

Figure 15: Maximum Frequency of N-Bit MAC UNIT

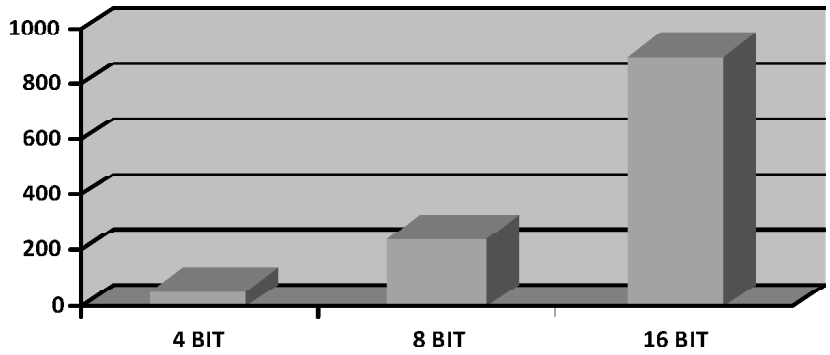

Figure 16: Number of LUT'S in N-Bit MAC UNIT

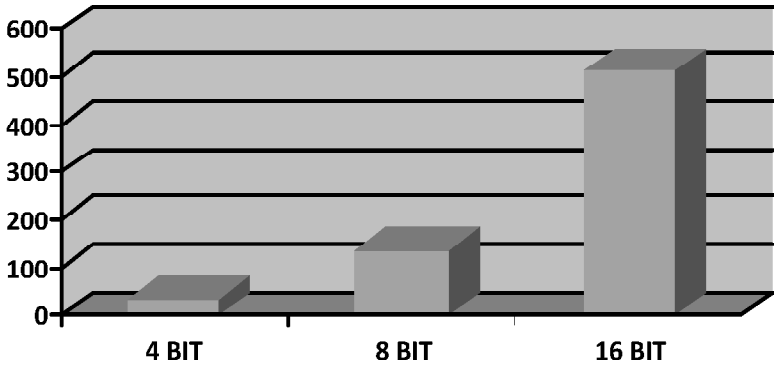

Figure 17: Number of Slices of N-Bit MAC UNIT

7. CONCLUSION

A MAC unit is implemented with Vedic multiplier unit with Parallel prefix adders and conventional adders.MAC unit with Parallel prefix adders has achieved less delay and operated with the maximum speed of 246.027MHz for 4bit, for 8bit 186.86MHz, for 16bit 156.553MHz.

REFERENCES

[1] Jagadguru Swami, Sri Bharati Krishna and Tirthaji Maharaja, "Vedic Mathematics or Sixteen Simple Mathematical Formulae from the Veda, Delhi (1965)", Motilal Banarsidass, Varanasi, India.

[2] M. Morris Mano "Computer System Architecture", 3rd edition, Prentice-Hall, New Jersey, USA, 1993, pp. 346-348.

[3] Cory Merkel, David Brenner, "8-bit Parallel Prefix Adders Using Brent-Kung Tre BIST", EECC730, November 2008.

[4] Andrew Beaumont-Smith and Cheng-Chew Lim, "Parallel Prefix Adder Design", Department of Electrical and Electronic Engineering, the University of Adelaide, 5005, Australia, 2001.

[5] K. Vitoroulis and A. J. Al-Khalili, "Performance of Parallel Prefix Adders Implemented with FPGA technology," IEEE Northeast Workshop on Circuits and Systems, pp. 498-501, Aug. 2007.

[6] Kihak Shin, Ik Kyun Oh, Sang MinBeom Seon Ryu, Kie Young Lee And Tae Won Cho " A Multi-Level Approach to Low Power Mac Design" IEEE Trans. VLSI systems, vol. 48, pp 361- 763, July 1999.

[7] S. Xing and W. W. H. Yu, "FPGA Adders: Performance Evaluation and Optimal Design," IEEE Design & Test of Computers, vol. 15, no. 1, pp. 24-29, Jan. 1998.

[8] T.T Hoang, M. Sjolander, "Double Throughput Multiply Accumulate Unit for Flexcore Processor Enhancements" IEEE International Symposium on Parallel & Distributed Processing, pp1-4, 2009.

[9] M.E.Paramasivam,Dr.Rabeenian"An Efficient Bit Reduction Binary Multiplication Algorithm using Vedic Methods, IEEE2nd International Advance Computing Conference 2010.

[10] Anvesh Kumar, Ashish Raman ,Dr. R.K. Sarin, Dr.Arun Khosla, Small area Reconfigurable FFT Design by Vedic Mathematics, 2010 IEEE.