DESIGN AND SIMULATION OF DIFFERENT TYPES OF MULTIPLIERS USING VHDL CODE

[1]Raghuveer Patel, [2]Dr. Namit Gupta
[1]Mtech. Student, [2]Professor
Electronics & Instrumentation Department
Shri Vaishnav Vidyapeeth Viswavidyalaya
Indore (MP), India

Abstract: ***This Paper is devoted for the design and analysis of high speed multiplier. In this paper I have used three different techniques to Design high speed multiplier. In the first technique Array multiplier is designed to perform 8*8 multiplication, in second technique Wallancetree multiplier is designed to perform 8*8 multiplication and in the last technique Vedic mathematics is used to design an 8*8 multiplier. VHDL programs of all this multipliers are synthesized using Xilinx 9.2i software and simulated using Xilinx ISE simulator.***

The comparative study of different multipliers is done for finding the efficient multiplier design to perform high speed operation of multiplication and it has been found that the Vedic multiplier is more efficient than Array and Wallancetree multiplier because it gives minimum delay for multiplication. The designed Vedic multiplier is based on "Urdhva-tiryakbhyam" algorithm or sutra of ancient Indian Vedic mathematics. It is one of the sixteen sutras of Vedic mathematics. Vedic mathematics sutra "Urdhva-tiryakbhyam" increases the speed of multiplier by reducing the number of partial products. Hence the speed of overall system or processor can be increased by designing high speed Vedic multiplier.

Keywords - VHDL, Multiplier, Array multiplier, Wallancetree multiplier, Vedic multiplier, Urdhva-tiryakbhyam sutra, SOP, POS, ISE

I. INTRODUCTION

We are living in the era of Technology and Science, as we see that every day some new technology is discovering with greater benefits and efficiency. Scientists and Researchers are continuously working on the fact that how we can update the current Technology to get more benefits in less effort. In past decade the work of man is continuously replaced by Computer or Machine. The Computer is an electronic machine and it processes the data according to the need of user. The man benefit of computer or controller is that it has better efficiency and also it takes less time to process the data. The heart of any computer or controller is CPU (Central Processing Unit) which has ALU (Arithmetic and Logical Unit). ALU is responsible to perform logical and arithmetic operations on digital or binary data, which is represented by '0' & '1'.

The time consumed by CPU to perform logical and arithmetic operations is most important factor and it should be minimum as possible. As a result of expanding computer and signal processing applications the requirement for high speed processing system has been increasing. In this paper I am going to explain some techniques which take less time to multiply two binary numbers.

In arithmetic operation Multiplication is an important fundamental function. The operations which are based on Multiplication takes more time than addition and subtraction, hence if we increase the speed of multiplication operation then we can also increase the speed of calculations of operations which are based on Multiplication. There are so many operations which are based on multiplication like Multiply and Accumulate (MAC) and computation intensive arithmetic function (CIAF) , convolution, FFT, filtering and in microprocessor & microcontroller's arithmetic and logical unit. There is always a need of high speed multiplier because it dominates the execution time of most DSP algorithm. The application area of computer is expanding day by day, so the need of high speed multiplier to increase the speed of the whole system is basic requirement of the system. [1]

In Vedic times when there are no calculators and computers were present to calculate the data values, the ancient Guru's were used some techniques or methods for their calculations, now that techniques are known as Vedic calculations. Vedic techniques are very fast and logical. The Vedic mathematics approach is totally different and considered very close to the way a human mind thinks and works.

It is also proved by many publications that the Vedic algorithms are very useful in designing digital multiplier to reduce calculation time or to increase the speed of multiplier.. Vedic multiplier makes the calculation of data very fast as compared to simple multiplier for complex multiplication. The conventional multiplication method generates partial products to produce output, in the technique of Vedic multiplier the number of partial products generated is less, hence the speed is high. [2]

For any multiplier circuit or system, the basic requirements are it should consume less power and contain smaller area. So optimizing the area and speed of multiplier is a major task. The multiplier is a slowest component in any system; hence if we decrease the calculation time of multiplier then we can increase the performance of overall system. As we know the number of partial products is less in Vedic multiplier than the power requirement of the system is also less as compare to conventional multiplier. There are different types of multipliers are available depending upon the arrangement of components used during the design of the multiplier. According to power, area and speed we can choose the best architecture for our application.

The most widely used operation in arithmetic computation is multiplication and there are different types of multipliers are available depending on the speed and hardware. In this paper I am going to use three techniques for multiplication and at the last I will compare this three technique's results to show the fastest method of multiplication of 8 bit multiplier to 8 bit multiplicand. Array multiplier, Wallancetree multiplier and Vedic multiplier are the three techniques which are used in this paper. Vedic mathematics is the name given to the ancient system of mathematics which is based on some unique techniques of calculations. It uses 16 sutras to perform mathematical calculations on numbers. The sutra Urdhva tiryagbhyam is used to design high speed multiplier and the project work proved that the efficiency of this sutra. [2]

II. VEDIC MULTIPLIER

The term Vedic Mathematics comes from VEDAS. According to Hindu Methodology the Vedas are a collection of hymes and other ancient religious texts written in India. The Vedas includes poems, prayers, formulas, liturgical material and mythological accounts. There are four Vedas are present, and they are as follows:

- Rig-Vedic: "Knowledge of the songs of praise of god", for Recitation
- Sama-Veda: "Knowledge of the Music", for Chanting
- Yajur-Veda: "Knowledge of the prose Mantras", for worship
- Atharva-Veda: "Knowledge of the Magic formulas" for Mathematical Calculation. [5]

The origin of Vedic Mathematics is Athrva Veda and this collection of magic formulas is rediscovered by Shri Bharthi Krishna Tirthaji Maharaj in between 1911 to 1918. The name given to this set of formulas (Vedic Mathematics) is based on their origin i.e. Vedas. According to Tirthaji Maharaj, the Vedic mathematics consists of sixteen formulas, which are intended to describe the way the mind naturally works. Complex arithmetic problems and difficult calculations can often be solved immediately by the use of these sixteen formulas. Many Research work is being carried out on Vedic mathematics sutra's applications in mathematical calculations and it has proved that this formulas are very helpful to increase the speed of processor, which is responsible for mathematical calculations. [3] [4]

The applications of Vedic mathematics formulas and sutras almost cover all the branches of mathematics. The mathematical calculations like Addition, Subtraction, Multiplication, Division, Squaring, Cubing etc. of complex numbers can be made very easy and interesting with the help of Vedic mathematics formulas because this formula uses mental ability and pattern to perform calculation. So it is the power of Vedic mathematics that it can converts a tedious subject into a playful and blissful one.

All the sixteen sutras rediscovered by Tirthaji Maharaj from Vedas are listed below along with their meanings:

- Nikhilam Navtascaramam Dastah : All from nine and last from ten
- Ekadhikina Purvena : By one more than the previous one
- Urdhva Tiryagbhyam : Vertically and crosswise
- Paravartya Yojayet : Transpose and adjust
- Sunyam Samyasamuccye : When the sum is the same, that sum is zero
- Sunyanmanyat or Anurupye : If one is in ratio, the other one is zero
- Sankalana Vyavakalanabhyam : By subtraction and addition
- Puranapuranabhyam : By non-completion or completion
- Calana kalanabhyam : Similarities and non similatities
- Yavadunam : Whatever the extent of its deficiency
- Vyastisamastih : Use the averages
- Sesanyankena Caramena : The remainders by the penultimate
- Sopantyadvayamantyam : The ultimate and twice the penultimate
- Ekanyunena Purvena : By one less than the previous
- Gunitasamuccayah : The POS is equal to the SOP
- Gunakasamuccayah : The factors of the sum is equal to the sum of factors

The above mentioned sixteen sutras are covers almost every branch of mathematics. Applications of these sutras in mathematical calculation save a lot of time and effort. There are so many sub-sutras are also discovered but are not discussed here. From above mentioned sixteen sutras I am going to use only Urdhva-tiryakbhyam sutra to design Vedic multiplier. According to the pattern used in multiplication by Urdhva-tiryagbhyam sutra, it is also known as Vertically and Crosswise. Out of sixteen sutras presented above five sutras can be used to perform multiplication operation namely Nilhilam sutra, Anurupyena sutra, Urdhva-tiryakbhyam sutra, Ekanyunena Purvena sutra and Antyayoreva sub-sutra. All the sutras have their own application areas where they can be used to perform calculation in less time.

III. METHODOLOGY

3.1 VEDIC MULTIPLIER

Urdhva-tiryakbhyam sutra is General method of multiplication in Vedic mathematics which takes less time to perform multiplication operation to multiply any types of numbers. Nikhilam sutra of Vedic mathematics is applicable in some special cases of multiplication, but Urdhva tiryagbhyam sutra in applicable to do all cases of multiplication. The pattern used in Urdhva tiryagbhyam sutra is Vertically and Cross-wise.

The application of Urdhva-tiryagbhyam sutra for multiplying two numbers consisting 2 digit, 4 digit and 8 digits are described as follows:

MULTIPLICATION OF TWO-DIGIT NUMBER:

Formula Used: (x=10)

First Number => A_2A_1

Second Number => B_2B_1

$$(A_2x + A_1)(B_2x + B_1) = A_2B_2x^2 + (A_2B_1 + A_1B_2)x + A_1B_1$$

Process (Right to Left):

- Vertical Multiplication of last digits of both the numbers.
- Crosswise Multiplication of digits of both numbers and adding them.
- Vertical Multiplication of First digits of both numbers

MULTIPLICATION OF FOUR-DIGIT NUMBER:

Formula Used: (x=10)

First number => $A_4A_3A_2A_1$

Second number => $B_4B_3B_2B_1$

$$(A_4x^3 + A_3x^2 + A_2x + A_1)(B_4x^3 + B_3x^2 + B_2x + B_1)$$
$$= A_4B_4x^6 + (A_4B_3 + A_3B_4)x^5 + (A_4B_2 + A_3B_3 + A_2B_4)x^4 + (A_4B_1 + A_3B_2 + A_2B_3 + A_1B_4)x^3$$
$$+ (A_3B_1 + A_2B_2 + A_1B_3)x^2 + (A_2B_1 + A_1B_2)x + A_1B_1$$

3.2 WALLANCETREE MUTIPLIER

The Wallance tree Multiplier is a fast way to multiply two binary numbers. It is suggested by C. S. Wallance as a fast multiplier during 1964. Multiplier designed by C. S. Wallance is composed of combination of Half adder and Full adder. Any multiplier basically consists of two operands, a Multiplicand and a Multiplier, both the operands together produces the output product. [1] [6]

The multiplication by Wallance tree has three steps as follows:

1. Partial Product Generation: In this step we perform multiplication of each bit of multiplier by each bit of multiplicand to generate partial products. Basically AND gate is employ to produce partial products.
2. Partial Products Addition: In this step the number of partial products is reduces to two by layers of Half adder and Full adders.
3. Final Addition: Group the wires in two numbers and add them with the conventional adder.

The second step is most important in Wallance tree multiplier because the overall speed of multiplier is mainly depends on this step. The procedure of second step is as follows, as long as there are more than three wires with same weight add them as follows:

- Take any three wires with the same weight and add them with the help of Full adder, means they are works as inputs of Full adder.
- If there are only two wires of same weight left than add them with the help of Half adder, means they are works as the inputs of Half adder.
- If there is just one wire left, than connect it to the next layer means it copied as it is.

The general block diagram of tree based multipliers is shown in figure below:

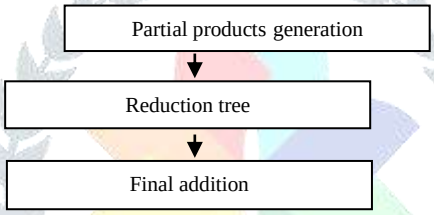

Figure 1: Architecture of Wallance tree

3.3 ARRAY MULTIPLIER

In Conventional multiplication method each digit of the multiplier is multiplied with the multiplicand and results are taken which are called Partial Products, than the second digit of multiplier is taken and multiplication of this digit is again performed with the same multiplicand and results are written down to the first partial product. But when we do that we do the shifting because we have taken the second position of the multiplier, after getting all the partial products, finally we add them to get the final product output. This conventional algorithm is called shift and add algorithm because we have to add it after shifting it.

Array Multiplier is a digital circuit used for multiplication of two binary numbers. It is composed of an array of Half adders and Full adders. Because of array used in this multiplier of half adders and full adders, it is known as array multiplier. Array multiplier is well known due to its regular structure. Array multiplier circuit is based on the principal of adding and shifting. The partial products are generated by the multiplication of the multiplicand with each bit of multiplier with the help of AND gate. The partial products are shifted according to their bit orders and then added to get final product.

The algorithm of array multiplier is explained below with an example:

123	Multiplicand
× 245	Multiplier
615	(this is 123×5)
492	(this is 123×4, shifted one position to the left)
246	(this is 123×2, shifted two position to the left
+30135	(addition of partial products)

IV. IMPLEMENTATION AND SYNTHESIS

The proposed 8-bit multipliers are designed using VHDL coding using three different architectures namely Array multiplier, Wallancetree multiplier and Vedic multiplier using Urdhva-tiryagbhyam sutra. It is evident that the speed of Vedic multiplier is higher than the other multipliers. The simulation results for Vedic, Wallancetree and Array multipliers are as follows:

4.1 ARRAY MULTIPLIER SYNTHESIS RESULTS

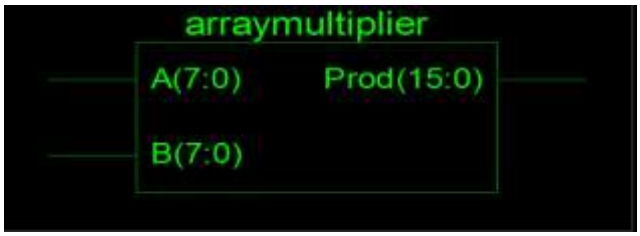

Figure 2: Top view of Array multiplier

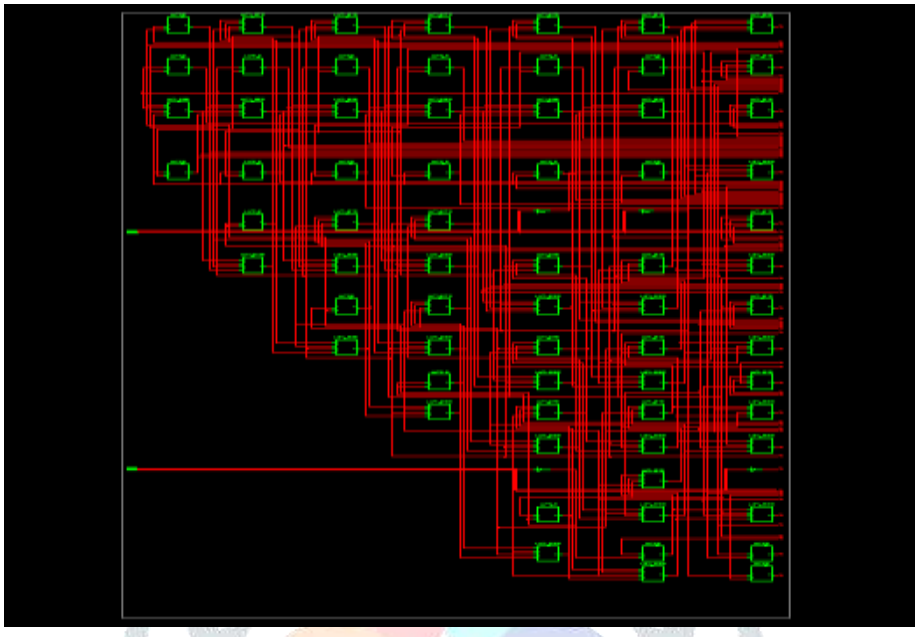

Figure 3: RTL view of Array Multiplier

4.2 WALLANCE MULTIPLIER SYNTHESIS RESULTS

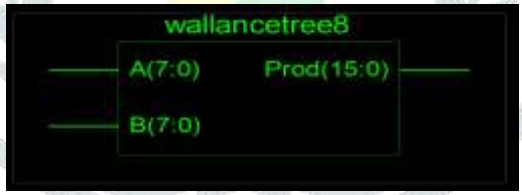

Figure 4: Top view of Wallance Multiplier

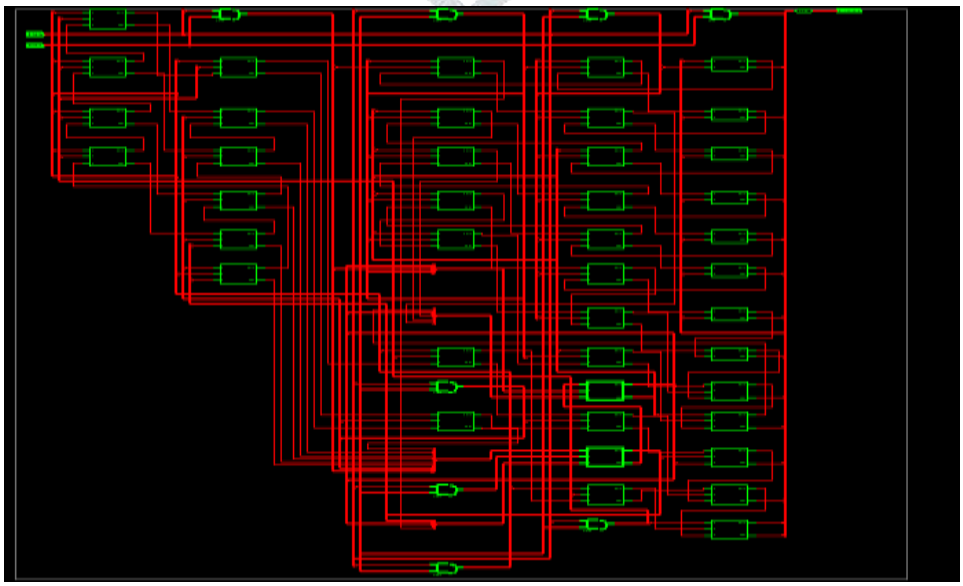

Figure 5: RTL view of Wallance Multiplier

4.3 VEDIC MULTIPLIER SYNTHESIS RESULTS

Figure 6 : Top view of Vedic multiplier

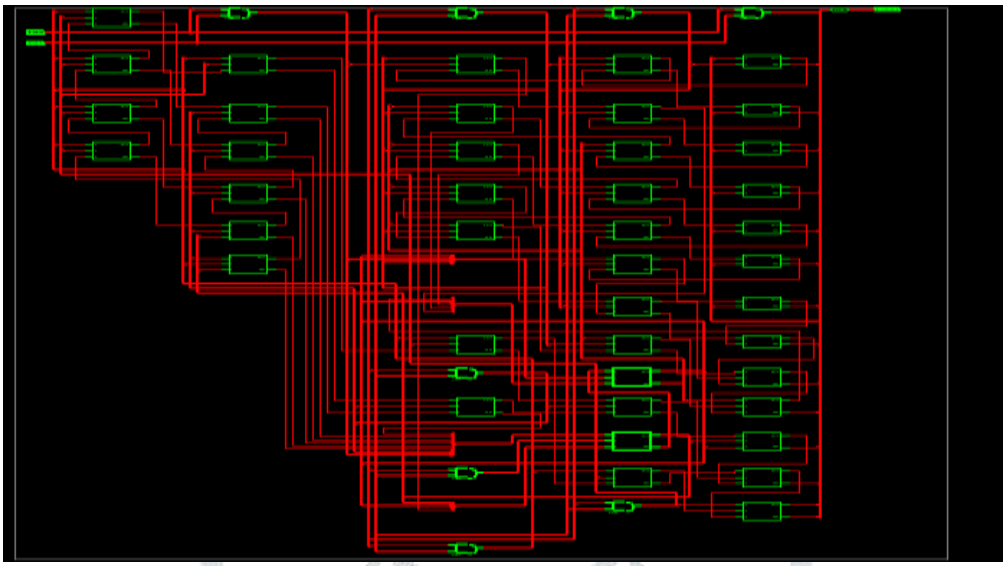

Figure 7: RTL view of Vedic Multiplier

4.4 SYNTHESIS REPORT:

Number of slices	72 out of 1920
Number of 4 input LUTs	126 out of 3840
Number of IOs	32
Number of bounded IOBs	32 out of 173
Maximum combinational path delay	36.406 ns
Total memory used	224948 kb

Table 1: Design summary of Array Multiplier

Number of slices	76 out of 1920
Number of 4 input LUTs	132 out of 3840
Number of IOs	32
Number of bounded IOBs	32 out of 173
Maximum combinational path delay	34.014 ns
Total memory used	177808 kb

Table 2: Design summary of Wallance Multiplier

Number of slices	91 out of 1920
Number of 4 input LUTs	159 out of 3840
Number of IOs	32
Number of bounded IOBs	32 out of 173
Maximum combinational path delay	28.351 ns
Total memory used	177852 kb

Table 3: Design summary of Vedic Multiplier

V. SIMULATION RESULTS

The input and output test bench waveforms of all multipliers are:

5.1 Array Multiplier simulation results:

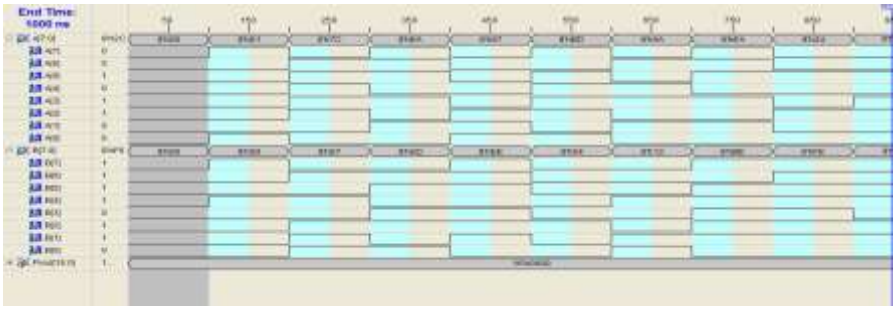

Figure 8: Input waveforms of Array Multiplier

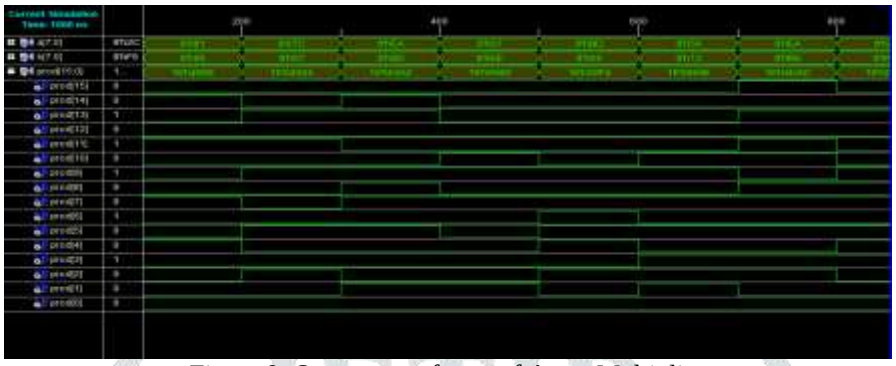

Figure 9: Output waveforms of Array Multiplier

5.2 Wallance-tree Multiplier simulation results:

Figure 10: Input waveforms of Wallance Multiplier

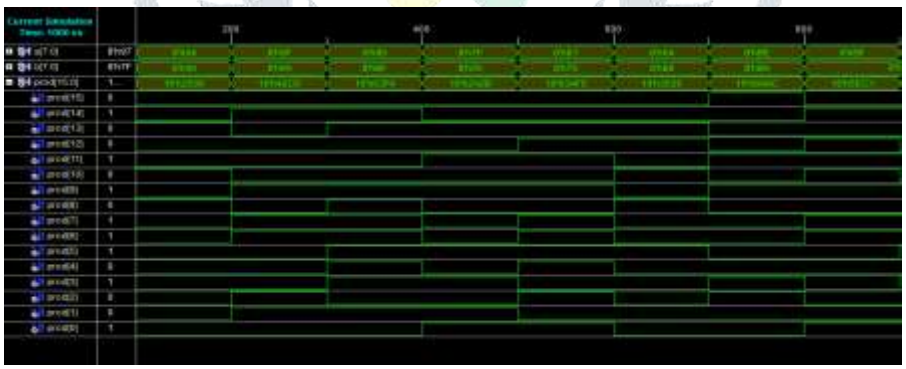

Figure 11: Output waveforms of Wallance Multiplier

5.3 Vedic Multiplier simulation results:

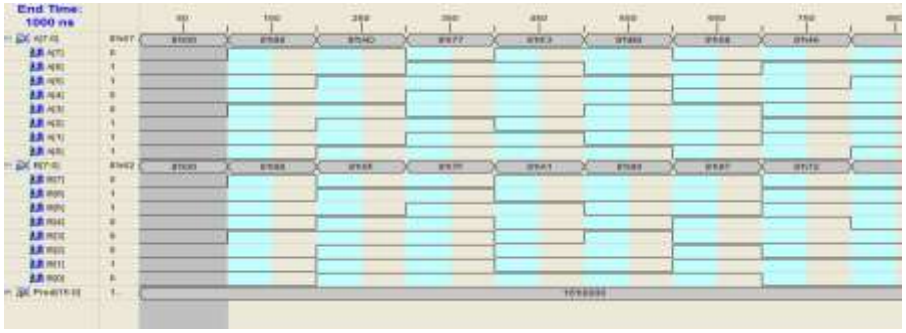

Figure 12: Input waveforms of Vedic Multiplier

Figure 13: Output waveforms of Vedic Multiplier

VI. CONCLUSION

The designs of 8*8 multipliers have been successfully simulated using Xilinx 9.2i software for array multiplier, Wallance multiplier and Vedic multiplier. The Vedic multiplier is based on Urdhvatiryagbhyam sutra of Vedic Mathematics. It is shown that the maximum combinational path delay in case of Vedic multipliers is 28.351 ns, which is less as compare to Array and Wallance multiplier. It is therefore proved that the Vedic multiplier is much faster than the Array and Wallance multipliers.

REFERENCES

[1] C.S. Wallance, "A suggestion for a fast multiplier," IEEE Transactions on Electronic Computers, volume: EC-13, Issue: 1, pages:. 14-17, Feb. 1964, Print ISSN: 0367-7508

[2] M. Ramalatha, K. Deena Dalayan, P. Dharani, S. Deborah Priya, "High Speed energy efficient ALU design using Vedic multiplication technique" IEEE International conference on Advances in computational tools for Eng. Applications. Pages: 600-603, DOI: 10.1109/ACTEA.2009.5227842, July 2009.

[3] G. Ganesh Kumar, V. Charishma "Design of high speed vedic multiplier using vedic mathematics techniques" International journal of scientific and research publications, Volume 2, Issue 3, march 2012 ISSN: 2250-3153

[4] Jagadguru Swami Sri Bharath, Krisna Tirathji, "Vedic mathematics or Sixteen simple sutras from the Vedas", Motilal Banarasidas, Varanasi (India), 1986.

[5] Cristian Violatti "The Vedas", Ancient history Encyclopedia. Retrieved from https://www.ancient.eu/The_Vedas/ (January 2013)

[6] Jasbir Kaur, kavita, "structural VHDL implementation of Wallance tree multiplier," International Journal of Scientific & Engineering Research, Vol: 4, Issue: 4, April-2013, ISSN: 2229-5518