HIGH SPEED SIXTY-FOUR BIT VEDIC MULTIPLIER

Musunuri Chinnikavya [1], M.Tech Scholar, Eluru college of Engineering and technology, JNTUK,AP

K. Ravi Kumar [2], M.Tech, Assistant Professor, Eluru college of Engineering and technology, JNTUK,AP

ABSTRACT

Multipliers are vital components of any processor or computing machine. Often, performance of microcontrollers and Digital signal processors are evaluated based on number of multiplications performed in unit time. So better repetitive structures will increase system efficiency. Vedic multiplier is one of the most promising solutions. Its simple structure combined with increasing speed creates an incomparable combination to supply any complex multiplication. Marked with these excellent images, using this in a repetitive sense further reduces energy depletion Energy depletion is another important obstacle to a neglected embedded system.

In this paper we publish a Vedic repetition known as "Urdhva Tiryakbhayam" which means vertical and horizontal, applied using a retracted mind, which is the first of its kind. This reporter can find applications in Fast Fourier Transforms (FFTs), and other DSP programs such as imaging, software-defined radios, wireless communication. For arithmetic multiplication, various Vedic multiplication techniques like Urdhva tiryakbhyam, Nikhilam and Anurupye has been thoroughly discussed. It has been found that Urdhva tiryakbhyam Sutra is most efficient Sutra (Algorithm), giving minimum delay for multiplication of all types of numbers, either small or large.

Further, the Verilog HDL coding of Urdhva tiryakbhyam Sutra for 64X64 bits multiplication and their FPGA implementation.

1. INTRODUCTION

The most important fundamental function in arithmetic operations is multiplication. Some of the frequently used Computation- Intensive Arithmetic Functions(CIAF) are Multiplication-based operations such as Multiply and Accumulate(MAC) and inner product. These are presently used in many Digital Signal Processing (DSP) applications such as Fast Fourier Transform(FFT), convolution, filtering and in microprocessors in its arithmetic and logic unit. There is a need of high speed multiplier since multiplication dominates the execution time of most DSP algorithms. At present the instruction cycle time of a DSP chip determination depends on multiplication time and this time is still a dominant factor.

The demand for high speed processing has been increasing as a result of expanding computer and signal processing applications. High arithmetic performance is essential to achieve the desired functionality in many real-

time signals and image processing applications. One of the key arithmetic tasks in such programs is repetition and the development of a rapid repetition cycle has become a topic of interest for decades. Reducing time delays and energy consumption are the most important requirements for many systems. This work introduces various architectures. Vedic Mathematics-based multiplication is one of the fastest and lowest repetitions.

Reducing energy consumption in digital systems involves full implementation at all design levels. This setting includes technologies used to use digital circuits, circuit style and topology, circuit implementation formats and the highest level of algorithms used. There are different types of multipliers available. Special repetition structures are selected based on application.

In most DSP algorithms, the multiplier lies in an important delay path and ultimately determines the operation of the algorithm. The speed of the duplicate function is very important for DSP and normal processors. In the past multiplication was often used in the sequence of addition, subtraction and replacement functions. There have been many suggestions for algorithms in the practice books, each offering different benefits and having a tradeoff depending on the speed, complexity of the circuit, location and power consumption.

A multiplier is a very large block of a computer system. In the repetition algorithms performed on DSP the application latency and output are two major concerns in terms of delay. The actual delay in computer performance, the measure of how long the input of the device is stable is the result of the output. Throughput is a measure of how many repetitions can be made over a period; duplication is not only a high delay but also a major source of energy depletion reduces energy consumption, there is great interest in reducing delays through various delay setting.

Digital duplicates are an integral part of all digital signal processors (DSPs) and the DSP speed is largely determined by the speed of its duplicates. Two common multiplication algorithms are followed by a digital hardware multiplication list algorithm and Booth multiplication algorithm. The calculation time taken by the same multiplier of the same members is relatively short because the proportion of products is calculated independently in relation to each other. Delays related to the multiplication of the same members are the time it takes for the signals to spread across the gates that form the repetition list. Booth Repetition is another important repetition algorithm.

The proposed multiplication algorithm is then shown to demonstrate its computer efficiency by taking the example of reducing 4X4-bit duplication to a single 2X2-bit duplicate

function. This work introduces a systematic design approach to multiplying digits quickly and efficiently based on Vedic calculations. Multiplier Architecture is based on the Vertical and Crosswise algorithm of ancient Indian Vedic Mathematics.

Producing an Automatic Pattern Test is a process of generating patterns to test a circuit that is strongly defined by a logical level network. They can generate circuit test vectors to detect errors in the circuit. They can detect unwanted or unwanted circuit logic and can verify whether a single circuit is being used. he functionality of Vedic multiplier using reversible logic gates and fundamental concepts of multiplications, architectures are going to be discussed in **chapter2.**

The hardware implementation of Vedic multiplier using reversible logic gates on FPGA using Xilinx tool is going to be conversed and the design flow of BIST is also clarified in **chapter 3**. In **chapter 4** results obtained from realization of Vedic multiplier using reversible logic gates on FPGA kit, in terms of speed, area and number of gates are exposed**.**

II IMPLEMENTATION AND TESTING OF PROPOSED MULTIPLIER

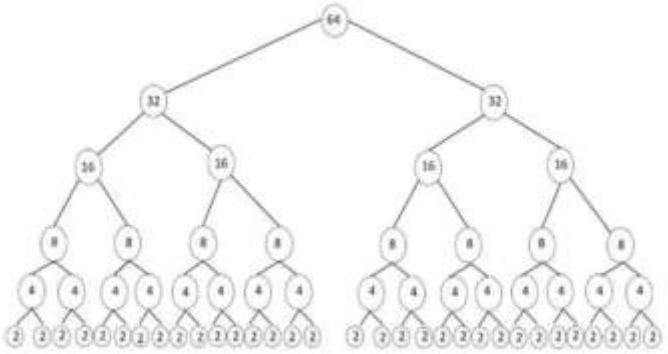

Figure 1 : Low Level Module

Initiation Each of the two operands which are of 64-bit wide are sliced into two 32-bit numbers, so that we perform two chunks of multiplication same as we do in bit multiplication. The 32-bit number is 2×2 further divided into two 16-bit numbers and the same method is repeated. Above figure shows the specified architecture of the 64-bit module which resembles the tree architecture. So, the delay of the multiplier will be in the order of (N)

DESIGN AND SOFTWARE SIMULATION

A novel technique of digital multiplication which is quite different from the conventional method of multiplication like add and shift is used for the designing of Vedic Multiplier. Verilog HDL is used for the design of Vedic Multiplier, since this gives effective utilization of structural method of modeling. Verilog hardware description language is used for the implementation of individual block. Model Sim and ISE are used for the verification of functionality of each block is using simulation software.

IMPLEMENTATION OF 2X2 BITS VEDIC

MULTIPLIER

One-bit multipliers and adders are the basic building blocks of this multiplier. Two input AND gate and for addition full adder can be utilized for performing the one-bit multiplication. Figure 3.2 shows the 2 x 2-bit multiplier.

Figure 2: Showing Block diagram of 2x2 Multiplier

As per basic method of multiplication, results are obtained after getting partial product and doing addition. A1 A0 X B1 B0 A1B0 A0B0 A1B1 A0B1 Q3 Q2 Q1 Q0

In Vedic method, Q0 is vertical product of bit A0 and B0, Q1 is addition of crosswise bit multiplication i.e. A1 & B0 and A0 and B1, and Q2 is again vertical product of bits A1 and B1 with the carry generated, if any, from the previous addition during Q1. Q3 output is nothing but carry generated during Q2 calculation. This module is known as 2x2 multiplier block.

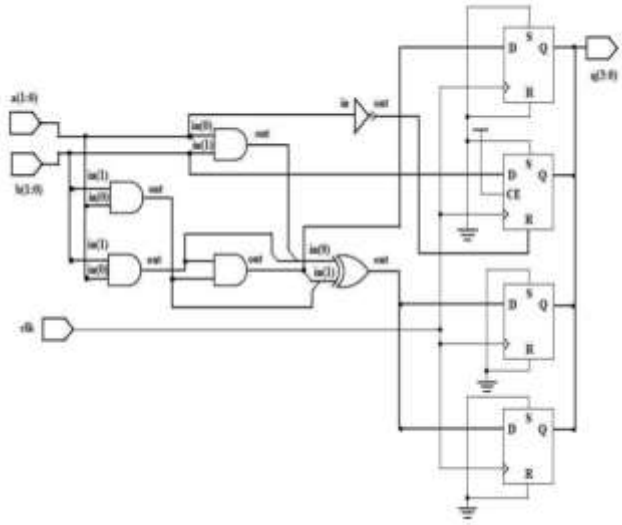

Figure 3: Showing RTL View of 2x2 Bits Multiplier by Model Sim

IMPLEMENTATION OF 4X4 BITS VEDIC MULTIPLIER

Little modification is required for higher no. of input bits. The input bits are divided into two equal no. of bits.

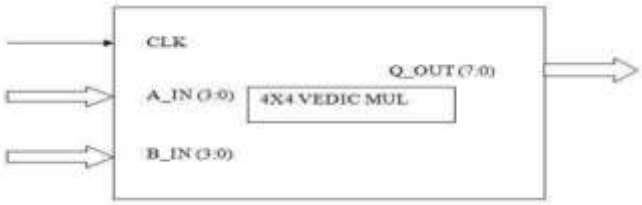

Figure 3: Showing Block diagram of 4x4 Bit Vedic Multiplier

Let us analyze 4x4 multiplications, say A3A2A1A0 and B3B2B1B0. The result of this 4X4 multiplication is Q7Q6Q5Q4Q3Q2Q1Q0.Fig 3 shows the block diagram of 4x4 Vedic Multiplier is given below. The fundamental of Vedic multiplication, taking two bits at a time and using 2

bit multiplier block, A3A2 A1A0 XB3B2 B1B0

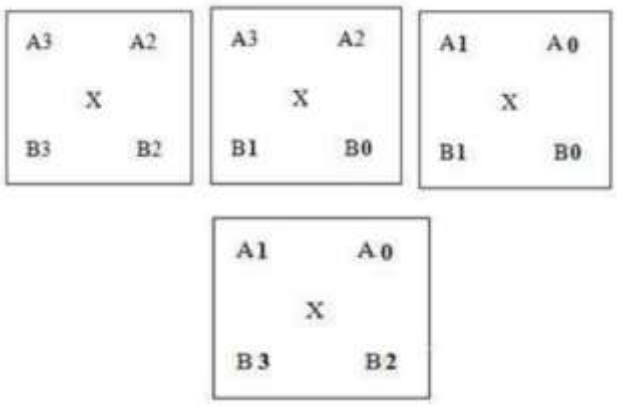

Figure 4: Showing Algorithm of 4x4 bit UT Multiplier.

2x2 bits multiplier is the basic building block as shown above is. A1 A0 and B1 B0 are the inputs for first 2x2 multiplier. A3 A2 and B3 B2 are the inputs for last block is 2x2 multiplier. A3A2 & B1B0 and A1A0 & B3B2 are the inputs for middle. Hencethe 8-bit answer Q7Q6Q5Q4Q3Q2Q1Q0 is the final result of multiplication.

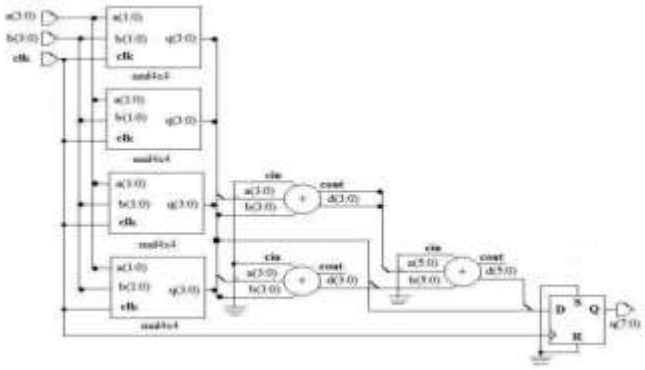

Figure 5: Showing RTL View of 4x4 Bit UT Multiplier by Model Sim 4.3

IMPLEMENTATION OF 8x8 BIT UT MULTIPLIER

The figure 6 shows the construction of 8x8 bit UT multiplier using 4X4 bit blocks using reversible logic gates. In this figure AH-AL are the two 4-bit pairs of the 8 bit multiplicand A. Similarly BH-BL are the two pairs of multiplicand B. The result of16 bit product can be written as: P= A x B= (AH-AL) x (BH-BL) = AH x BH+AH x BL + AL x BH+ AL x BL The final product can be obtained suitably by adding the outputs of 4X4 bit multipliers. Also two adders are required similarly in the final stage. Now 4x4 bits multiplier which is the basic building block of 8x8 bits UT multiplier is which implemented in its structural model. The fastest design of a circuit can be obtained by structural modeling.

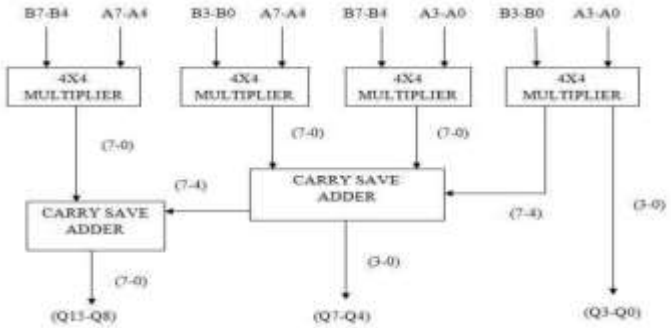

Figure 6: Showing 8x8 Bits decomposed Vedic Multiplier.

RESULT = (Q15- Q8) & (Q7- Q4) & (Q3-Q0)

IMPLEMENTATION OF 16x16 BIT UT MULTIPLIER

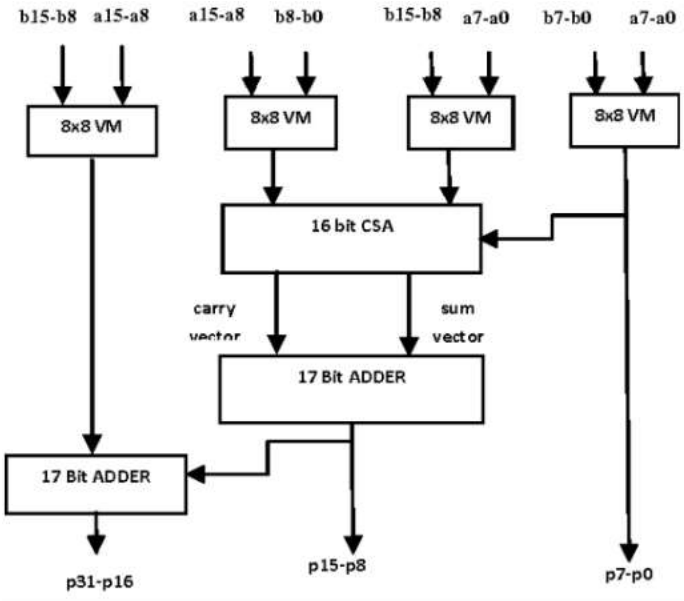

Figure 7: Showing 16X16 Bits decomposed Vedic Multiplier.

The Basic of 16x16 multiplier is four 8x8 bit vedic multiplier as shown in the figure. The operations are carried out using UT algorithm and we get an output of 32 bits as shown in the figure. Here we use the reversible logic gates in the place of adders to get output in more easy and simplified way.

IMPLEMENTATION OF 32x32 BIT UT MULTIPLIER

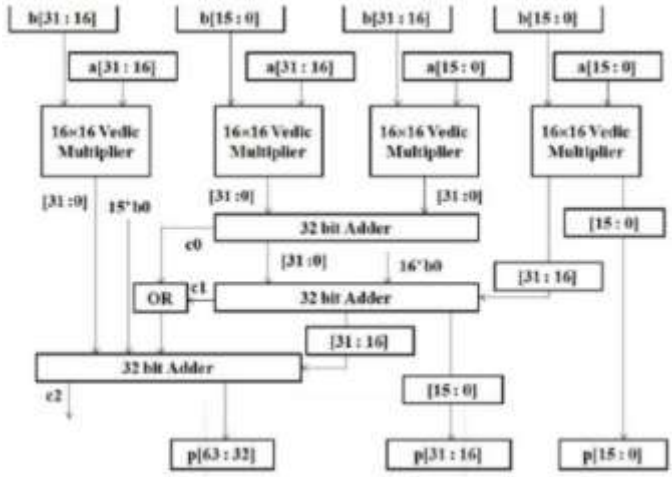

Figure 8: Showing 32x32 Bits decomposed Vedic Multiplier.

The Basic of 32x32 multiplier is four 16x16 bit vedic multiplier as shown in the figure. The operations are carried out using UT algorithm and we get an output of 64 bits as shown in the figure. Here we use the reversible logic gates in the place of adders to get output in more easy and simplified way.

IMPLEMENTATION OF 64x64 BIT UT MULTIPLIER

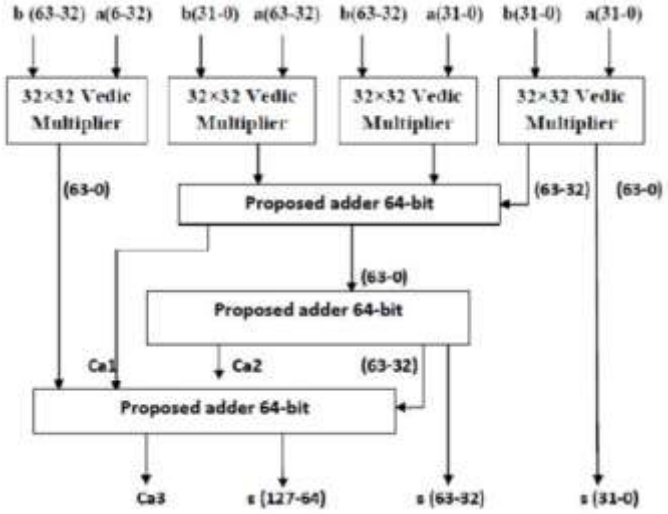

Figure 9: Showing 64x64 Bits decomposed Vedic Multiplier.

The Basic of 64x64 multiplier is four 32x32 bit vedic multiplier as shown in the figure. The operations are carried out using UT algorithm and we get an output of 128 bits as shown in the figure. Here we use the reversible logic gates in the place of adders to get output in more easy and simplified way. This proposed architecture works on a function

call method. The higher module calls the lower level module and so on and so forth**.**

IV XILINX SYNTHESIS AND OUTPUT SCREENS

Proposed Method Output Screens

Synthesis report of proposed Urdhva tiryakbhyam method

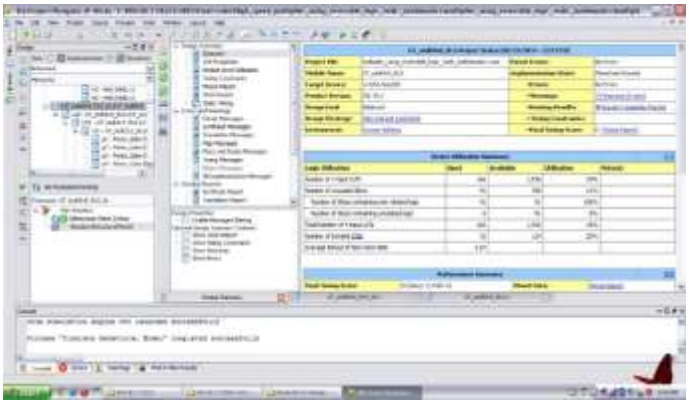

Figure 10: Showing Synthesis report

Top view of Proposed Urdhva tiryakbhyam System

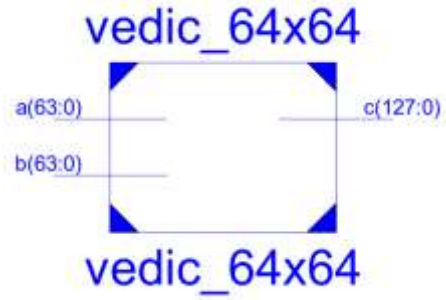

Figure 5: Showing Top view of system

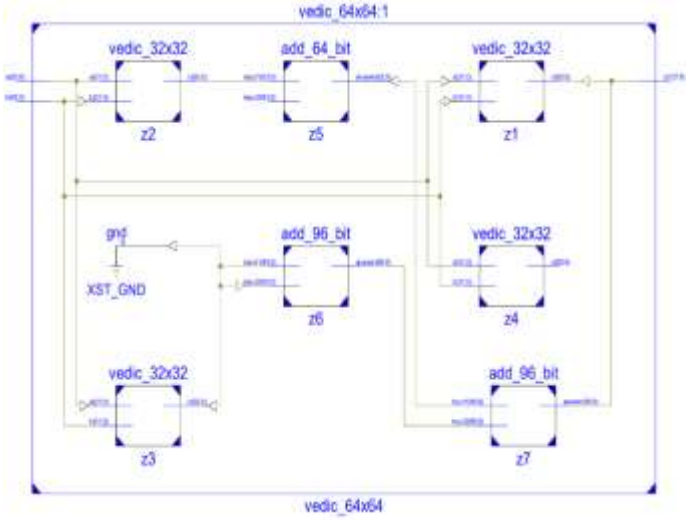

Figure 6: Showing RTL view of system

RTL View of Proposed Urdhva tiryakbhyam System

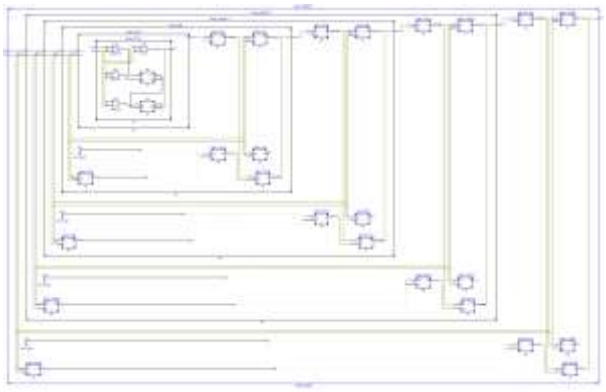

Figure 7: RTL Schematic View

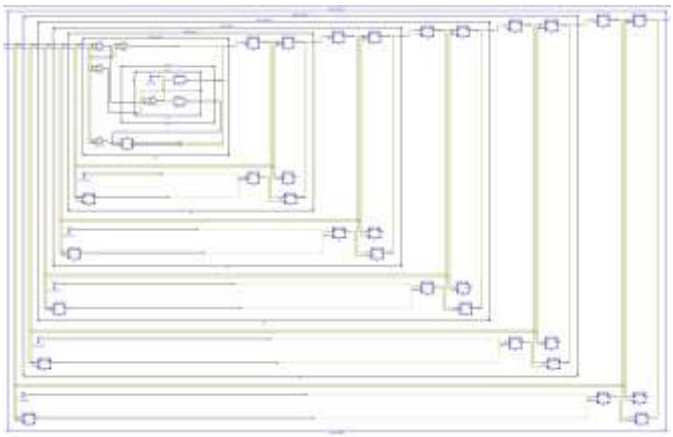

Figure 8: RTL Schematic View

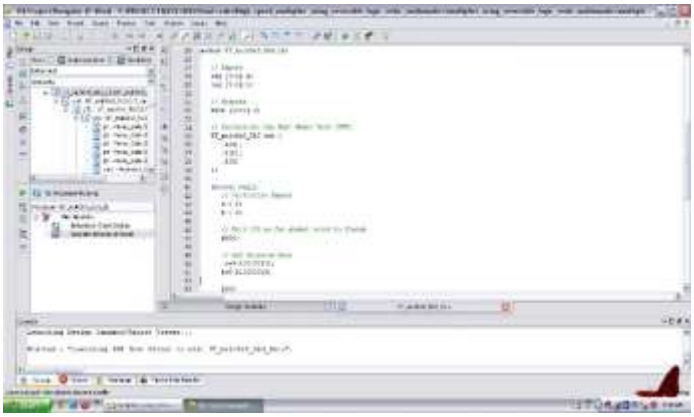

Figure 9: Test bench model of Proposed Urdhva tiryakbhyam method

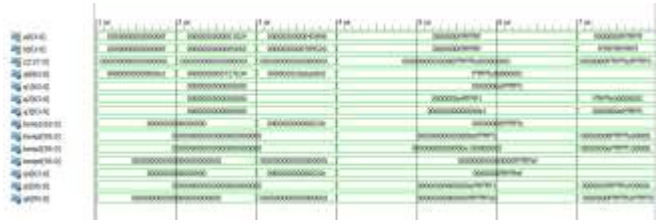

Figure 10: Showing Simulated output

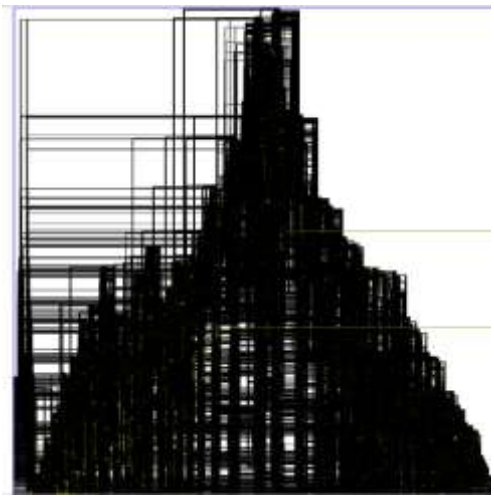

Figure 11: Time delay required for a system

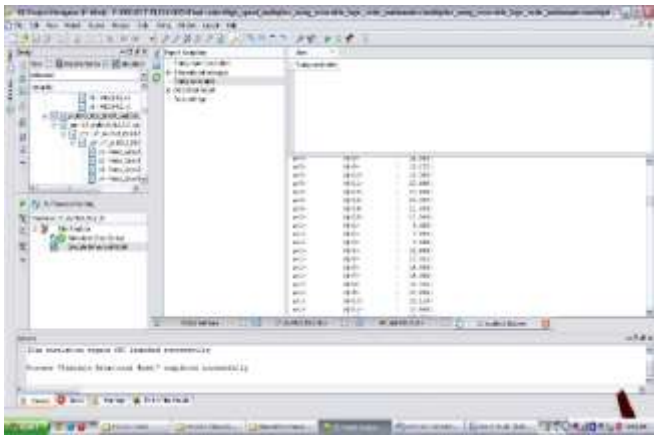

Figure 12: Showing Time Delay of system

Release 14.7 - xst P.20131013 (nt64)

Copyright (c) 1995-2013 Xilinx, Inc. All rights reserved.

--> Parameter TMPDIR set to xst/projnav.tmp

Total REAL time to Xst completion: 0.00 secs

* Low Level Synthesis *

Optimizing unit <vedic_64x64> ...

Optimizing unit <add_16_bit> ...

Optimizing unit <add_48_bit> ...

Optimizing unit <add_64_bit> ...

Mapping all equations...

Building and optimizing final netlist ...

Found area constraint ratio of 100 (+ 5) on block vedic_64x64, actual ratio is 25.

==========

* Design Summary *

==========

Top Level Output File Name : vedic_64x64.ngc

Primitive and Black Box Usage:

BELS : 9812

GND : 1

LUT2 : 384

LUT3 : 511
LUT4 : 2033
LUT5 : 3197
LUT6 : 3686
IO Buffers : 256
IBUF : 128
OBUF : 128

Device utilization summary:

Number of LUT Flip Flop pairs used: 9811

Number with an unused Flip Flop: 9811 out of 9811 100%

IO Utilization:

Number of IOs: 256

Number of bonded IOBs:256 out of 210 121% (*)

Timing Summary:

Speed Grade: -3

Maximum combinational path delay: 39.945ns

Timing Details:

Total REAL time to Xst completion: 46.00 secs

Total CPU time to Xst completion: 45.71 secs

Total memory usage is 4805144 kilobyte

Number of errors : 0 (0 filtered)

Number of warnings : 12 (0 filtered)

Number of infos : 11 (0 filtered)

Time Delay comparison table:

	Existing VM	Proposed VM
Path Delay	40.624	39.94
Levels of Logic	73	71

Table 2: Time delay comparison table

V. CONCLUSION

The designs of 64X64 bits Vedic multipliers using reversible gates and normal gates have been implemented on Spartan Xc3s100e-5vq100. The computation delay for 64X64 bits Wallace multiplier was 25.712 ns and for 64X64 bits Vedic multiplier using normal gates was 40.624 ns, and for Vedic multiplier using reversible logic gates was 39.94 ns

It is therefore seen that the Vedic multiplier using reversible logic gates is much faster than the conventional Wallace and Vedic multiplier using normal gates multipliers. The algorithms of Vedic mathematics are much more efficient than of conventional mathematics.

Urdhva tiryakbhyam sutram, Nikhilam sutram and Anurupyena sutram are the important among such vedic algorithms which can reduce the delay, power and hardware requirements for multiplication of numbers.The hardware realization of the Vedic mathematics algorithms is easily possible through the FPGA implementation of these multipliers.

FUTURE SCOPE

A clue of symmetric computation was developed around 2500 years ago which is brought out by Vedic mathematics. trigonometry, calculus, geometry and basic arithmetic are dealt by this symmetric computation. All these methods are very powerful as far as manual calculations are considered.

The computational speed drastically reduces if all those methods are effectively used for the

hardware implementation. Hence there is a chance for implementing a complete ALU using Vedic mathematics methods. Vedic mathematics is long been known but has not been implemented in the DSP and ADSP processors employing large number of multiplications in calculating the various transforms like FFTs and the IFFTs. By using these ancient Indian Vedic mathematics methods world can achieve new heights of performance and quality for the cutting edge technology devices.

REFERENCES

1. Swami Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics", MotilalBanarsidass Publishers, 1965.

2. Rakshith T R and RakshithSaligram, "Design of High Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach", International Conference on Circuits, Power and Computing Technologies (ICCPCT-2013), ISBN: 978-1-4673-4922-2/13, pp.775-781.

3. M.E. Paramasivam and Dr. R.S. Sabeenian, "An Efficient Bit Reduction BinaryMultiplication Algorithm using Vedic Methods", IEEE 2nd International Advance Computing Conference, 2010, ISBN: 978-1-4244-4791-6/10, pp. 25-28.

4. Sushma R. Huddar, Sudhir Rao Rupanagudi, Kalpana M and Surabhi Mohan, "Novel High Speed Vedic Mathematics Multiplier using Compressors", International Multi conference on Automation, Computing, Communication, Control and Compressed Sensing(iMac4s), 22-23 March 2013, Kottayam, ISBN: 978-1-4673-5090-7/13, pp.465- 469.

5. L. Sriraman and T. N. Prabakar, "Design and Implementation of Two Variables Multiplier Using KCM and Vedic Mathematics", 1st International Conference on Recent Advances in Information Technology (RAIT -2012), ISBN: 978-1-4577-0697-4/12.

6. Prabir Saha, Arindam Banerjee, Partha Bhattacharyya and Anup Dandapat, "High Speed ASIC Design of Complex Multiplier Using Vedic Mathematics", Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January,2011, IIT Kharagpur, ISBN: 978-1-4244-8943-5/11, pp.237-241.

7. Soma BhanuTej, "Vedic Algorithms to develop green chips for future", International Journal of Systems, Algorithms & Applications, Volume 2, Issue ICAEM12, February 2012, ISSN Online: 2277-2677