OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Design of High Speed Low Power Multiplier Using Nikhilam Sutra with Help of Reversible Logic

Manjeet Singh Sankhwar, Rajesh Khatri
Department of Electronics & Instrumentation SGSITS Indore,India

ABSTRACT: *Multipliers are vital components of any processor or computing machine. More often than not, performance of microcontrollers and Digital signal processors are evaluated on the basis of number of multiplications performed in unit time. Hence better multiplier architectures are bound to increase the efficiency of the system. Vedic multiplier is one such promising solution. Its simple architecture coupled with increased speed forms an unparalleled combination for serving any complex multiplication computations. Tagged with these highlights, implementing this with reversible logic further reduces power dissipation. Power dissipation is another important constraint in an embedded system which cannot be neglected. In this paper we bring out a Vedic multiplier known as " Nikhilam Sutra multiplier". The ―Nikhilam Navatascaram Dasatah literally means ―All from Nine and the last from Ten. The sutra basically means start from the left most digit and begin subtracting _9' from each of the digits; but subtract _10' from the last digit.This will be implemented using reversible logic, which is the first of its kind. This multiplier may find applications in Fast Fourier Transforms (FFTs), and other applications of DSP like imaging, software defined radios, wireless communications.*

Index Terms: *Reversible Logic, Nikhilam Sutra, vedic multiplier, Quantum cost, Total reversible logic implementation cost.*

I. INTRODUCTION

Vedic mathematics is the ancient Indian system of mathematics which mainly deals with Vedic mathematical formulae and their application to various branches of mathematics. Vedic mathematics was reconstructed from the ancient Indian scriptures (Vedas) by Sri Bharati Krishna Tirtha after his research on Vedas . He constructed 16 sutras and16 upa sutras after extensive research in Atharva Veda. The most famous among these 16 are Nikhilam Sutram,Urdhva Tiryakbhayam, and Anurupye. It has been found that Nikhilam is the most efficient among these. The beauty of Vedic mathematics lies in the fact that it reduces otherwise cumbersome looking calculations in conventional mathematics to very simple ones. This is so because the Vedic formulae are claimed to be based on the natural principles on which the human mind works. Hence multiplications in DSP blocks can be performed at faster rate. This is a very interesting field and presents some effective algorithms which can be applied to various branches of engineering. Digital signal processing (DSP) is the technology that is omnipresent in almost every engineering discipline. Faster additions and multiplications are the order of the day. Multiplication is the most basic and frequently used operations in a CPU. Multiplication is an operation of scaling one number by another. Multiplication operations also form the basis for other complex operations such as convolution, Discrete Fourier Transform, Fast Fourier Trans forms , etc.With ever increasing need for faster clock frequency it becomes imperative to have faster arithmetic unit. Hence Vedic mathematics can be aptly employed here to perform multiplication.

Reversible logic is one of the promising fields for future low power design technologies. Since one of the requirements of all DSP processors and other hand held devices is to minimize power dissipation multipliers with high speed and lower dissipations are critical. This paper proposes an implementation of Reversible Nikhilam Multiplier which consists of two cardinal features. One is the fast multiplication feature derived from Vedic algorithm Nikhilam sutra and another is the reduced heat dissipation by the virtue of implementing the circuit using reversible logic gates. The paper is partitioned into Six sections. Section (ii) gives literature survey, Section (iii)deals with reversible logic. Section (IV) explains the Nikhilam sutra algorithm. Section (V) elaborates on the design aspects of Reversible Nikhilam vedic sutra Multiplier. Section (vi) Evaluation Conc1usions and references follow.

II. LITERATURE SURVEY

Energy loss is an important consideration in digital circuit design. A part of this problem arises from the technological non ideality of switches and materials. The other part of the problem arises from Landauer's principle for which there is no solution. **Landauer's Principle states that logical computations that are not reversible necessarily generate k*T*ln(2) joules of heat energy, where k is the Boltzmann's Constant k=1.38xlO-23 J/K, T is the absolute temperature** at which the computation is performed. Although this amount of heat appears to be small, Moore's Law predicts exponential growth of heat generated due to information lost, which will be a noticeable amount of heat loss in next decade. **Also by second law of thermodynamics any process that is reversible will not change its entropy.** On thermo dynamical grounds, the erasure of one bit of information from the mechanical degrees of a system must be accompanied by the thermalization of an amount of k*T*ln (2) joules of energy. The information entropy H can be calculated for any probability distribution. Similarly the thermodynamic entropy S refers to thermodynamic probabilities specifically. Thus gain in entropy always means loss of information, and nothing more. Design that does not result in information loss is called reversible. It naturally takes care of heat generated due to information loss. Bennett showed that zero energy dissipation would be possible only if the network consists of reversible logic gates, Thus reversibility will become an essential property in future circuit design technologies.

III. REVERSIBLE LOGIC

Reversible logic is a promising computing design paradigm which presents a method for constructing computers that produce no heat dissipation. Reversible computing emerged as a result of the application of quantum mechanics principles towards the development of a universal computing machine. Specifically, the fundamentals of reversible computing are based on the relationship between entropy, heat transfer between molecules in a system, the probability of a quantum particle occupying a particular state at any given time, and the quantum electrodynamics between electrons when they are in dose proximity. The basic principle of reversible computing is that a bi-jective device with an identical number of input and output lines will produce a computing environment where the electrodynamics of the system allow for prediction of all future states based on known past states, and the system reaches every possible state, resulting in no heat dissipation A reversible logic gate is an N-input N-output logic device that provides one to one mapping between the input and the output. It not only helps us to determine the outputs from the inputs but also helps us to uniquely recover the inputs from the outputs. Garbage outputs are those which do not contribute to the reversible logic realization of the design. Quantum cast refers to the cost of the circuit in terms of the cost of a primitive gate. Gate count is the number of reversible gates used to realize the function. Gate level refers to the number of levels which are required to realize the given logic functions.

The following are the important design constraints for reversible logic circuits.

I. Reversible logic gates do not allow fan-outs.
2. Reversible logic circuits should have minimum quantum cost.
3. The design can be optimized so as to produce minimum number of garbage outputs.
4. The reversible logic circuits must use minimum number of constant inputs.
5. The reversible logic circuits must use a minimum logic depth or gate levels.

The basic reversible logic gates encountered during the design are listed below:

1. **Feynman Gate :**

It is a 2x2 gate and its logic circuit is as shown in the figure. It is also known as Controlled Not (CNOT) Gate. It has quantum cost 1 and is generally used for Fan Out purposes.

2. **Peres Gate :**

It is a 3x3 gate and its logic circuit is as shown in the figure. It has quantum cost 4. It is used to realize various Boolean functions such as AND, XOR.

3. **Fred kin Gate :**

It is a 3x3 gate and its logic circuit is as shown in the figure. It has quantum cost 5. It can be used to implement a Multiplexer.

4. **HNG Gate :**

It is a 4x4 gate and its logic circuit is as shown in the figure. It has quantum cost 6. It is used for designing ripple carry adders. It can produce both sum and carry in a single gate thus minimizing the garbage and gate counts.

5.**Toffoli Gate :** The 3*3 Reversible gate with 3 inputs and 3 outputs. It has Quantum cost **5**

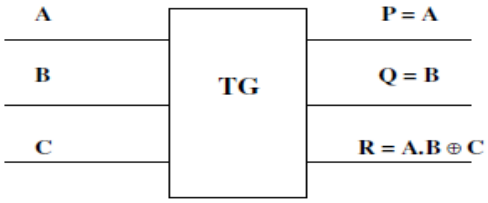
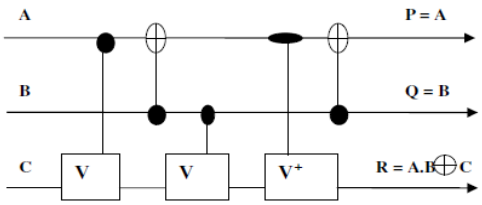


Figure 1. Toffoli gate



Quantum implementation of Toffoli gate

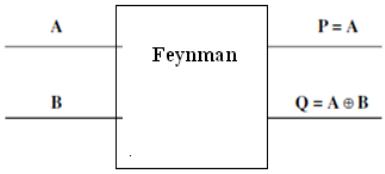
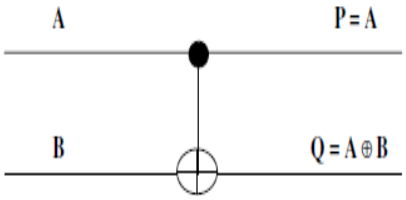


Figure 2. Feynman gate



Feynman/CNOT gate quantum implementatio

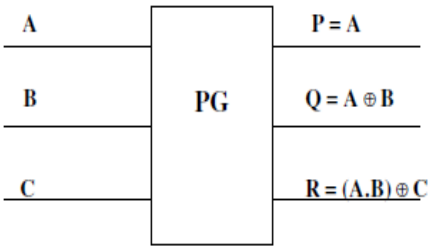
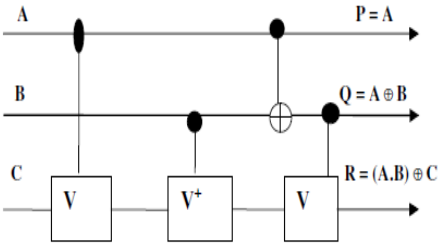


Figure 3. Peres gate



Quantum implementation of Peres gate

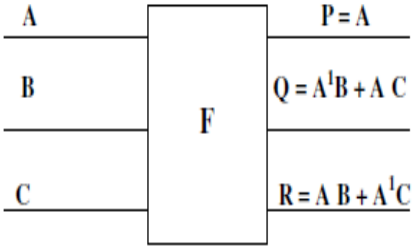
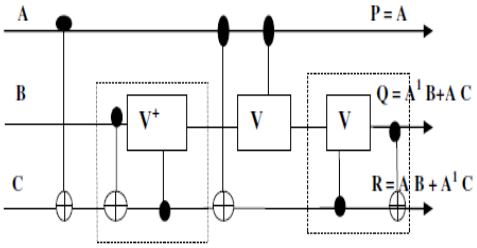


Figure 4. Fredkin gate



Quantum implementation of Fredkin gate

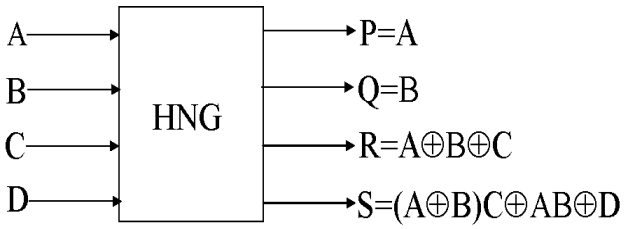


Figure 5. HNG gate

IV. NIKHILAM SUTRA MULTIPLICATION ALGORITHM

The ―Nikhilam Navatascaram Dasatah literally means ―All from Nine and the last from Ten. The sutra basically means start from the left most digit and begin subtracting _9' from each of the digits; but subtract _10' from the last digit.The following example illustrates the way in which this Sutra could reduce the number of iterations to reduce the whole Multiplication.

To multiply 92 and 89. Apply Nikhilam Sutra – ―All from nine and last from ten on both the numbers ―

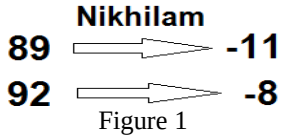


Figure 1

The arrows in Figure 2 indicate the operation of the Nikhilam Sutra being performed, viz. the subtraction of 10 from the last digit and 9‘s from all the other digits starting with the leftmost digit.

 Now we write this down side-by-side,

92	-08
89	-11

 Multiply (-08) and (-11) to get ‗88‘.

92	-08	x
89	-11	
	88	

 Now we cross-add. This is done by both ―adding 92 and -11 to get 81or ―adding 89 and –08 to get 81.

 Note that in both operations you get the same answer that is ‗81‘ which is written below to get the solution.

92	-08	
x		
89	-11	
81	88	

This technique works very well if the numbers to be multiplied are near a base. Upon little alteration, this also works very well for any pair of numbers.

Case (i): Both the numbers are lower than the base.

EXP.1 986*989. Base is 1000

986	**14**
989	**11**

986-11 14*11
989-14

975 154

So 986*989=975154

Case (ii): Both the numbers are higher than the base.

EXP.2 104*102. Base is 100

104	**04**
102	**02**

104+2 04*02

102+4

106 08

So 104*102=10608

Case (iii): One number is more and other is less than the base.

EXP.3 998*1025.Base is 1000

998 002
1025 025

998-25 2*25
1025+2

1023 50 (since the complement of 50 is 950 for the base 1000)

We get 1023 950 But we have to subtract 1 from 1023

So 998*1025= 1022950

After this illustration, we now discuss the operational principle of Nikhilam Sutra by taking the case of multiplication of two
n–bit numbers x and y having complements $\boldsymbol{x} = \mathbf{10}\boldsymbol{n} - \boldsymbol{x}$ and $\boldsymbol{y} = \mathbf{10}\boldsymbol{n} - \boldsymbol{y}$ respectively.
The required product ‗*p*' is defined as: **p = xy ... (1)**
which can be reframed by adding and subtracting $\mathbf{102}\boldsymbol{n}$ +$\mathbf{10}\boldsymbol{n}(\boldsymbol{x} + \boldsymbol{y})$ to the right hand side as:
$\boldsymbol{p} = \boldsymbol{xy} + \mathbf{102}\boldsymbol{n} - \mathbf{102}\boldsymbol{n} + \mathbf{10}\boldsymbol{n}\ \boldsymbol{x} + \boldsymbol{y} - (\boldsymbol{x} + \boldsymbol{y})$ **... (2)**

The above terms can be clubbed as follows:
$\boldsymbol{p} = \mathbf{10}\boldsymbol{n}\ \boldsymbol{x} + \boldsymbol{y} - \mathbf{102}\boldsymbol{n} + \mathbf{102}\boldsymbol{n} - \mathbf{10}\boldsymbol{n}\ \boldsymbol{x} + \boldsymbol{y} + \boldsymbol{xy} = \mathbf{10}\boldsymbol{n}\ \boldsymbol{x} + \boldsymbol{y} - \mathbf{10}\boldsymbol{n} + \mathbf{10}\boldsymbol{n} - \boldsymbol{x}\ \mathbf{10}\boldsymbol{n} - \boldsymbol{y}$
$= \mathbf{10}\boldsymbol{n}\ \boldsymbol{x} - \boldsymbol{y} + \boldsymbol{x}\ \boldsymbol{y} = \mathbf{10}\boldsymbol{n}\ \boldsymbol{y} - \boldsymbol{x} + \{\boldsymbol{x}\ \boldsymbol{y}\}$ **... (3)**

From (3), the expressions of LHS and RHS can be deduced,
which come out to be:
$\boldsymbol{LHS} = \boldsymbol{x} - \boldsymbol{y} = \{\boldsymbol{y} - \}$ **... (4)**
$\boldsymbol{RHS} = \{\}$ **... (5)**

Hence the multiplication of two n- bit numbers is reduced to the multiplication of their complements. To take full advantage of this reduction, it should be ensured that the
numbers obtained after taking the complements are lesser than the original numbers.
This condition is satisfied if both the original numbers are greater than 10n=2, i.e., x > 10n=2 and y > 10n=2. This is the reason why it is said that the Nikhilam Sutra is
efficacious in the multiplication of large numbers than the smaller ones.

V. DETAILED DESIGN (PROPOSED ARCHITECTURE)

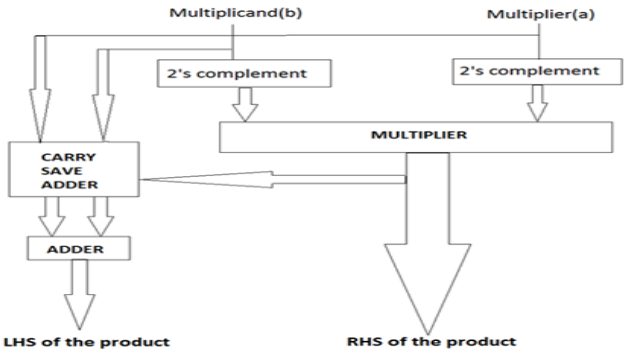


Figure 2

5.1. Top Module:

The block diagram for proposed multiplier is shown in the figure2. As we are using binary numbers in digital signal processing applications we have implemented for binary system. The multiplication can be done using the complementer, CSA adder and adder. The RHS of the product can be obtained by the multiplication of complimented outputs of multiplier and multiplicand and the LHS of the product can be obtained by addition using a CSA. This can be used for the multiplication of any number of bits. In this paper, we have presented a 4x4 architecture applying the Nikhilam Algorithm. In the figure 2, the two inputs a and b represents the 4 bit multiplier and 4 bit multiplicand respectively.

5.2. Internal blocks:

5.2.1 2's complementer:

The multiplicand and the multiplier are given as inputs to the two 2's complementer blocks. The logic implementation of the 4 bit 2's complementer is presented in figure3

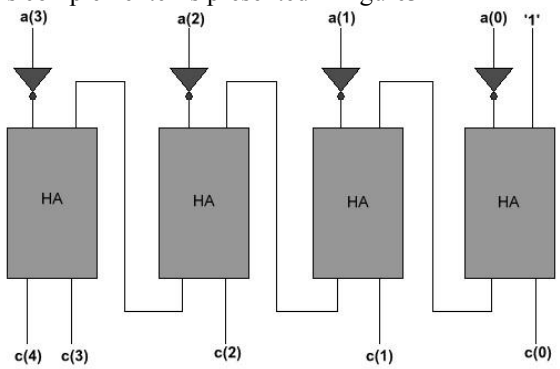


Figure3

In figure3, the ―HA represents a Half Adder block.

5.2.2 Multiplier:

Now, the complemented output of multiplier (-a) and the complemented multiplicand (-b) are then produced. These complemented outputs of multiplier and multiplicand are given as inputs to the multiplier block. The 4x4 multiplier architecture that we employed is based on calling a 2x2 multiplier so as to ease the multiplication procedure. This implementation is represented by the
following figure 4. Here, a and b are the 2-bit (or the 4-bit) multiplier and multiplicand respectively which are being multiplied to produce the final 4-bit(or the 8-bit) product vector.The half of the LSB bits of multiplication output is taken as RHS product of the total multiplication.

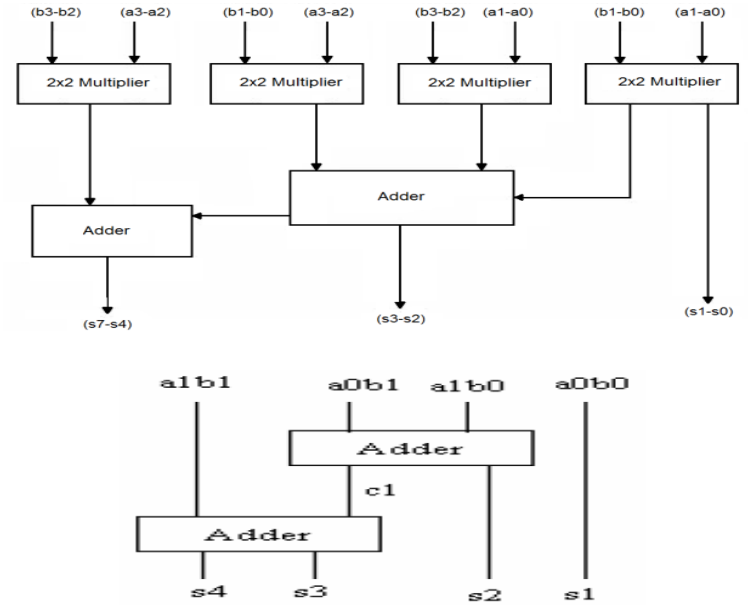


Figure 4

5.2.3 CSA:

The carry-save unit consists of *n* full adders, as shown inFigure5 each of which computes a single sum and carry bitbased solely on the corresponding bits of the three input numbers.

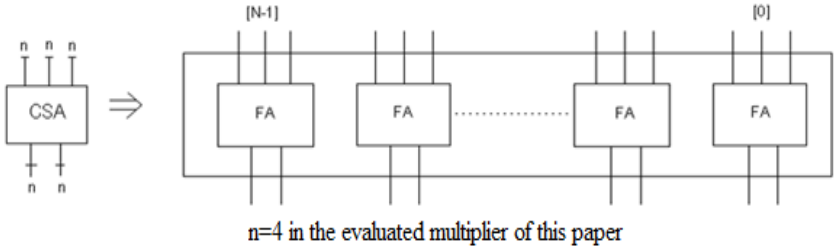


Figure 5

With three *n* - bit numbers ai, bi, and ci given to it, it produces a partial sum psi and a shift-carry sci according to the below equations:-

Psi=ai □ bi □ ci

Sci=(ai □ bi) V (ai □ ci) V (bi □ ci)

These psi and sci are then added using a conventional adder, to produce the sum of the three inputs. The multiplier, multiplicand and the half of MSB bits are given as inputs to the CSA. The two outputs i.e., sum vector and carry vector obtained from the CSA adder are given to the
inputs for the adder block. The output he we obtain is labeled the LHS of the required multiplication product. As we are using CSA, the delay will be reduced and the number of components gets reduced for addition mechanism. This is the main advantage of the multiplier based on Nikhilam over the multiplier based on conventional algorithms
proposed in the a fore mentioned algorithms in _I'.

VI. EVALUATION

The design of the reversible 2,4,8,16 bit multiplier is logically verified using XILINX 8.1i and MODELSIM SE 6.5. Synthesis and Implementation is done using Xilink Spartan-3E Fpga Board of device family xc3s500e-4fg320.The simulation results are as shown in figures .

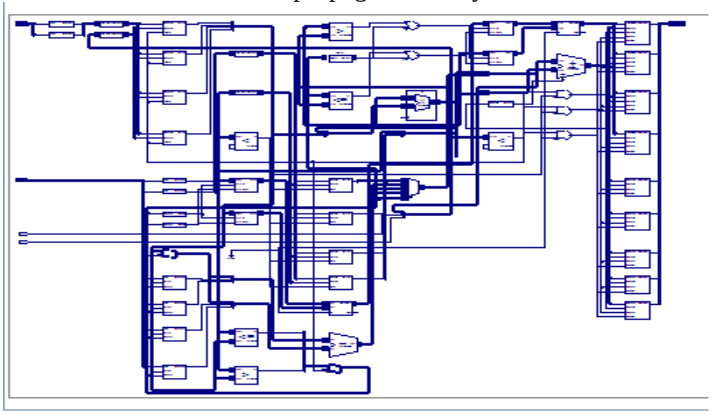
Here we introduce a new function called the "Total Reversible Logic Implementation Cost (TRLlC)" which is defined as the sum of all the cost metrics of a given reversible circuit. The TRLIC can be deemed as a parameter which reflects the overall performance of a reversible logic circuit. TRLIC=∑(NG,CI,QC,GO)

Where NG is the number of gates in the reversible circuit.CI is the number of constant inputs, QC is the quantum cost of the circuit. GO is the number of garbage outputs. The following are the important design constraints for any reversible logic circuits.

I. Reversible logic circuits should have minimum quantum cost.
2. The design can be optimized so as to produce minimum number of garbage outputs.
3. The reversible logic circuits must use minimum number of constant inputs.
4. The reversible logic circuits must use a minimum number of reversible gates.

6.1. Simulation Methodology

Xilinx 8.1i has been used to simulate the wave forms. The simulator carefully modeled the interconnections, the associated blocks and the propagation delays.

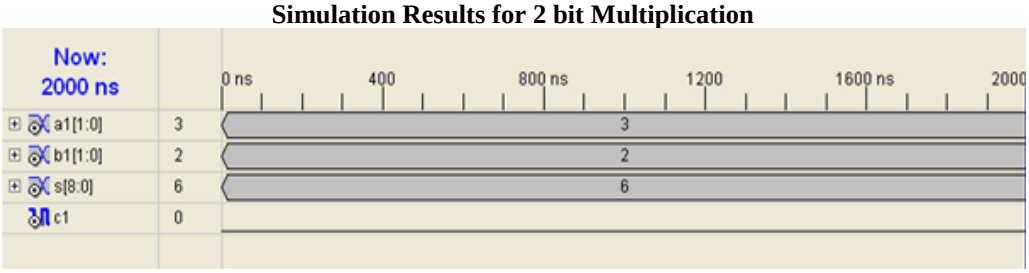


RTL Schematic

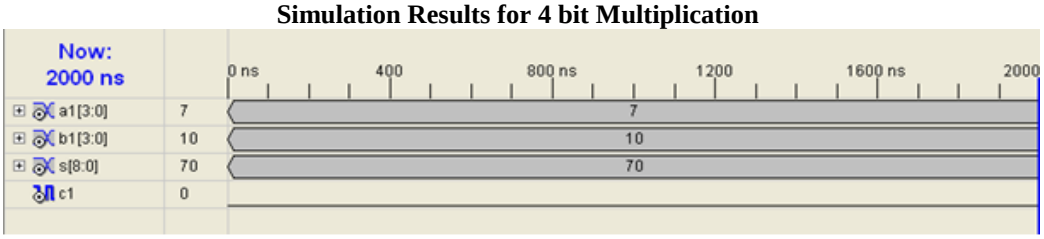
6.2.Results

In this section we show results for the Vedic multiplier based on Nikhilam Sutra . Multiplier based on Nikhilam Algorithm utilizes smaller area and produces littler delay than the conventional multiplier.

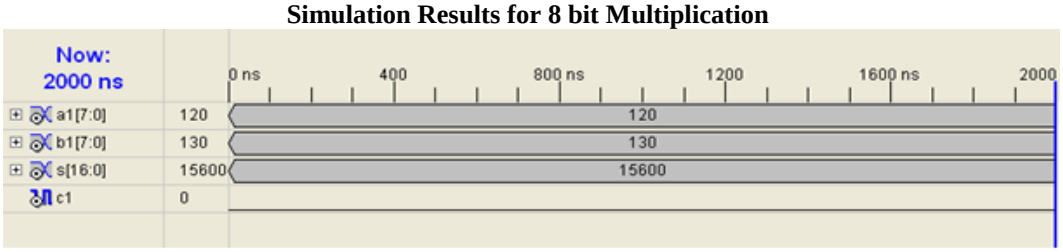
Simulation Results for 2 bit Multiplication



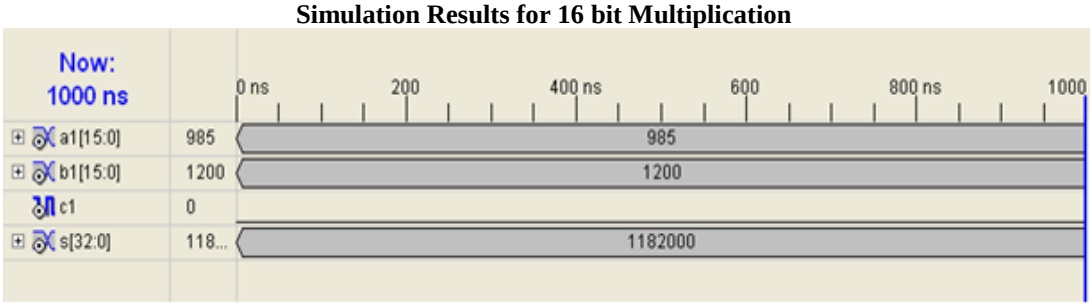
Simulation Results for 4 bit Multiplication



Simulation Results for 8 bit Multiplication



Simulation Results for 16 bit Multiplication



CONCLUSION

The proposed Vedic multiplier architecture exhibits speed improvements. The Vedic multiplier employing Nikhilam Sutra found to be better than conventional multiplier in terms of speed when magnitude of both operands are more than half of their maximum values . This approach may be well suited for multiplication of numbers with more than 16 bit size.

REFERENCES

[1] Charles. Roth Jr. "Digital Systems Design using VHDL," Thomson Brooks/Cole, 7th reprint, 2005.

[2] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaja, Vedic Mathematics: Sixteen Simple Mathematical Formulae from the Veda, Delhi (1965).

[3] H. Thapliyal , M. B. Shrinivas and H. Arbania, "Design and Analysis of a VLSI Based High Performance Low Power Parallel Square Architecture", Int. Conf. Algo. Math. Comp. Sc., LasVegas, June 2005, pp. 72-76.

[4] P. D. Chidgup kar and M. T. Karad, "The Imp lamentation of Vedic Algorithms in Digital Signal Processing", Global J. of Engg. Edu, vol.8, no.2, 2004.

[5] Shamim Akhter,"VHDL Implementation Of Fast NXN Multiplier Based On Vedic Mathematics", Jay p ee Institute of Information Technology University, Noida, 201307 UP, INDIA, 2007 IEEE.

[6] Rakshith Saligram and Rakshith T.R. "Design of Reversible Multipliers for linear filtering Applications in DSP" International Journal of VLSI Design and Communication systems, Dec-12
[7] R. Landauer,"Irreversibility and Heat Generation in the Computational Process", IBM Journal of Research and Development, 5, pp.183-191, 1961.
[8] C.H. Bennett, "Logical reversibility of Computation", IBM J. Research and Development, pp.525-532, November 1973.
[9] R. Feynman, "Quantum Mechanical Computers," Optics News,Vol.1l, pp. 11-20, 1985.
[10] H. Thapliyal and M.B. Srinivas, "Novel Reversible Multiplier Architecture Using Reversible TSG Gate", Proc.IEEE International Conference on Computer Systems and Applications, pp. 100-103, March 2006.
[11] Shams, M., M. Haghparast and K. Navi, Novel reversible multiplier circuit in nanotechnology. World Appl. Sci. J., 3(5): 806-810.
[12] Somayeh Babazadeh and Majid Haghparast, "Design of a Nanometric Fault Tolerant Reversible Multiplier Circuit" Journal of Basic and Applied Scientific Research, 2012.
[13] Thapliyal, H., M.B. Srinivas and H.R. Arabnia, 2005, A Reversible Version of 4x4 Bit Array Multiplier with Minimum Gates and Garbage Outputs, Int. Conf. Embedded System, Applications (ESA'05), Las Vegas, USA, pp: 106 114.
[14] M. Haghparast et al. , "Design of a Novel Reversible Multiplier Circuit using HNG Gate in Nanotechnology," in World Applied Science Journal, Vol. 3, No. 6, pp. 974-978,2008.
[15] M. S. Islam et al. , "Realization of Reversible Multiplier Circuit," in Information Tech. 1, Vol. 8, No. 2, pp. 117-121,2005.
[16] K. Navi, M. Haghparast, S. JafaraliJassbi, O.Hashemipour, Design of a novel reversible multiplier circuitusing HNG gate, World Sci. 1 3 (6).
[17] M. Shams et al., "Novel Reversible Multiplier Circuits in Nanotechnology," in World Applied Science Journal, Vol. 3, No. 5, pp, pp. 806- 810, 2008.
[18] M S Islam, M M Rahman, Z Begum and M Z Hafiz, 2009. Low Cost Quantum Realization of Reversible Multiplier Circuit. Information Technology Journal, vol. 8(2), pp. 208-213.
[19] E. Fredkin and T. Toffoli,"Conservative Logic", Int'l 1 Theoretical Physics Vo121, pp.219-253, 1982.
[20] A. Peres, Reversible logic and quantum computers, Phys. Rev. A 32 (1985) 3266-3276.
[21] Rakshith Saligram and Rakshith T.R. "Novel Code Converter Employing Reversible Logic", International Journal of Computer Applications (IJCA), August 2012.
[22] G Ganesh Kumarand V Charishma, Design of high speed vedic multiplier using vedic mathematics techniques, ltn'l J. of Scientific and Research Publications, Vol. 2 Issue 3 March 2012
[23] Vedic Mathematics: http://www.hinduism.co.za/vedic.html
[24] 24.Rakshith Saligram and Rakshith T.R. " Design of high speed low power multiplier using Reversible logic : a vedic Mathematical Approach (ICCPCT), 2013.

About The Authors

	**Manjeet Singh Sankhwar** has 3 years of Teaching experience and presently pursuing Mtech in Micro Electronics & VLSI Design in SGSITS Indore, India in 2014 .He received his Bachelor of Engineering in Electronics and Communication from UIT-RGPV Bhopal,India in 2008 . His main research interests include Low Power VLSI design , Digital design.
	**Mr. Rajesh Khatri** has 20 years of Teaching experience and presently working as Associate Professor in Department of Electronics & Instrumentation Engineering at SGSITS Indore. He received his Bachelor of Engineering in Electronics and Communication from SGGS Institute of Technology Nanded , India in 1990 . He received his Master of Engineering in Instrumentation from SGGS Institute of Technology Nanded ,India in 1995 .