IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 3, March 2017

Design of Low Power Multiplier using Reversible logic: A Vedic Mathematical Approach

D.V.R Mohan[1], K.Vidyamadhuri[2], Y.RamaLakshmanna[3], K.H.S Suresh kumar[4]

Professor, Department of ECE, SRKR Engineering College, Bhimavaram, India[1]

P.G Student, Department of ECE, SRKR Engineering College,Bhimavaram, India[2]

Associate Professor Department of ECE, SRKR Engineering College, Bhimavaram, India[3]

P.G Student, Department of ECE, SRKR Engineering College, Bhimavaram, India[4]

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 theefficiency 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 "UrdhvaTiryakbhayam"meaning vertical and crosswise, 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.

Keywords: VedicMultiplier, Reversible Logic, UrdhvaTiryakbhayam, Power utilization.

I.INTRODUCTION

The design for low power has become one of the greatest challenges in high-performance very large scale integration (VLSI) design in recent years. As a result, many methods have been introduced to minimize the power consumption of new VLSI systems.Most of these methods focus on the power consumption during normal mode of operation, while test mode operation has not normally been a predominant concern. However, it has been found that the power consumed during test mode operation is often much higher than during normal mode operation. This is because most of the consumed power results from the switching activity in the nodes of the circuit under test (CUT), which is much higher during test mode than during normal mode operation [1].

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 ancientIndian scriptures (Vedas) by Sri BharatiKrishna Tirthaafter his research on Vedas .

He constructed 16 sutras and 16 upa sutras after extensive research in Atharva Veda. The most famous among these 16 are NikhilamSutram, UrdhvaTiryakbhayam, and Anurupye. It has been found that UrdhvaTiryakbhayam 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[2].

Digital signal processing (DSP) is the technology thatis omnipresent inalmost every engineering discipline. Fasteradditions 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 thebasis 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 apply employed here to perform multiplication.

Vedic Mathematics offers a fresh and highly efficient approach to mathematics covering a wide range - starts with elementary multiplication and concludes with a relatively advanced topic, the solution of non-linear partial differential equations. But the Vedic scheme is not simply a collection of rapid methods it is a system, a unified approach. Vedic Mathematics extensively exploits the properties of numbers in every practical application.

The objective of good multiplier to provide a physically compact high speed and low power consumption unit.

Being a core part of arithmetic processing unit, multipliers are in extremely high demand on its speed and low power consumption.

To reduce significant power consumption of multiplier design it is a good direction to reduce number of operations there by reducing a dynamic power which is a major part of total power dissipation. In the past considerable effort were put into designing multiplier in VLSI in this directionReversible logic is one of the promismg fields forfuture low power design technologies. Since one of therequirements 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 UrdhvaTiryakbhayamMultiplier which consists of two cardinal features. One is the fast multiplication feature derived from Vedic algorithmUrdhvaTiryakbhayam and another is the reduced heatdissipation 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 UrdhvaTiryakbhayam algorithm. Section V elaborates on the design aspects of Reversible UrdhvaTiryakbhayaMultiplier.Section VI Conc1usions and references follow.

II. LITERATURE SURVEY

Energy loss is an important consideration in digitalcircuit design.A part of this problem arisesfrom thetechnological non ideality of switches and materials. The other part of the problem arises fromLandauer's principle for which there is no solution. Landauer'sPrinciple states that logical computations that are not reversible necessarily generate k*T*ln(2) joules of heat energy, where k is the Boltzmann'sConstant k=1.38x10-23 J/K, T is the absolute temperature atwhich 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 willbe a noticeable amount of heat loss in next decade.

Also by second law of thermodynamics any processthat is reversible will not change its entropyOnthermodynamical 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 ofenergy. 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 iscalled 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 ofreversible 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 bijective device with an identical number of input and output lines will produce acomputing environment where the electrodynamics of thesystem allow for prediction of all future states based on known past states, and the system reaches every possible state,resulting in no heat dissipation [3].

A reversible logic gate is an N-input N-output logicdevice 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 cost 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 logicfunctions.

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

1. 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 1. It is also known as Controlled Not (CNOT) Gate. It has quantum cost one and is generally used for Fan Out purposes.

2. Peres Gate (PG):

It is a 3x3 gate and its logic circuit is as shown in the figure 2. It has quantum cost four. It is used to realize various Boolean functions such as AND, XOR.

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

4.Double Peres Gate(DPG):
This is one of the four input Reversible logic Gate is shown in figure 4.which can be used as full-adder. The full adderusing DPG is obtained with C=0 and D= Cin and its quantum cost is equal to 6.

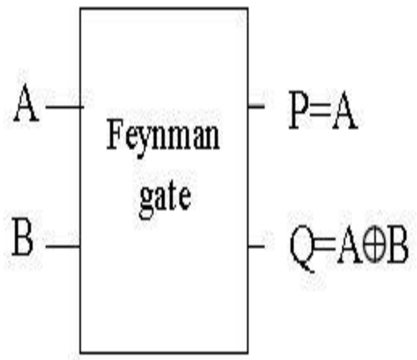

Fig 1.logic circuit of Feynman gate

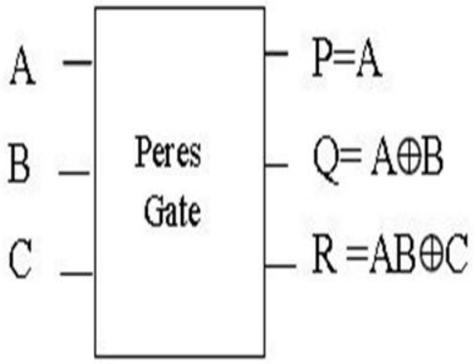

Fig 2.logic circuit of Peres gate

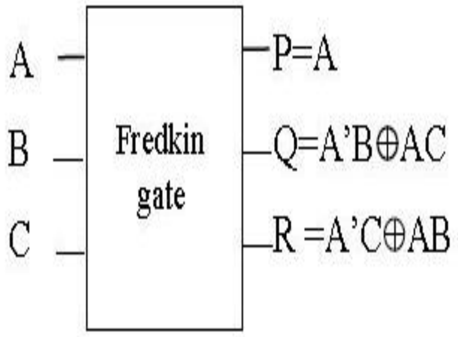

Fig 3.logic circuit of Fredkin gate

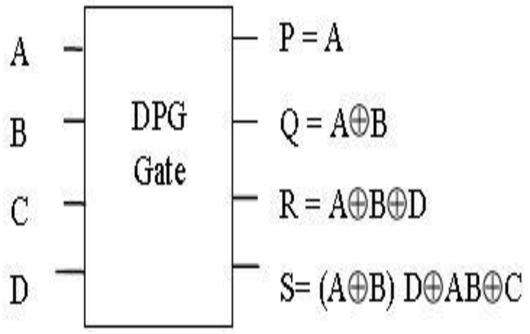

Fig 4.logic circuit of DPG gate

IV. URDHVA TIRYAKBHAYAM MULTIPLICATION ALGORITHAM

UrdhvaTiryakbhayam (UT) is a multiplier based onVedic mathematical algorithms devised by ancient Indian Vedic mathematicians. UrdhvaTiryakbhayam sutra can be applied to all cases of multiplications viz. Binary, Hex and alsoDecimals. It is based on the concept that generation of all partial products can be done and then concurrent addition of these partial products is performed [4].

The parallelism in generation of partial products and their summation is obtained using UrdhvaTiryakbhayam. Unlike other multipliers with theincrease in the number of bits of multiplicand and/or multiplier the time delay in computation of the product does not increase proportionately. Because of this fact the time of computation is independent of clock frequency of the processor. Hence one can limit the clock frequency to a lower value [5]. Also, since processors using lower clock frequency dissipate lower energy, it is economical in terms of power factor to use low frequency processors employing fast algorithms like the above mentioned. The Multiplier based on this sutra has the advantage that as the number of bits increases, gate delay and area increases at a slow pace as compared to other conventional multipliers.The algorithm can be illustrated using the following visual walkthrough in the below figure5.

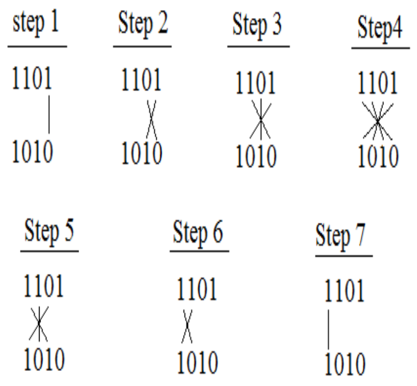

Fig.5 Using UrdhvaTiryakbhayam for binary numbers

To illustrate this multiplication scheme, let us consider the multiplication of two decimal numbers (43*68). The digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result and a carry. This carry is added in the next step and hence the process goes on. If more than one line are there in one step, all the results are added to the previous carry. In each step, least significant bit acts as the result bit and all other bitsact as carry for the next step. Initially the carry is taken to be zero.

The algorithm can be illustrated in decimals using the following visual walkthrough in the below figure 6.

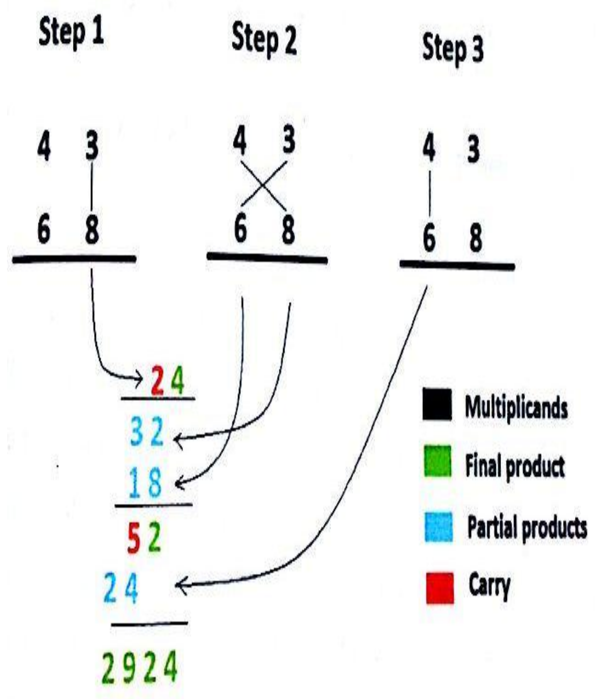

Fig.6 Multiplication of 2 digit decimal numbers using UT Sutra

The Algorithm: Multiplication of 101 by 110.

l. We will take the right-hand digits and multiply the together. This will give us LSB digit of the answer.
.

2. Multiply LSB digit of the top number by the second bit of the bottom number and the LSB of the bottom number by the second bit of the top number. Once we have those values, add them together.

3. Multiply the LSB digit of bottom number with the MSB digit of the top one, LSB digit of top number with the MSB digit of bottom and then multiply the second bit of both, and then add them all together.

4. This step is similar to the second step, just move oneplace to the left. We will multiply the second digit of one number by the MSB of the other number.

5. Finally, simply multiply the LSB of both numbers together to get the final product.

V.ARCHITECTURE OF REVERSIBLE URDHVA TIRYAKBHAYAM MULTIPLIER

Block diagram of 2x2 bit multiplier

To design the 2x2 bit multiplier we use two half-adders is as shown in figure 7. On multiplication of two 2-bit numbers we get 4-bit result, where three of them are sums and other is the final carry.

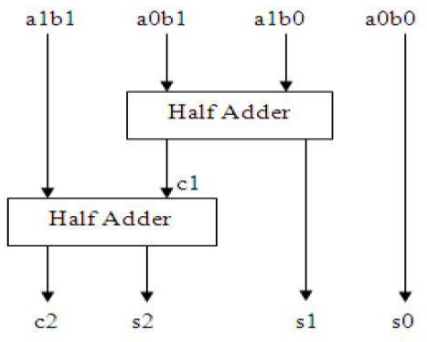

Fig 7:Block diagram of 2x2 bit multiplier

The first result bit is S0 and is generated directly on multiplication of least significant bits of multiplier and multiplicand (assuming initial carry as zero).Then, the LSB of the multiplicand is multiplied with the next higher bit of the multiplier and added with the product of LSB of multiplier and next bit of the multiplicand (crosswise). The sum gives second bit of the product and the carry is added in the output of next stage sum obtained by the crosswise and vertical multiplication and addition of three bits of the two numbers from least significant position.

Implementation of 2X2 multiplier using the conventional logic gate

The digital logic implementation of the 2X2 Urdhva Tiryakbhayam multiplier using the conventional logic gates is as shown in figure8 . The expressions for the Four output bits are given under.

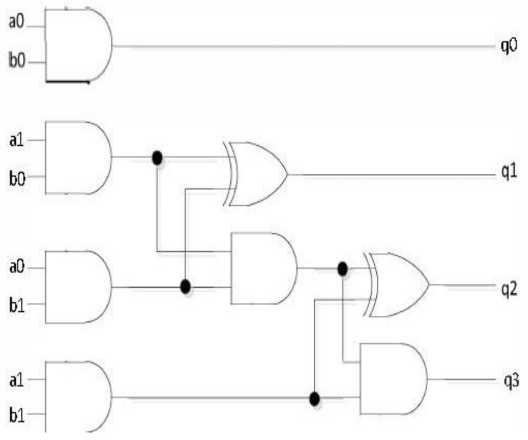

Fig 8.Conventionl logic implementation of 2*2 UT multipler.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 3, March 2017

The expressions for the four output bits are given below,

qO= aO.bO
ql= (a1.bO) xor (aO.bl)
q2= (aO.al.bO.bl) xor (al.bl)
q3= aO.al.bO.bl

Implementation of 2x2 UT multiplier using reversible logic gate

A 2 x 2 UrdhvaTiryakbhayam Multiplier using reversible logic gates can be implemented based on the logical expressions obtained while designing the 2 x 2 UT Multiplier using the conventional logic gates is shown in figure 9.
In designing the 2 x 2 UT Multiplier using reversible logic gates we use three fundamental reversible gates namely, PERES Gate(PG) , FEYNMAN Gate(which is also termed as control NOT Gate(CNOT Gate)) [6].

The Peres Gate can be used as half-adder by making the third input as logic zero(C=0). The sum and the carry of the half-adder are obtained on the third and fourth pins of the Peres Gate. The CNOT Gate is used to generate the Ex-or product of the two inputs [7].

- The circuit requires a total of six reversible logic gates out of which five are Peres gates and remaining one is the CNOT gate.The quantum cost of the 2x2 UT Multiplier is enumerated to be 21.
- The number of Garbage outputs is 9 and the number of Constant inputs is 4.
- This 2x2 multiplier block is cascaded to obtain 4x4 multiplier.

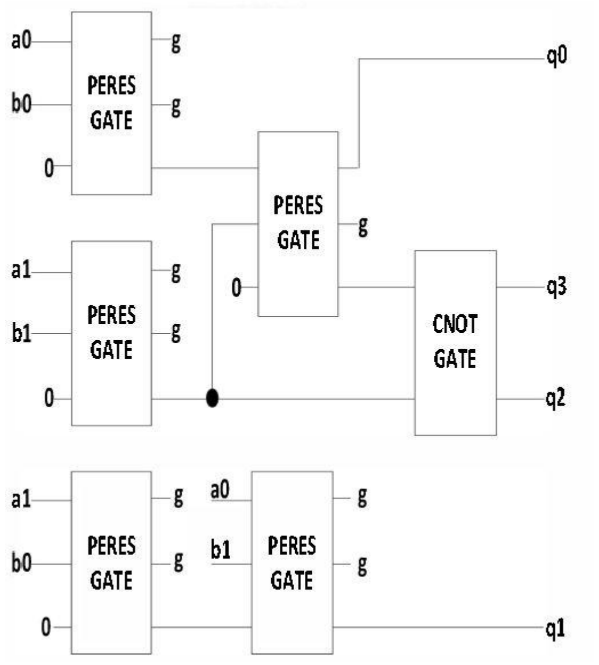

Fig. 9 Reversible implementation of 2x2 UT Multiplier

4-Bit Ripple carry adder using reversible gates

In implementing 4 x 4 UrdhvaTiryakbhayam Vedic Multiplier using Reversible Logic Gates we require to implement the 4-bit parallel adder for the addition of partial products, which are generated intermediately.Thus a 4 bit ripple carry adder needs 4 DPG gates and the 5 bit adder requires 5 DPG gates is shown in figure 10. This design also does not take intoconsideration the fan out gates [8].

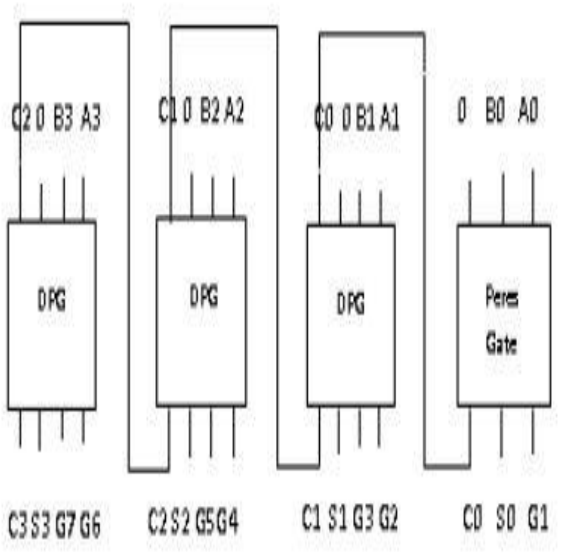

Figure10: 4-Bit Ripple Carry (RC) Adder using DPG and PG gates.

Here, in designing the 4-bit parallel adder we make use of Peres Gate (PG) and Double Peres Gate (DPG) to generate the sum. The Peers Gate(PG) is used as half-adder by making the third input as logic zero (C = 0), While the carry generated in present stage is propagated to the next stage as in case of ripple carry adder and the propagated carry is considered as third input to the full-adder. The Double Peres Gate (DPG) is used as full-adder to add the three bits (including the carry generated in the previous stage) and it is used as full-adder by making the third input as logic zero (C = 0) to generate the carry of that stage by adding three bits in that stage (including the carry generated in the previous stage)[9].

Block Diagram of 4x4 bit UT Vedic Multiplier

The Reversible 4X4 UrdhvaTiryakbhayaMultiplier design emanates from the 2X2 multiplier. The block diagram of the 4X4 Vedic Multiplier is presented in the figure 11. It consists of four 2X2 multipliers each of which procures four bits as inputs; two bits from the multiplicand and two bits from the multiplier. The lower two bits of the output of the first 2X2 multiplier are entrapped as the lowest two bits of the final result of multiplication[10]. Two zeros are concatenated with the upper two bits and given as input to the four bit ripple carry adder.

The other four input bits for the ripple carry adder are obtained from the second 2X2 multiplier. Likewise the outputs of the third and the terminal2X2 multipliers are given as inputs to the second four bit ripple carry adder. The outputs of these four bit ripple carry adders are in turn 5 bits each which need to be summed up. This is done by a five bit ripple carry adder which generates a six bit output. These six bits form the upper bits of the finalresult. The ripple carry adder is consummated (realized) using the DPG gates . The number of bits that need to be ripple carried verdicts the number of DPG gates to be used[11].

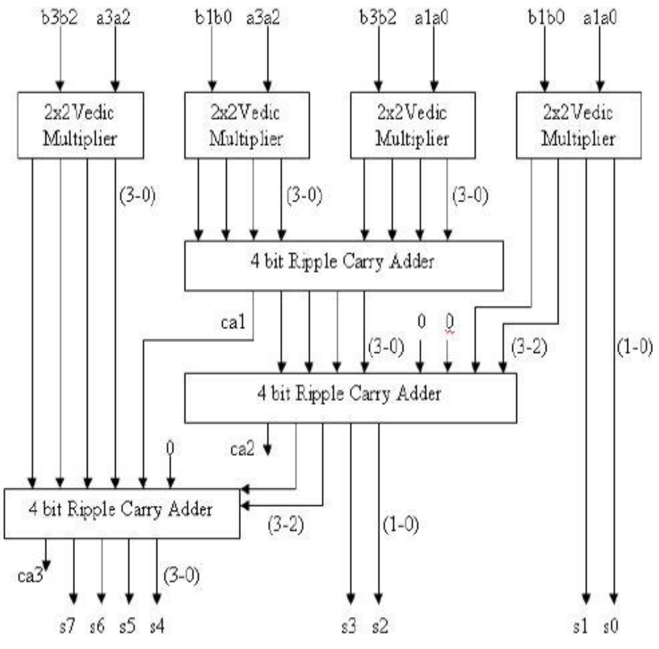

Fig 11.Block diagram of 4*4 UT multiplier

To get final product S7S6S5S4S3S2S1S0 four, 2-bit Vedic multiplier and three 4-bit Ripple Carry (RC) Adders are required. In this proposal, the first 4-bit RC Adder is used to add two 4-bit operands obtained from cross multiplication of the two middle 2x2 bit multiplier modules. The second 4-bit RC Adder is used to add two 4-bit operands, i.e.concatenated 4-bit (“00” & most significant two output bits of right hand most of 2x2 multiplier module as shown in Fig 11.) and one 4-bit operand we get as the output sum of first RC Adder. Its carry “ca1” is forwarded to third RC Adder. Now the third 4-bit RC Adder is usedto add two 4-bit operands, i.e. concatenated 4 –bit (carry ca1, “0” & most significant two outputsum bits of 2ndRCAdder as shown in Fig 11.)and one 4-bit operand we get as the output sum of left hand most of 2x2 multiplier module. Early literature speaks about Vedic multipliers based on array multiplier structures. The Ripple Carry Adder.helps us to reduce delay.

Design of 8×8 block

The design of 8×8 block is a similar arrangement of 4×4 blocks in an optimized manner as in figure 12. The first step in the design of 8×8 block will be grouping the 4 bit (nibble) of each 8 bit input. These quadruple terms will form vertical and crosswise product terms. Each input bit-quadruple is handled by a separate 4×4 Vedic multiplier to produce eight partial product rows. These partial products rows are then added in an 8-bit carry look ahead adder optimally to generate final product bits.

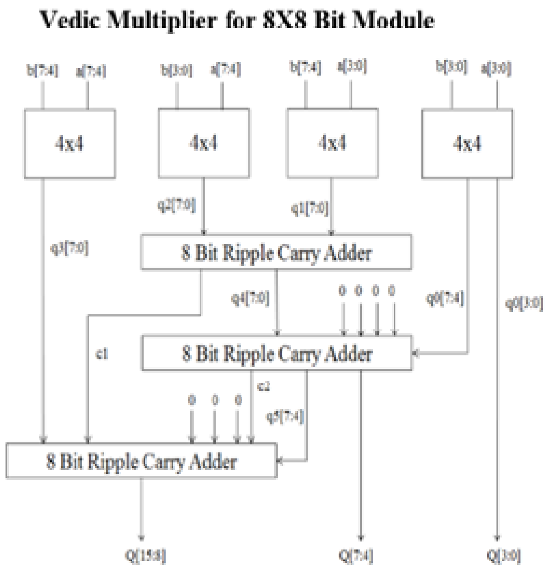

Fig 12.Block diagram of 8*8 UT multiplier

Design of a 16×16 Multiplier

The design of 16×16 block is as shown in figure 13.It is similar arrangement of 8×8 block. The first step in the design of 16×16 block will be grouping the 8 bit (byte) of each 16 bit input. These lower and upper bytes pairs of two inputs will form vertical and crosswise product terms. Each input byte is handled by a separate 8×8 Vedic multiplier to produce sixteen partial product rows. These partial products rows are then added in a 16-bit carry look ahead adder optimally togenerate final product bits.

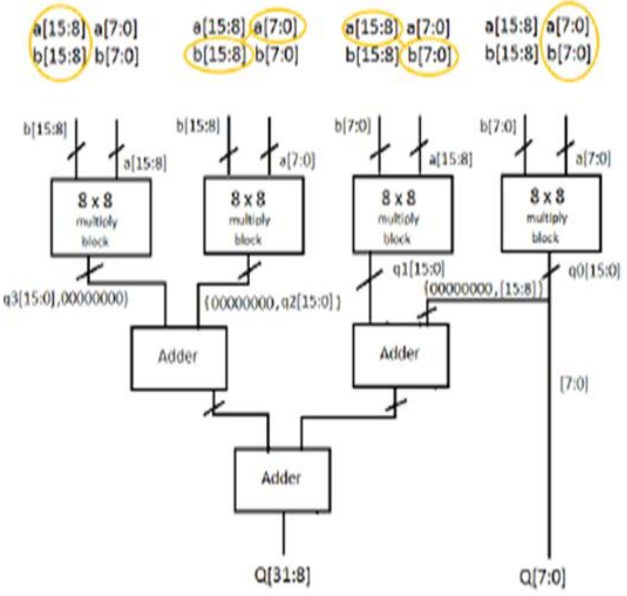

Figure 13 : 16-Bit multiplier using UrdhvaTiryakbhyam Sutra

Table 1: Table of design comparison of VedicMultipliersusing different logic gates

Logic utilization	16x16 Vedic Multiplier Using basic logic gates	16x16 Vedic Multiplier Using reversible logic gates
Power	86.73	33.59
No.Of Slice Registers	493	411
No.of IOB's	66	64
No.of LUT's	1195	730
Memory in KB	213388	318720
Delay in ns	38.15	46..418

IV. CONCLUSION

The main aim of UT algorithm with reversible logic is mainly to design a low power multipliers. This is the optimized design as compared to vedic multiplier using basic logic gates. The power utilization is reduced drastically compared to the basic logic gates. The further optimization of the circuit in terms of high speed and low power as future work.

REFERENCES

[1]Rakshith TR RakshithSaligram Design 0f High Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach TR. 2013 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013]

[2] RakshithSaligram andRakshith T.R. "Design of Reversible Multipliers for linear filtering Applications in DSP" International Journal of VLSI Design and Communication systems, Dec-12

[3] R. Landauer,"Irreversibility and Heat Generation in the Computational Process", IBM Journal of Research and Development, 5, pp.183-191, 1961.

[4] C.H. Bennett, "Logical reversibility of Computation", IBM J. Research and Development, pp.525-532, November 1973.

[5] 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.

[6] Shams, M., M. Haghparast and K. Navi, Novel reversible multiplier circuit in nanotechnology. World Appl. Sci. J.,3(5): 806-810.

[7] SomayehBabazadeh andMajidHaghparast, "Design of a Nanometric Fault Tolerant Reversible Multiplier Circuit" Journal of Basic and Applied Scientific Research, 2012.

[8] H. Thapliyal and M.B. Srinivas, "Reversible Multiplier Architecture Using TSG Gate", Proc. IEEE International Conference on Computer Systems and Applications, pp. 241-244, March 20 07.

[9] M S Islam, M MRahman, Z Begum and M Z Hafiz, 2009. Low Cost Quantum Realization of Reversible Multiplier Circuit. Information Technology Journal, vol. 8(2), pp. 208-213.

[10] RakshithSaligram and Rakshith T.R. "Novel CodeConverter Employing Reversible Logic", InternationalJournal of Computer Applications (IJCA), August 2012.

[11]Y. Rama Lakshmanna, G.V.S. Padma Rao, K. BalaSindhuri, N. Udaya Kumar "Modified Vedic Multiplier using Koggstone Adders" International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 5, Issue 10, October 2016 IJARCCE ISSN (Online) 2278-1021 ISSN (Print) 2319 5940 DOI10.17148/IJARCCE.2016.51074 page no 361-371.

[12]Sonalis.Kothule,GovindU.kharat,shekhar H.Bodake_"A Review on vedic multiplier using Reversable logic gates"Internationaljournalof innovative research in science engineering and technology vol.5 issue 4,April 2016 IISN (online):2319 8753.DOI:10.15680/IJIRSET.2016.0504120 research in science engineering and technology vol.5 issue 4,April 2016 IISN (online):2319 8753.DOI:10.15680/IJIRSET.2016.0504120