IJETST- Vol.||02||Issue||03||Pages 2139-2146||March||ISSN 2348-9480 2015

Open access Journal International Journal of Emerging Trends in Science and Technology

VHDL implementation of a Novel Low Power Squaring Circuit Using YTVY Algorithm of Vedic Mathematics

Authors

Pabitra Kumar Mohapatra[1], Siba Kumar Panda[2], Sambita Dalal[2], Shibashis Pradhan[2]

[1]M.Tech Scholar, CUTM, Jatni, Bhubaneswar, Odisha, India
Email: *pabitra20@gmail.com*
[2]Assistant Professor, Department of ECE, Centurion University Of Technology & Management,
Bhubaneswar, Odisha, India
Email: *panda.sibakumar08@gmail.com, siba.panda@cutm.ac.in*

Abstract

Squaring plays an important role in VLSI signal processing applications. The multiplier piece is used to square of a number in much complex multiplication. For carrying out the large hardware circuit, the multiplier unit is most efficient and time consuming. In multiplier unit, the squaring operation is unique case. A exclusive and proper squaring circuit can be a remarkable upgrade the computation time and in the power reduction to a large extend. Ingeneral, the squaring circuits use very fast multiplier. A innovative idea of a squaring circuit without using multiplier is suggested in this paper. In this paper, we have implemented a novel algorithm Yavadunam Tavadunikrtya Vargarica Yojayet (YTVY) of ancient vedic mathematics for the squaring operation circuit. The main advantage of this paper is that no multiplier is used for the squaring circuit. The circuit is designed with the help of VHDLlanguages and synthesized in Xilinx ISE Design Suite 14.1.

Keywords: *Squaring Circuit, VLSI Signal Processing,YTVY Sutra, Vedic Mathematics,FPGA,VHDL*

1. Introduction

With the encroachment in VLSI technology, more and more functionality, intricacy has been integrated into digital designs for better prolongment of target applications. ManyVLSI Signal Processing applications require support for floating point arithmetic, complex arithmetic modules like multipliers and powering units. Scaling is a technique by which we are reducing the dimension of a transistor so as to meet the design criteria. Thequandary with these intricatereckoning blocks like multipliers and squaring units is that, they entail longer cycle times for working out.In order to achieve the requirementsof low power, less area and high speed in the design, we are adopting various architectures.The various architectures are pipelining, parallel processing, digital signal processors, implementation of vedic algorithms etc.Ancient Vedic mathematics comprises of sixteen sutras and thirteen sub sutras that are widely used for the enhancement in the field of VLSI Signal Processing Applications.

In many Signal processing applications multipliers are frequently used to bring out squares and advanced orderof power operations. Multiplier design requires large area and consumes a substantialquantity of power per computation. But for powering operations, a general-purpose multiplier resultspower in a great extent, which is a contradiction to the very large scale integration technology concept.

In this paper weput forward to use devoted squaring circuit unit which performs the squaring operation without using any multiplication operation. The advantage with using dedicated units is that it consumes less power compared to general-purpose multipliers. Widespread applications of Squaring operations includes floating point divide and square-root, cryptography, computation of Euclidean distance among nodes in graphic processing, and rectangular to polar conversion etc.Due to innatereward in reducing energy required per computation for powering units, a committed squaring unit performs superior as compared to a general-purpose multiplier.

The paper is structured as follows: Section 2 presents a brief description of related works. Section 3 deals with Yavadunam Tavadunikrtya Vargarica Yojayet (YTVY) sutra of ancient vedic mathematics. The proposed technique of Squaring circuit design without using multiplier is presented in section 4.Implementation and verification of the proposed technique is described in Section 5. Results and deliberations are presented in Section 6. At last the Conclusions are strained in Section 7.

2. Background and Related Work

Squaring operation has extensive applications in diverse areas of engineering and technology specifically in the area of VLSI Signal Processing. In this framework, numerous squaring techniques are proposed in the literature.

K.sethi & R.panda(2012) proposed a technique for implementing high speed squaring circuit for binary numbers. High speed Vedic multiplier is used for design of the proposed squaring circuit. The solution to the achievement is that they only uses one Vedic multiplier instead of four multipliers reported in the literature.

'Urdhva Tiryagbhyam' is a broad-spectrum multiplication method of vedic mathematics applicable to all suitcases of multiplications. Siba Kumar Panda, R.Das et.al (2015) used this algorithm to design vedic multipliers in VHDL environment. The advantage of this algorithm [2]is that partial products and their sums are calculated in parallel.

Srikanth G, NasamSai Kumar (2014) proposed a 32X32 bit reversible Vedic multiplier[7] using "Urdhva Tiryakabhayam" sutra meaning Vertical and crosswise, is designed using reversible logic gates, which is the first of its kind. Also in this paper they propose a new reversible unsigned division circuit. This circuit is designed using reversible components like reversible parallel adder, reversible left-shift register, reversible multiplexer, reversible n-bit register with parallel load line.

Kasliwal, Patil, and Gautam (2011) proposed a technique for implementing squaring operation using Vedic methods. In this method, two basic operations (multiplication and addition) are used. Concurrent operation of multipliers and use of first adders reduce the delay in Kasliwal et al. The demerit with the earlier proposed method by Kasliwalet al. (2011) is that they use four numbers of such Vedic multipliers to evaluate squaring of an n-bit binary number.

This has motivated us to exploreainnovative method for squaring of binary numbers without using multiplier circuits. In this paper, we propose anwell-organized method to find the square of binary numbers. For this operation we had used the concept of ancient Indian methods (Maharaja, 2009; Tirthaji, 1965; Vedic Mathematics, 2007).Ancient Indian Vedic mathematics was revitalized from Vedas by Sri BharatiKrisna Tirthaji (1884–1960) after his widespread research on Vedas (Maharaja, 2009; Tirthaji, 1965; Vedic Mathematics, 2007). In his book, it is manifest that Vedic mathematics is principally focused on 16 very important principles or word-formulae, which are well known as Vedic Sutras.Grippingly, the most important feature of the ancient Indian Vedic mathematics is its consistency. It is significant to mention here that the entire system of ancient mathematics is wisely organized and united. In Vedic mathematics, the general multiplication scheme can easily be reversed to achieve one-line division of decimal numbers. In the same way, the simple squaring method can easily be reversed to produce one-line square root of decimal number.The Vedic methods is very straightforward to learn. In the next section the application of ancient Indian Vedic formulae called 'Yavadunam

Tavadunikrtya Vargarica Yojayet' (Maharaja, 2009; Tirthaji, 1965) is explored which helps in design of low-power squaring circuit without using multipliers.

3. Yavadunam Tavadunikrtya Vargarica Yojayet

The meaning of this vedicsutra Sutra is 'whatever the deficiency subtract that deficit from the number and write alongside the square of that deficit'. This Sutra can be applicable to obtain squares of numbers close to bases of powers of 10.

Method-1 **:** Numbers near and less than the bases of powers of 10.

Example-1:9^2 Here base is 10.

The answer is separated in to two parts by a'/'

Note that deficit is 10 - 9 = 1

Multiply the deficit by itself or square it

1^2 = 1. As the deficiency is 1, subtract it from the number i.e., 9–1 = 8.

Now put 8 on the left and 1 on the right side of the vertical line or slash

i.e., 8/1.

Hence 81 is answer.

Example-2:96^2 Here base is 100.

Since deficit is 100-96=4 and square of it is 16 and the deficiency

subtracted from the number 96 gives 96-4 = 92, we get the answer 92 / 16

Thus 96^2 = 9216.

Example-3:994^2 Base is 1000

Deficit is 1000 - 994 = 6. Square of it is 36.

Deficiency subtracted from 994 gives 994 - 6 = 988

Answer is 988 / 036 [since base is 1000]

Example-4:9988^2 Base is 10,000.

Deficit = 10000 - 9988 = 12.

Square of deficit = 12^2 = 144.

Deficiency subtracted from number = 9988 - 12 = 9976.

Answer is 9976 / 0144 [since base is 10,000].

Example-5 :88^2 Base is 100.

Deficit = 100 - 88 = 12.

Square of deficit = 12^2 = 144.

Deficiency subtracted from number =88 - 12 = 76.

Now answer is 76 / 144 =7744 [since base is 100]

Algebraic proof of the numbers near and less than the bases of power of 10 can be treated as (xy),where x is the base and y, the deficit.Thus

(1) 9 = (10 -1) (2) 96 = (100-4) (3) 994 = (1000-6)
(4) 9988 = (10000-12) (v) 88 = (100-12)

$(x – y)^2 =x^2 – 2xy + y^2$
$= x (x – 2y) + y^2$
$= x (x – y – y) + y^2$
= Base (number – deficiency) + (deficit)^2

Thus

$985^2 = (1000 – 15)^2$
$= 1000 (985 – 15) + (15)^2$
= 1000 (970) + 225
= 970000 + 225
= 970225.

4. Proposed Technique for Squaring Circuit Design

4.1. Squaring Of Decimal Numbers

It is proved that the Vedic mathematics converts a complex calculation into a simple mean. The Vedic mathematics is developed in such a way that a student can learn can learn it very easily. So the computer Arithmetic can be developed with this way. In this squaring operation we are using 'Yavadunam Tavadunikrtya Vargarica Yojayet'. 'whatever the deficiency subtract that deficit from the number and write alongside the square of that deficit'. For example, calculating the square of 8.For that we need to follow the rules.

- For 8,the nearest base is 10.
- So the difference between 8 and 10 is: 10-8=2.So the difference is 2,the left most number will be (8-2=6).
- Now for right most it is "$(2)^2$=4 ".
- So the result will be as 8^2=(8-2) ×(10×1)+2^2=64.

The base number for thes 2 digit is as:

For 10(i.e. 10 ×1), 20 (i.e. 10 ×2) and so on till 90 (i.e. 10×9). So for the 35 ,it should be 35^2=(35-5) ×(10×4) + (5^2)=1225.The numbers greater than 10 , alternatively, expecting at the deficit we can go for the surplus. For example: 33^2=(33+3) ×(3×10)+33^2 =1089. To calculate the square of a single-digit number, the base should be 10.So the base number for 2-digit number can be 10^1 or 10^2. The fact is that the surplus or deficit is computed with the reference to the base. In consideration of the surplus, the same s added to the number and the result is multiplied by the base number. Further, the square of the deficiency as added to the final result. The base number for 2-digit number can be expressed as(1×10),(2×10),(3×10)….(9×10) or (1×100).Finally it is clear that the square of 95 is either declared as:95^2=(95-5) × (1×100) + 5^2=9095 or 95^2=(95+5) × (9×10) + 5^2=9095.The squaring method can be expressed as

$$S = (A+C)\,(A-C) + C^2 \qquad (1)$$

Where A is the given number whose square to calculate, C is the deficit (or surplus) from the nearest product of 10 (base number B) that is C = A − B or B − A.Figure2 shows the block diagram of a 2-bit squaring circuit.

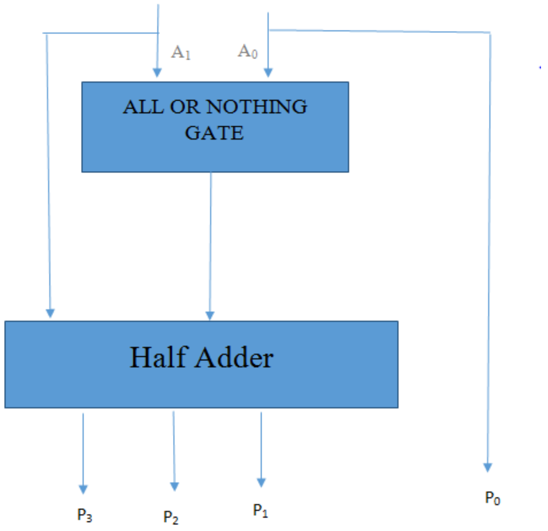


Figure 1: 2-bit Squaring Unit

4.2. Wing For Binary Number

The idea for squaring operation is extended for the binary number. The main reason in this implementation is to apply the Vedic methods for computer arithmetic. The DSP processors always deal with the binary numbers. In DSP, multiplication and squaring operation are used very often. For all this reason, squaring circuit is absolutely effective for various signals processing application. So in this we are developing an effective squaring circuit for binary numbers. Here we are implementing the simplest method as "Yavadunam Tavadunikrtya Vargarica Yojayet" to calculate of the decimal numbers.We can replace (A-C) by B as the base number in the equation (1), then we will get

$$A^2=(A+C)B + C^2 \qquad (2)$$

So to compute the square of any number, we need two adders and two multipliers. The equation (2) can be used to compute any bits of binary number. But here we are not using any multiplier for squaring. For any binary number, the base number is chosen as $2^{(n-1)}$ .For example :for 3-bit number , the base is "100".For a 6- bit number ,the base number is "100000" and so on. Multiplying a binary number is nothing but the left shift of the number. The base number always should be as $2^{(n-1)}$.when we compute the square of a n-bit number, the base number is $2^{(n-1)}$.So multiplying the base (B) by (A+C) is equivalent to the left shifting (n-1)-bits of (A+C).The shifting is done by concatenating (n-1) zeros at the LSB onwards. In such a way the multiplier circuit can be efficiently skipped. Now for the second term of the equation, we need a multiplier to compute C^2. But in this paper, we can skip the multiplier for computing C^2 , by using a squaring circuit . To compute the square of 2-bit number , one can use the multiplier. But in this paper, we have implemented a appropriate circuit for the squaring of 2-bit without using a multiplier. Figure 2 explains the algorithmic flow chart for squaring operation of binary numbers.

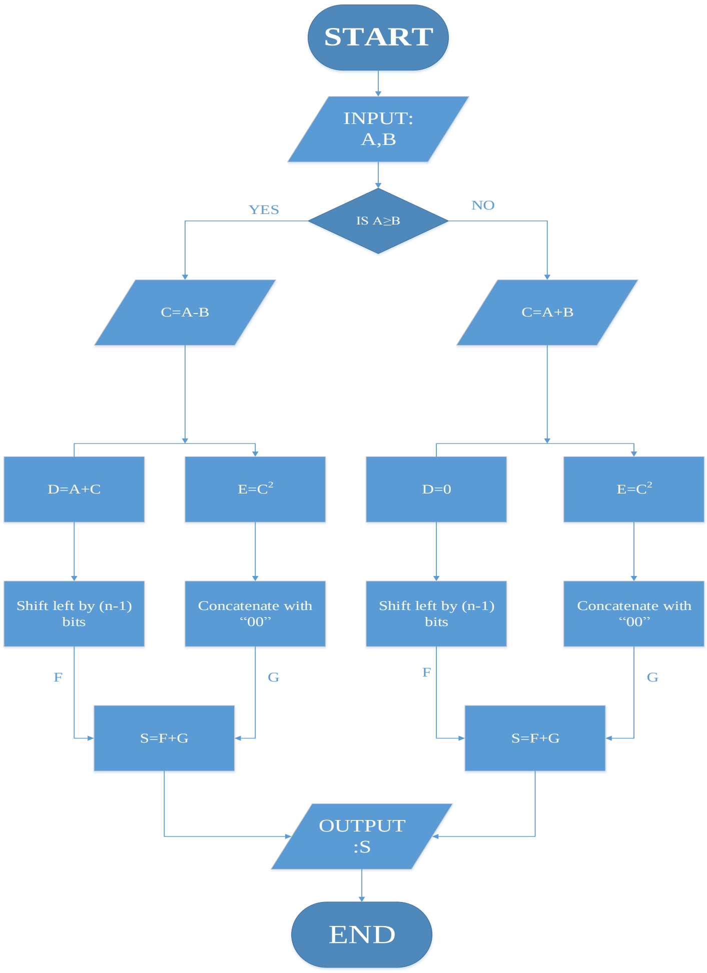


Figure 2: Algorithmic flow chart for squaring operation of binary number

This 2 bit squaring circuit can be made of one 2 – input AND gate and one half adder .As per the figure 1, A1A0 is the two bit binary number. Square of the 2-bit binary number is produced 4-bit number, G0G1G2G3.In our approach the 2-bit squaring circuit is recycled once to calculate the square of 3-bit number. Furthermore to calculate the square of 4-bit number, we recycle 3-bit squaring circuit and so on. Hence, to calculate the square of n-bit number, we need to recycle the (n-1) bit squaring circuit. So to calculate the square of n-bit number, we have to use the 2-bit squaring circuit for (n-2) times.So according to the equation (2), we need a adder/ subtractor for the addition/subtraction of A and B, depending on the condition.

Condition 1: DEFICIT

When A is greater than or same as B, we have to subtract B from A. Then C is added to A as the result (D). Again, there will be 2-bit (n-1) of left shifting to D as a result F. Then calculating (E) by squaring of C by using 2-bit squaring circuit. After that, the result E is concatenating 2-bit (n−1) number of zeroes at MSB as G.The final result S can be obtained by adding G and F.

Condition: 2 SURPLUS

When A is less than B, we have to add A and B to get C. Again excluding the MSB of the C , calculate the squaring by the 2-bit squaring circuit to get E. Then can concatenating 2-bit (n−1) number of zeroes at MSB as G. After that assign D as "0000" and there will be 2-bit (n-1) of left shifting to D as a result F. And the final result will be S by adding F and E by the 6-bit (2n) binary adder.

For EXAMPLE:

Case-I, when A ≥ B. Let A is 111 and B is 100.

Step-1: 100(B) is subtracted from 111(A).

111(A)
(−) 100(B)
- - - - - - - - - - - -
011(C)

Step-2: The square of C (excluding MSB) is (11)2 = 1001(E)

Step-3: 011(C) is added with 111(A).

111(A)
(+) 011(C)
- - - - - - - - - - - -
1010(D)
- - - - - - - - - - - -

Step-4: 1010 (D) & 00 gives 101000 and 00 & 1001(E) gives 001001.

Note that the symbol '&' represents the concatenation operation. Here, the sum (A + C) bits are left shifted by 2-bits. Because n-1 = 2,where n is the number of bits in A.

Step-5: Addition of 101000 (input1 of 2n-bit adder) and 001001 (input2 of 2nbitadder) gives

101000
(+) 001001
- - - - - - - - - - - -
110001(S)

Hence, the square of 111(A) is 110001(S).

Case-II, when A<B. Let A is 010 and B is 100.

Step-1: 010(A) is added with 100(B).

010(A)
(+) 100(B)

- - - - - - - - - - - - -

110(C)

Step-2: The square of C leaving aside MSB (i.e., 10) using 2-bit squaring circuit.

(10)2 = 0100(E)

Step-3: 0000 is assigned to D.

Step-4: 0000 (D) & 00 gives 000000 and 00 & 0100(E) gives 000100.

Note that the symbol ‘&’ represents the concatenation operation.

Step-5: Addition of 000000 (F) and 000100(G) gives

000000(F)

(+) 000100(G)

- - - - - - - - - - -

001001(S)

Hence, the square of 011(A) is 001001(S)

5. Implementation and Verification

In this exertion, the squaring circuit is designed inVHDLenvironment(Pedroni, 2008).Then it is successfully Synthesisedand Simulated in Xilinx ISE Design Suite 14.1.After that the power analysis is done with the help of FPGA device (XC3S100E) spartan3.The results are verified and found correct.

6. Result Analysis

This section mainly focuses on the simulation result part. The device utilization summary is displayed in Table 1 which shows that three number of slices are used and number of four input LUTs are 5.Then by means of power analyzer report which shows that it consumes 0.032w power which is very less as compared to[3].The below figure 3.1 and 3.2 shows the RTL schematic of the designed Squaring circuit by YTVY algorithm of vedic mathematics.

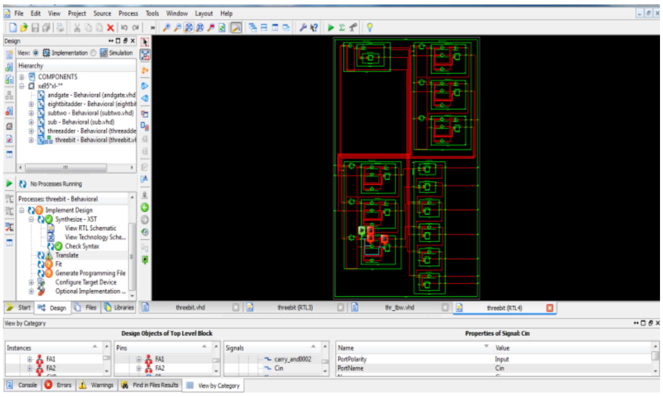

File Edit View Project Source Process Tools Window Layout Help
Design
Hierarchy
COMPONENTS
andgate - Behavioral (andgate.vhd)
eightbitadder - Behavioral
subtwo - Behavioral (subtwo.vhd)
sub - Behavioral (sub.vhd)
threeadder - Behavioral
threebit - Behavioral
No Processes Running
Processes: threebit - Behavioral
Implement Design
Synthesize - XST
View RTL Schematic
View Technology Schematic
Check Syntax
Translate
Fit
Generate Programming File
Configure Target Device
Optional Implementation
Design Objects of Top Level Block
Properties of Signal: Cin
Instances
FA1
FA2
Pins
Signals
carry_andD002
Cin
Name
Value
PortPolarity
Input
PortName
Cin
Console Errors Warnings Find in Files Results View by Category

Figure 3.1: RTL view of Squaring circuit

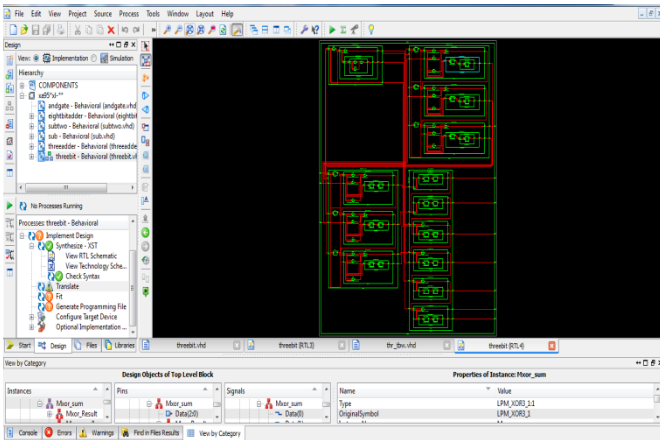
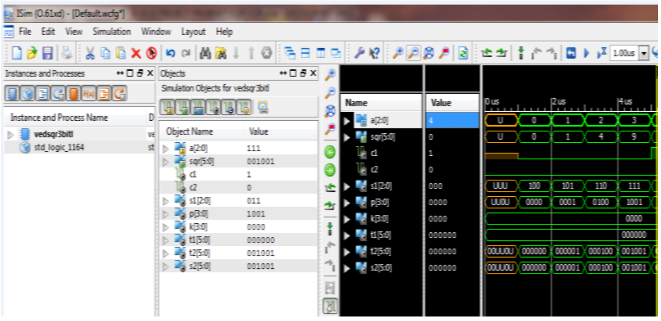


Figure 3.2: RTL view of Squaring circuit(Internal arrangement)

The below figure 4 shows the Test bench waveform of the designed Squaring circuit.


ISim (O.61xd) - [Default.wcfg]
File Edit View Simulation Window Layout Help
Instances and Processes
Instance and Process Name
vedsqr3bit
std_logic_1164
Objects
Simulation Objects for vedsqr3bit
1.00us
Name
Value
0 us
2 us
4 us

Object Name	Value
a[2:0]	111
sqr[5:0]	001001
c1	1
c2	0
s1[2:0]	011
p[3:0]	1001
k[3:0]	0000
t1[5:0]	000000
t2[5:0]	001001
s2[5:0]	001001

Name	Value					
a[2:0]	4	U	0	1	2	3
sqr[5:0]	0	U	0	1	4	9
c1	1					
c2	0					
s1[2:0]	000	UUU	100	101	110	111
p[3:0]	0000	UUUU	0000	0001	0100	1001
k[3:0]	0000					0000
t1[5:0]	000000					000000
t2[5:0]	000000	00UUOU	000000	000001	000100	001001
s2[5:0]	000000	00UUOU	000000	000001	000100	001001

Figure 4: Test bench waveform of Squaring operation

The figure 5 and figure 6 shows the graph between power vs input voltage and power vs temperature respectively. From the graph it reveals that the power consumption is constant throughout the input voltage that is 0.032w .

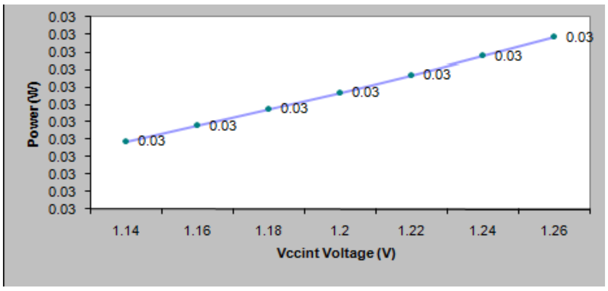


Figure 5: Power vs input voltage analysis

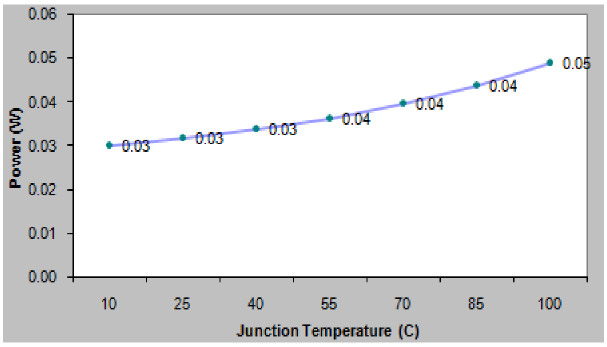


Figure 6: Power vs junction Temperature analysis

No. of slices	3
No. of 4 input LUTs	5
No. of bonded IOBs	10
Power Consumption	0.032W

Table 1-Device Utilization in Designed Squaring Circuit

7. Conclusions

The performance of the proposed squaring circuit using YTVY algorithm proved to be efficient in terms of power consumption. Due to its customary and parallel structure, it can be realized easily on silicon as well. Theinitiativeprojected here may set path for future research in this direction. Futureextent of research is to condense area requirements. The Performance of the Proposed method in terms of Simulation results have been presented which shows that the proposed method is more superior in terms of power consumption. Hence our method out performs its counterparts.

References

1. Maharaja, J.S.S.B.K.T., “Vedic mathematics,” Motilal Banarsidass Publishers Pvt. Ltd, Delhi, 2009
2. SibaKu. Panda, R.Das et.al “VLSI implementation of Vedic Multiplier using Urdhva-Tiryakbhyam sutra in VHDL environment: A Novelty”, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), Volume 5, Issue 1, Ver. III (Jan - Feb. 2015), PP 17-24.
3. K.sethi & R.panda “ An improved squaring circuit for binary numbers”, International Journal of Advanced Computer Science and Applications, page111–116 , 2012
4. Taek-Jun Kwon, Jeff Sondeen, Jeffrey Draper, “Floating-Point Division and Square Root using a Taylor-Series Expansion Algorithm”, Proceedings of the 50th IEEE International Midwest Symposium on Circuits and Systems, August 2007, pp. 305 - 308.
5. Swami BharatiKrisna Tirtha, “Vedic Mathematics,” Motilal Banarsidass Publishers, Delhi, 1965.
6. B. Parhami, “Computer Arithmetic Algorithms and Hardware Architectures,” 2nd ed, Oxford University Press, New York, 2010
7. Srikanth G, NasamSai Kumar” Design of High speed Low Power Reversible Vedic multiplier and Reversible Divider” IJERAISSN : 2248-9622, Vol. 4, Issue 9(Version 5), September 2014)
8. P.D. Chidgupkar, and M.T. Karad, “The Implementation of Vedic Algorithms in Digital Signal Processing,” Global Journal of Engng. Educ., vol.8 , pp.153-158, 2004
9. Prabha S., Kasliwal, B.P. Patil and D.K. Gautam, “Performance Evaluation of Squaring Operation by Vedic Mathematics”, IETE Journal of Research, vol.57, Issue 1, Jan-Feb 2011
10. V.A. Pedroni, “Circuit Design with VHDL,” 2008
11. Xilinx ISE User Manual’, Xilinx Inc, USA, 2007

Author Profile

Mr. Pabitra Kumar Mohapatra was born in Balasore, Odisha , India, in 1988. He has completed B.Tech in Electronics and Communication Engineering, from KNSIT (VTU), BANGLORE, kARNATAKA .Currently pursuing his M.tech degree in VLSI from ECE , Centurion University of Technology and Management, Bhubaneswar, Odisha 751020, India, in the academic period of 2013-2015.

Mr. Siba Kumar Panda was born on November 09,1989.He received the B.Tech degree in Electronics & Communication Engineering from BijuPatnaik University of Technology, odisha in 2012 andM.Tech degree in *VLSI Signal Processing* Specialization from Veer Surendra Sai University Of Technology, odisha in 2014.

Currently he is working as an Assistant Professor at Centurion University of Technology and Management, Bhubaneswar; Odisha. He also awarded the University Silver medal for best Electronics & Telecommunication Engineering Post Graduate for the academic year 2012-2014 at VSSUT, Odisha.

His research area of interest includes Ultra-wideband (UWB) device design, RF circuit design using CMOS Technique, VLSI implementation of Vedic mathematics, VLSI Signal Processing.

Mrs. Sambita Dalal currently working as Assistant Professor in ECE Department, CUTM, Jatani, Odisha. She have completed M.Tech in VLSI Design in 2012 and B.Tech in Applied Electronics and Instrumentation Engineering in 2010. Her Research area of interest is Signal and Image Processing, Soft computing and VLSI. She have contributed three International journal papers and two International conference paper in the field of research.