ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

ECC ENCRYPTION SYSTEM USING ENCODED MULTIPLIER AND VEDIC MATHEMATICS

Bonifus PL[1], Dani George[2]

Asst. Professor, Dept. of ECE, Rajagiri School of Engineering & Technology, Kochi, Kerala, India [1]

PG Student [VLSI& ES], Dept. of ECE, Rajagiri School of Engineering & Technology,Kochi, Kerala, India[2]

ABSTRACT: This paper presents an efficient design and implementation of ECC Encryption System using Encoded Multiplier. ECC algorithm is implemented based on ancient Indian Vedic Mathematics. The speed of the system mainly depends on multipliers and adders. To improve the speed of the system, the multiplier architecture is modified using a new encoded algorithm. Using this algorithm number of partial products in the multiplier architecture is reduced to half and thus it speeds up the operation. Effectively no multipliers are required and number of adders required is reduced drastically. The most significant aspect of this paper is the development of encoded architecture and embedding it in Point Multiplication circuitry of ECC algorithm. The coding is done in Verilog HDL and FPGA implementation using Xilinx Spartan 6 library.

Keywords:ECC, Encryption, Decryption, Vedic Mathematics, Encoder, Cryptography, Point Addition, Point Doubling, Scalar Multiplication.

I.INTRODUCTION

Cryptography is a technique for making the message secure. Sensitive information can be stored or transmitted across insecure network so that unauthorized persons cannot access it. Cryptography is implemented by means of various encryption algorithms. These encryption algorithms are classified into symmetric key algorithms (private key) and asymmetric key algorithms (public key). Private key cryptography uses one key shared by both sender and receiver. Public key cryptography uses two keys one for encryption and one for decryption.

In public key algorithms use of elliptic curves was proposed by Victor Miller and Neal Koblitz in 1980[2]. Its small key sizes make it a most powerful algorithm for cryptography while comparing to standard algorithms such as RSA. It is commonly used in security protocols such as IP data security, transport data security, email security, terminal connection security, conferencing service security, etc. The smaller key size reduces the power consumption and increases the speed of the cryptographic system. The major time consuming arithmetic operations in ECC are point addition and point doubling.

In encryption systems and most digital signal processing applications such as FFT and convolution multipliers play fundamental role in its computation. Implementing high speed systems with low power consumption and time delay mainly depends on multiplier execution time. Comparing to conventional multiplication Vedic method of multiplication requires very less number of operations resulting in a faster and high performance multiplier.

This paper describes a new multiplier architecture using encoder that requires half the number of operations than Vedic Multipliers. ECC algorithm using Vedic mathematics and encoder multiplier architecture makes the encryption system more efficient.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

II.MULTIPLIER USING ENCODER

Multiplier is the core component for most applications such as digital signal processing, encryption and decryption algorithm in cryptography and in other logical computations. Performance of a system mainly depends on the speed of the multiplier. Array multipliers and booth multipliers are the most common multipliers used in digital hardware. To improve the speed and power consumption of multipliers, many studies have taken place and are going on.

Vedic method of multiplication using ancient Indian Mathematics is much simpler and easier to understand. Swami Bharati Krishna TirthajiMaharaj reintroduced Vedic mathematics into the world. According to his research all mathematical operations are based on sixteen sutras and thirteen sub-sutras. It is used in all arithmetical operations such as multiplication, squaring, cubing, quadratic equations, etc. Vedic multipliers [6] designed using Urdhvatiryakbhyam sutra and Nikhilam sutras are some of the fastest multipliers. Vedic multiplier using Urdhvatiryakbhyam sutra and array multipliers have almost the same architecture. For eight bit multiplication the number of partial products required for both is eight. So large number of adder circuits are required to find the final product. The speed of a multiplier can be improved by either using fast addition algorithms or by reducing the number of partial products.

Multiplier using encoded algorithm [1] illustrates a new architecture for multiplication. The number of partial products generated by using this algorithm is half compared to Vedic and conventional methods. For eight bit multiplier the number of partial products will be four. The number of adder circuits required will be drastically reduced. This results in having multipliers with minimal adder circuits and faster adder algorithms.

ENCODING TECHNIQUE: In this technique the multiplier bits are grouped in a combination of 2-2 bits starting from LSB. The grouping of bits is shown in Fig 1[1]. Each group is given to the encoder circuit. The output from the encoder is based on the encoder table shown in Table 1[1].

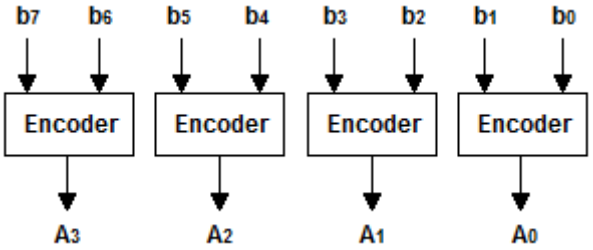

Fig. 1 Grouping bits using Encoder

b_{i+1}	b_i	A_i
0	0	0
0	1	1
1	0	2
1	1	3

Table 1 Encoder output

ENCODING ALGORITHM:

1. If A_i is 0 then partial product P_i is 0.
2. If A_i is 1 then partial product P_i is the multiplicand.
3. If A_i is 2 then partial product P_i is obtained by shifting the multiplicand one bit left.
4. If A_i is 3 then partial product P_i is the sum of partial products of A_i for i = 1 and 2.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

ENCODING STEPS:

1. Group the multiplier into 2 bits each starting from the LSB.
2. Find the value of A_i from the encoder table.
3. Find the partial products based on the value of A_i.
4. Partial products are given to the adder circuit with a shift of one bit, two bit, four bit, six bit one by one.
5. The adder circuit gives the final product.

For example 10101001 x 11001001 works as follows
Group the multiplier bits starting from LSB

Multiplier bits	11	00	10	01
A_i	3	0	2	1

From the algorithm the partial products are

A_i	Partial Products
01	10101001
10	101010010
00	00000000
11	111111011

Shifting the partial products by 2, 4 and 6 and given to the adder circuit to get the final product. The way multiplication is performed is shown below. Here the partial products are written one below the other after the required shifting.

								1	0	1	0	1	0	0	1
					1	0	1	0	1	0	0	1	0		
				0	0	0	0	0	0	0	0				
	1	1	1	1	1	1	0	1	1						
1	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1

ENCODED MULTIPLIER ARCHITECTURE:Fig. 2 shows the block diagram of the multiplier architecture. The input to the encoder circuit is the multiplicand and the multiplier. Partial products generated from the encoded circuit is given to shift register and then to the adder circuit. Adder circuits are optimized by using carry save adders. Fig. 3 shows the block diagram of the proposed adder circuit for eight bit multiplication.

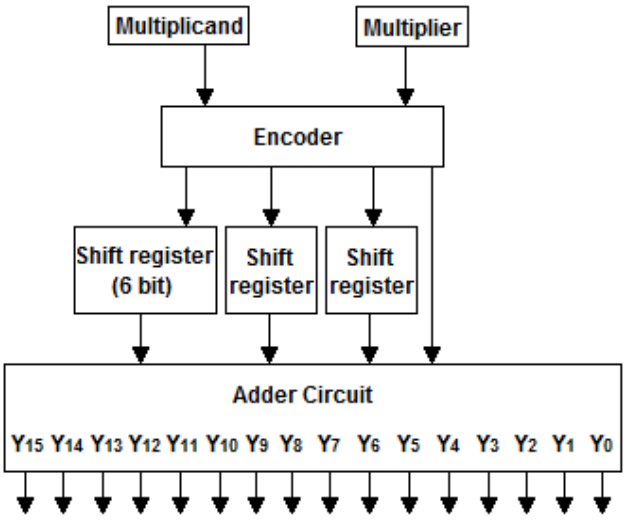

Fig. 2 Encoded Multiplier Architecture for 8 bit Multiplier

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

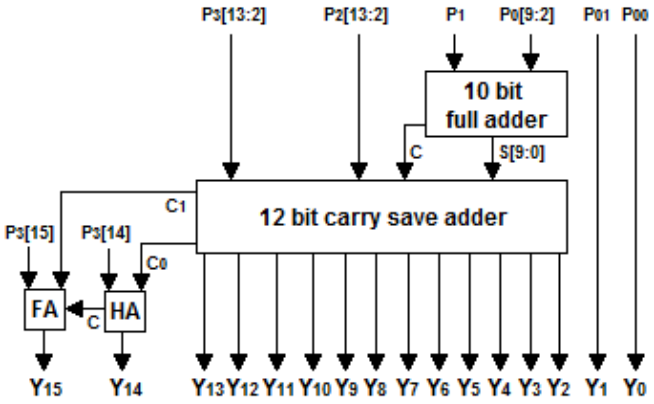

Fig. 3 Block Diagram of Adder circuit in 8 bit Encoded Multiplier

COMPARISON WITH VEDIC MULTIPLICATION ALGORITHM:Vedic multiplication involves both multiplication and addition. In encoded multiplier architecture multiplier circuit is not used. Partial products are generated directly from the proposed algorithm. The comparison result is shown in Table 2. It shows a large reduction in hardware structure. So power consumption, cost and delay are less.

Bit size	Vedic Multiplier		Encoded Multiplier	
	M	***A***	***M***	***A***
8	64	97	–	42
16	256	413	–	92

Table 2 Comparison of Encoded Multiplier using Vedic Multiplier

III. ELLIPTIC CURVE ARITHEMATIC

An Elliptic Curve is defined as an equation having set of solutions along with the point at infinity. The elliptic curve equations $y^2+xy = x^3 + ax + b$ in GF(2m) and $y^2 = x^3 + ax + b$ in GM(p) are called Weierstrass equations [4]. Variables and coefficients are chosen from a large finite field. These points form a group. The group operations for elliptic curve cryptography are point multiplication, point addition and point doubling. Point multiplication means multiplying a point P(x,y,z) with a scalar value k, i.e, Q = kP. The security of ECC in cryptography is based on finding the value of k if P and Q are given. This is called Elliptic Curve Discrete Logarithmic Problem. For large values of k it is computationally infeasible. Point Multiplication is the main operation in ECC [3].

POINT DOUBLING IN PROJECTIVE COORDINATES:To double a point P, i,e. Q = 2P

Let $2(X_1, Y_1, Z_1) = (X_3, Y_3, Z_3)$ then

$Z_3 = X_1^2 . Z_1^2$

$X_3 = X_1^4 + b . Z_1^4$

$Y_3 = b . Z_1^4 . Z_3 + X_3.(a . Z_3 + Y_1^2 + b.Z_1^4)$

POINT ADDITION IN PROJECTIVE COORDINATES:To add two points in a curve, i.e L = J + K

Let $(X_1, Y_1, Z_1) + (X_2, Y_2, 1) = (X_3, Y_3, Z_3)$ then

$A = Y_2 . Z_1^2 + Y_1$

$B = X_2 . Z_1 + X_1$

$C = Z_1 . B$

$Z_3 = C^2$

$D = B^2.(C + a . Z_1^2)$

$E = A.C$

$X_3 = A^2 + D + E$

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

$F = X_3 + X_2 . Z_3$
$G = X_3 + Y_2 . Z_3$
$Y_3 = E . F + Z_3 . G$

DOUBLE AND ADD ALGORITHM FOR POINT MULTIPLICATION:

1. The base point on the curve P(x,y) is given as the input.
2. Scalar value k = (k_{m-1}, k_{m-2}, …k_0).
3. Another point on the curve Q = kP will be the output

Algorithm to compute Q = kP

```
Q = P
fori = m-2 down to 0
    do Q = 2.Q
    if  k_i =1
      do Q = Q + P
end
```

From this algorithm we can see implementing point addition and point doubling for point multiplication requires multiplication operation and squaring.

IV. SQUARING USING VEDIC MATHEMATICS

For squaring, a dedicated architecture can improve its performance than using multiplier architecture. Using Duplex D property of binary numbers from the sutra Dwandwayoga of Vedic Mathematics algorithm for squaring is implemented [4].

1. Duplex of a number is twice that number, Duplex of a is a^2
2. Duplex of two numbers is multiplying two with the product of that number, Duplex of ab is 2*a*b
3. Duplex of three numbers is multiplying the product of the outer most pair with two plus the square of the middle number, Duplex of abc is $2*a*c+b^2$

Example: $12431^2 = 154529761$

No	Duplex
1	1*1 = 1
12	2*1*2 = 4
124	2*1*4 + 2*2 = 12
1243	2*1*3 + 2*2*4 = 22
12431	2*1*1 + 2*2*3 + 4*4 = 30
2431	2*2*1 + 2*4*3 = 28
431	2*4*1 + 3*3 = 17
31	2*3*1 = 6
1	1*1 = 1

1/4/12/22/30/28/17/6/1 can be written in the form shown below and finally find their sum

1	4	2	2	0	8	7	6	1	+
0	1	2	3	2	1	0			
1	5	4	5	2	9	7	6	1	

V. IMPLEMENTATION RESULTS

Point Addition, Point doubling and Scalar Multiplication are done in Verilog. The code is synthesized using Xilinx 12.1 to verify the functionality. Simulation results for Point Doubling, Point Addition and Scalar Multiplication is shown in Fig 4, 5 and 6

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

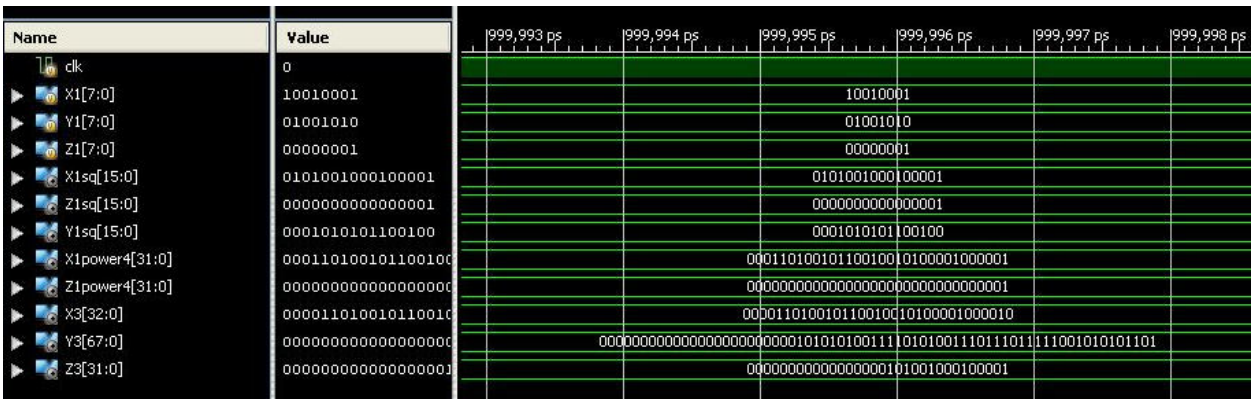

Fig. 4 Point Doubling

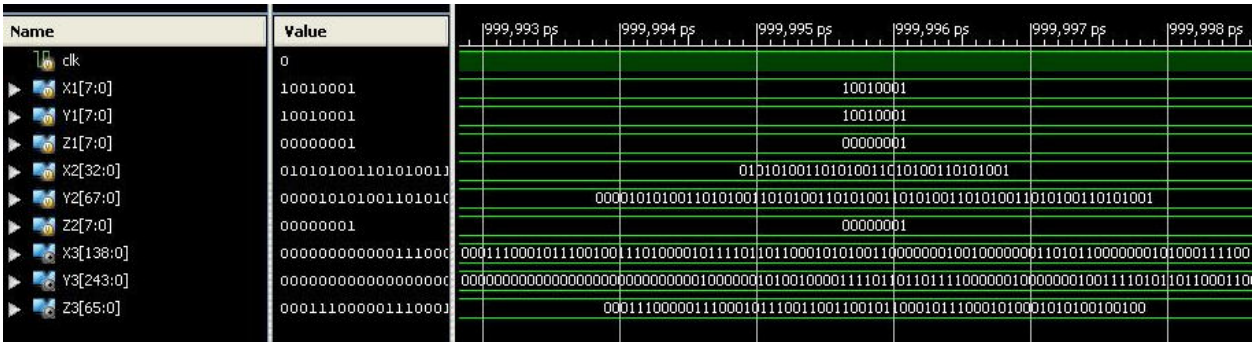

Fig. 5 Point Addition

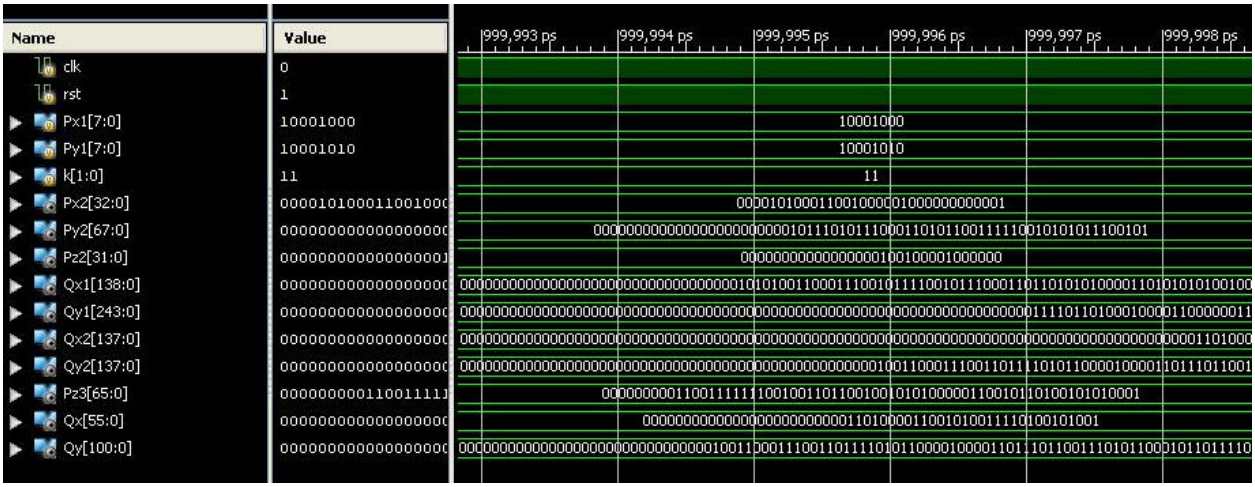

Fig. 6 Scalar Multiplication

Comparative study of point addition and point doubling using different multipliers such as Vedic, Booth and Array are done. Fig.7 shows the delay comparison of point addition and point doubling using different multipliers. Fig.8 shows the comparison results of occupied slice LUTs.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

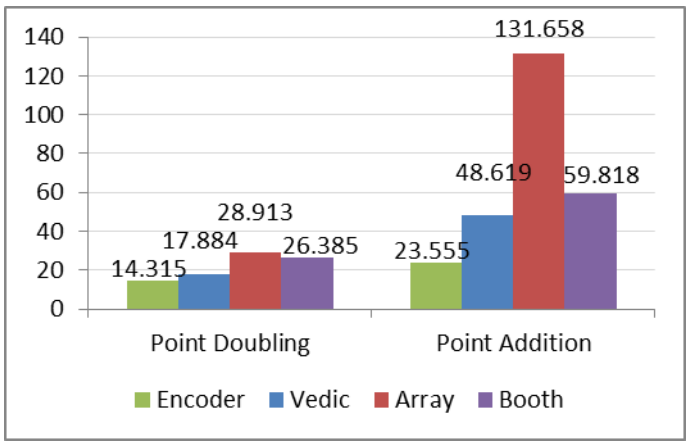

Fig. 7Comparison of delay using different multipliers in ECC Algorithm

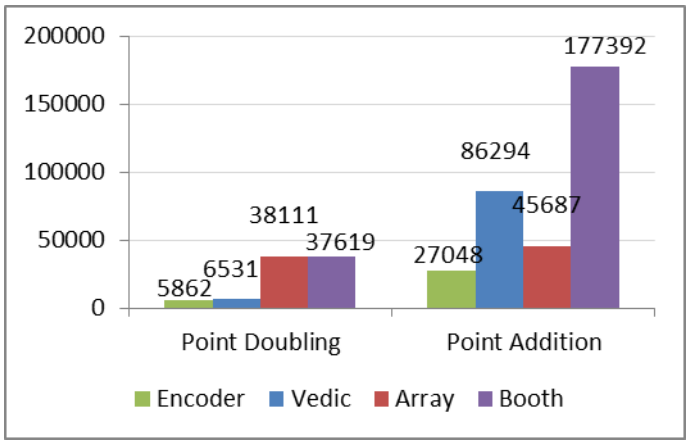

Fig. 8 Comparison of slice LUTs using different multipliers in ECC Algorithm

VI.CONCLUSION

The aim of this project is the development of high performance ECC encryption system. The proposed multiplier architecture achieves a significant improvement in performance. The encoded multiplier having only shift registers and adder circuits thus reduces the complexity, cost, power consumption and delay. The Point Addition using various multipliers was taken for comparison and the implementation using encoder multiplier was found to be 2.06 times faster than Vedic multiplier, 2.54 times faster than Booth multiplier and 5.59 times faster than Array multiplier. While comparing the Point Doubling the implementation using encoder multiplier was found to be 1.25 times faster than Vedic multiplier, 1.84 times faster than Booth multiplier and 2.02 times faster than Array multiplier. The usage of slice LUTs isalso found to be more efficient with encoder multiplier. In Point Addition encoder multiplier uses 219% lesser slice LUTs compared to Vedic multiplier, 556% lesser slice LUTs compared to Booth multiplier and 69% lesser slice LUTs compared to Array multiplier. In Point Doubling encoder multiplier uses 11% lesser slice LUTs compared to Vedic multiplier, 542% lesser slice LUTs compared to Booth multiplier and 550% lesser slice LUTs compared to Array multiplier.

REFERENCES

[1] Jai Skand Tripathi, Priya Keerti Tripathi, Deepti Shakti Tripathi, "An Efficient Design of Vedic Multiplier Using New Encoding Scheme" International Journal of Computer Applications (0975-8887) Volume 53-No 11,September 2012.

[2] Jian Huang, Hao Li, and Phil Sweany "An FPGA Implementation of Elliptic Curve Cryptography for Future Secure Web Transaction", University of North Texas.

[3] Anoop MS " Elliptic Curve Cryptography - An Implementation Tutorial".

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 11, November 2013

[4] Himanshu Thapliyal, M.B Srinivas "A High Speed and Efficient Method of Elliptic Curve Encryption Using Ancient Indian Vedic Mathematics ", International Institute of Information Technology.
[5] Erlangung des Doktorgrades (Dr. rer. nat.), Mathematisch-Naturwissenschaftlichen Fakult¨at, Rheinischen Friedrich-Wilhelms-Universit¨at Bonn"Efficient Implementation of Elliptic Curve Cryptography on FPGAs", Jamshid Shokrollahi Tehran, Iran, 2006
[6] Harpreet singh Dhillon & Abhijiit Mitra, "A Digital Multiplier Architecture using Urdhava Tiryakbhyam Sutra of Vedic Mathematics" IEEE Conference Proceeding,2008.
[7] Pushpalata Verma, "Design of 4*4 bit Multiplier using EDA Tool", Vol 48 International journal of Computer Application,2012.
[8] Sumit Vaidya & Deepak Dandekar, "Delay-Power performance comparison of Multipliers in VLSI circuit design", International journal of Computer Networks & Communications, July 2010.