RESEARCHERID
THOMSON REUTERS

[Potharaju, 4(11): November 2017] ISSN 2348 – 8034
DOI- 10.5281/zenodo.1067986 Impact Factor- 4.022

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

FPGA IMPLEMENTATION OF ELLIPTIC CURVE DISCRETE LOGARITHMUSING VERILOG HDL

Vidya Sagar Potharaju*

*Associate Professor, Department of Electronics and Communication Engineering,
Vignana Bharathi Institute of Technology, Hyderabad, Telangana,INDIA

ABSTRACT

Elliptic Curve Discrete Logarithm (ECDL) are most popular choice Elliptic Curve Cryptography (ECC),which gives provision for shorter key lengths as compared to as compared to its counterpart public key cryptosystems, and it can be used for security in embedded systems,wirless communications and personal communication systems. In this paper Elliptic Curve Discrete Logarithm code has been written in Verilog Hardware Description Language (HDL) and implemented on Xilinx Spartan3E Field Programmable Gate Array (FPGA),has taken 403 encoders, decoders with minimum period of 5.043 ns,maximum frequency 198.295 MHz and with a total memory usage of 269824 Kilobytes respectively. The performance of the crypto system is much faster than the software implementation of the same system.

Keywords: *Elliptic Curve Discrete Logarithm (ECDL),VerilogHardware Description Language (HDL), Elliptic Curve Cryptography (ECC),Field Programmable Gate Array (FPGA),Finite Fields.*

I. INTRODUCTION

The idea of information security lead to the evolution of Cryptography. In other words, Cryptography is the science of keeping information secure. It involves encryption and decryption of messages. Encryption is the process of converting a plain text into cipher text and decryption is the process of getting back the original message from the encrypted text. Cryptography, in addition to providing confidentiality, also provides Authentication, Integrity and Non-repudiation. There have been many known cryptographic algorithms.

Based on the key, cryptosystems can be classified into two categories: S*ymmetric* and A*symmetric*. In Symmetric Key Cryptosystems, we use the same key for both Encryption as well as the corresponding decryption. i.e. if K was the key and M was the message, then, we have D_K (E_K(M)) = Asymmetric or Public key or shared key cryptosystems use two different keys. One is used for encryption while the other key is used for decryption. The two keys can be used interchangeably. One of the keys is made public (shared) while the other key is kept a secret. i.e. let k1 and k2 be public and private keys respectively. Let M be the message, then D_{k2} (E_{k1} (M)) = D_{k1} (E_{k2} (M)) = M. In general, symmetric key cryptosystems are preferred over public key systems due to the following factors:

1. Ease of computation
2. Smaller key length providing the same amount of security as compared to a larger key in Public key systems.

The idea of using Elliptic curves in cryptography was introduced by Victor Miller and Neal Koblitz as an alternative to established public-key systems such as DSA and RSA. The Elliptical curve Discrete Log Problem (ECDLP) makes it difficult to break an ECC as compared to RSA and DSA where the problems of factorization or the discrete log problem can be solved in sub-exponential time. This means that significantly smaller parameters can be used in ECC than in other competitive systems such as RSA and DSA. This helps in having smaller key size hence faster computations.

[Potharaju, **4(11): November 2017]** **ISSN 2348 – 8034**
DOI- 10.5281/zenodo.1067986 **Impact Factor- 4.022**

Elliptical Curve Discrete Logarithm Problem:

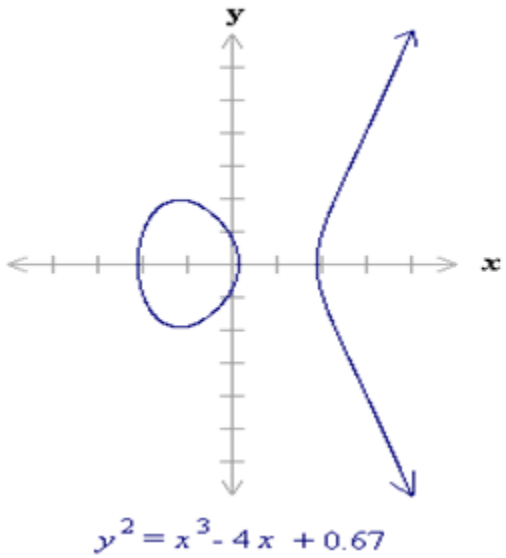
The strength of the Elliptic Curve Cryptography lies in the Elliptic Curve Discrete Log Problem (ECDLP). The statement of ECDLP is as follows.

Let E be an elliptic curve and P ∈ E be a point of order n. Given a point Q ∈ E withQ = mP, for a certain m ∈{2, 3, ……, m – 2}.Find the m for which the above equation holds.

When E and P are properly chosen, the ECDLP is thought to be infeasible. Note that m = 0, 1 and m – 1, Q takes the values ***O***, P and – P. One of the conditions is that the order of P i.e. n be large so that it is infeasible to check all the possibilities of m.

The difference between ECDLP and the Discrete Logarithm Problem (DLP) is that, DLP though a hard problem is known to have a sub exponential time solution, and the solution of the DLP can be computed faster than that to the ECDLP. This property of Elliptic curves makes it favorable for its use in cryptography.

Elliptic Curves:An *Elliptic Curve* is a set of point on a curve $y^2=x^3+ax+b$ given certain real numbers$a$and$b$. For example**Figure 1.**



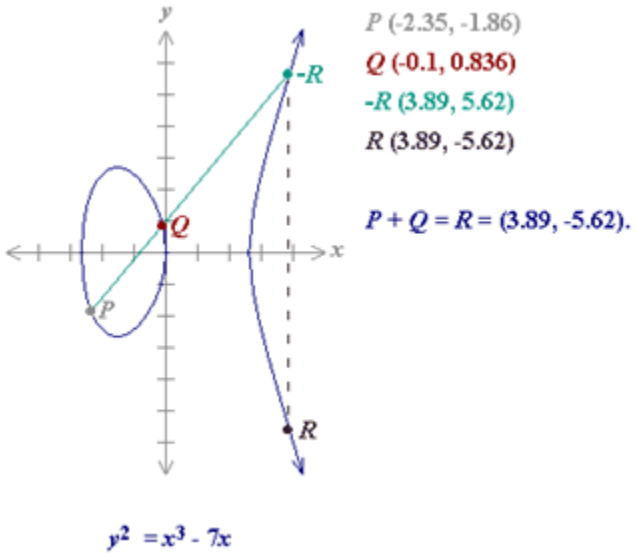


Figure 1: An Elliptic Curve Defined Over RFigure 2: Adding Two Points on an Elliptic Curve

***Elliptic Curve Groups*:** The set of points on an elliptic curve, plus a special point ∞ form and additive group. The addition of two points on an elliptic curve is defined geometrically, as shown in the **Figure 2**.

ISSN 2348 – 8034
Impact Factor- 4.022

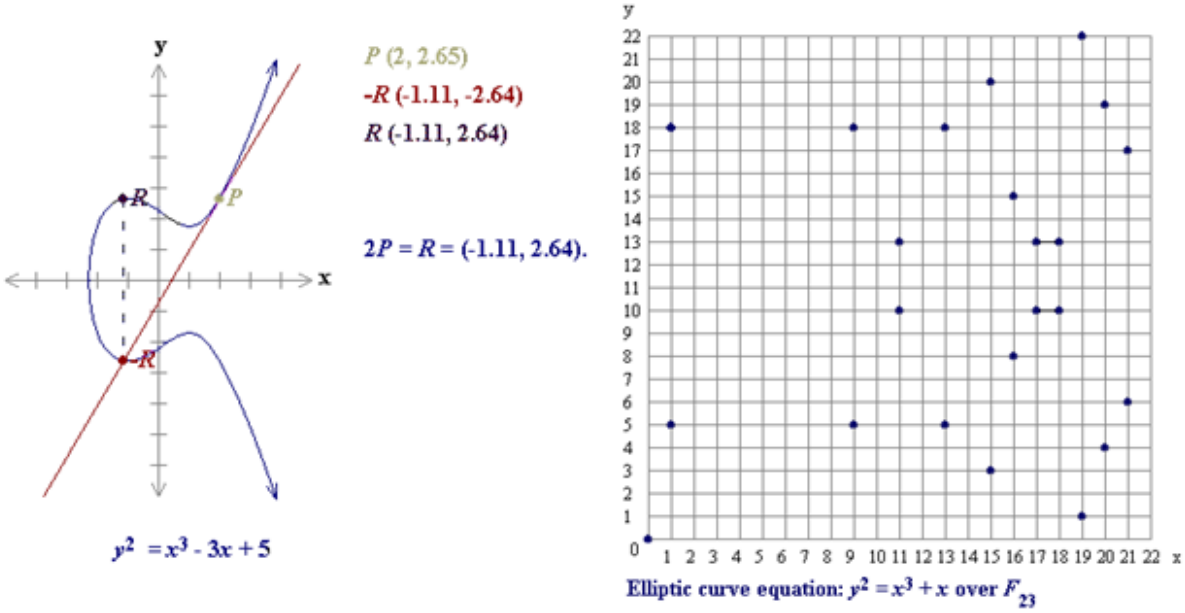


Figure 3 :Doubling a Point on an Elliptic CurveFigure 3 : Elements of this ECG

Elliptic Curve Encryption Algorithmsdepend on the difficulty of calculating$k^P$where ka product of two large primes is and P is an element in the Elliptic Curve Group. Geometrically to add a point P to itself you first construct the tangent line to the curve at the point. Then the line will intersect the curve at only one point, and the addition of $2P$is then defined to be the negative of the point of intersection as seen in figure.Elliptic curve groups over real numbers are not practical for cryptography due to slowness of calculations and round-off error. This Elliptic Curves Over Finite Fields are used. An elliptic curve over a finite field $\mathbf{F_p}$ of characteristic greater than three can be formed by choosing the variables $\boldsymbol{a}$and$\boldsymbol{b}$within the field$\mathbf{F_p}$.

Roughly speaking the elliptic curve is then the set of points $\boldsymbol{(x, y)}$ which satisfy the elliptic curve equation $\boldsymbol{y^2=x^3+ax+b}$ modulo$\boldsymbol{p}$, where$\boldsymbol{x, y \epsilon F_p}$; together with a special point$\boldsymbol{\alpha}$. If $\boldsymbol{x^3+ax+b}$ contains no repeated factors, or equivalently if$4a^3+27b^2 \equiv 0 \ (mod\ p)$, then these points form a group.

It is well known that EGC (the Elliptic Curve Group) is an additive abelian group with $\boldsymbol{\alpha}$ serving as its identity element.

Example: In the ECG of $y^2=x^3+x^2$ over the field F_{23} the point $(9, 5)$ satisfies the equation $y^2 \equiv x^3+x^2 \ (mod 23)$ as $25 \equiv 729+9 \ (mod 23)$. The elements of this ECG are given in the pictured above.

Obviously we no longer have a curve to define our addition geometrically. Emulating the geometric construction for addition, the formulas for addition over $\boldsymbol{F_p}$ (characteristic 3) are given as follows: Let $P(x1, y1)$ and $Q(x2, y2)$ be elements of the ECG. Then $P+Q = (x3, y3)$, where

$$x_3 = \lambda^2 \text{and} x_1 - x_2 \qquad y_3 = \lambda(x_1 - x_3) - y_1 \qquad \lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & if \quad P \neq Q \\ \dfrac{3x_1^2 + a}{2y_1} & if \quad P = Q \end{cases}$$

***Definition of the Discrete Logarithm Problem*:**

In the multiplication group $_{Fp}^{x}$, the discrete logarithm problem that is: Given elements$r$ and $q$in$_{Fp}^{x}$, find a number k such that $r=qk(modp)$.

Similarly the Elliptic Curve Discrete Logarithm Problem is: Given points P and Q in an ECG over a finite field find an integer k such that $Pk=Q$. Here k is called the discrete log of Q to the base P.

This doesn't seem like a difficult problem, but if you don't know what k is calculating $Pk=Q$ takes roughly $2^{k/2}$ operations. So if k is say, 160 bits long, then it would take about 2^{80}operations!! To put this into perspective, if you could do a billion operations per second, this would take about 38 million years. This is a huge savings over the standard public key encryption system where 1024 and 3074 bit keys are recommended. The smaller size of the keys for Elliptic Curve Encryption makes it ideal for applications such as encrypting cell-phone calls, credit card transactions, and other applications where memory and speed are an issue. There are pros and cons to both ECC and RSA encryption. ECC is faster than RSA for signing and decryption, but slower than RSA for signature verification and encryption. Much of the material used in this paper can be found in the websites listed in the references.

II. OBJECTIVE OF THE PAPER

The objective of this paper is to design and implement an efficientElliptic Curve Cryptography algorithm for security in data transfer usingVerilog HDL. The algorithm is simulated, synthesized and implemented on FPGA.HDL based algorithm occupies less memory space forexecution and the execution speed of HDL is very high when compared tothe algorithm implemented using highlevel language. The VerilogHDLlanguage based for hardware implementation of the ECC provides betterreliability, than software based security system. The reconfiguration andrecustomisation becomes easier at any stage.

III. PROPOSED WORK

The growing possibility of modern communications needs specialmeans of security especially on computer network. The network security isbecoming more important as the number of data being exchanged on theinternet increases. Therefore, the confidentiality and data integrity arerequired to protect against unauthorized access. This resulted in an explosivegrowth in the field of cryptography. All the most popular forms of security protection for informationhiding rely on cryptographic techniques which encrypt the private data so thatit cannot be read by anyone except the intended recipient who has the rightkey to convert the encrypted message back to its original form. In this paper information security is done by ImplementingElliptic curve cryptography in Verilog HDL. The implementation in VerilogHDL consists of three main modules namely,

1. Main Controller
2. Multiplier and
3. Adder

The program written in Verilog HDL run by Xilinx ISE toolwhen synthesized gives the hardware module of the program up to the gatelevel and then given for implementation using FPGA which does the chiplayout. Before implementation the program is verified by using simulationwhich gives the output as signal waveformand even the testbench program can be written and verified using simulationsoftware before synthesis using Xilinx ISE.

IV. MAIN MODULES OF ELLIPTIC CURVE CRYPTOGRAPHY :

The basic unit of description in Verilog is the module. A moduledescribes the structure or functionality of a design and also describes the portsthrough which it communicates externally with the other modules.The structure of a design is described using switch-level primitives,gate-level primitives and user defined primitives. Dataflow behaviour of adesign is described using continuous assignments and sequential behaviour isdescribed using procedural constructs.

ISSN 2348 – 8034
Impact Factor- 4.022

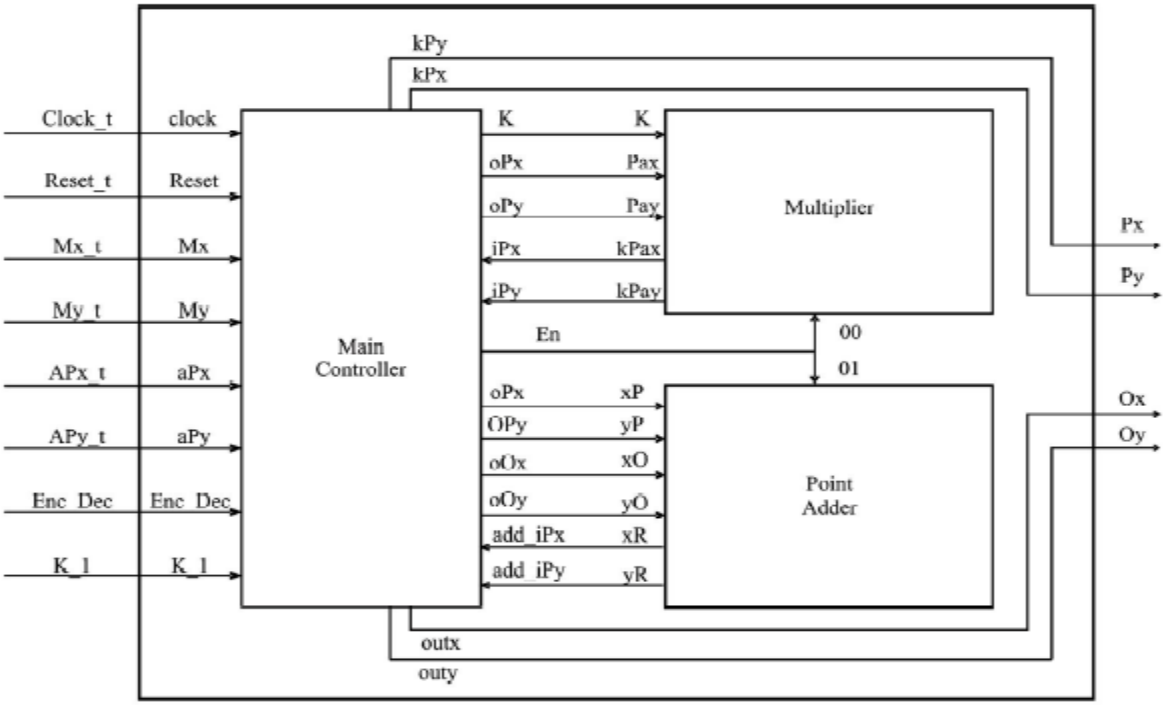


Figure 5 : Main Modules in Elliptical Curve Cryptography

Figure 5 shows the above three modules. The main controllercontrols the functioning of the adder and multiplier components. Themultiplier block is selected when the Enable line is '00'. The multiplierperforms multiplication of an integer with a point on the elliptic curve. Themultiplication is done by successive addition.The adder block is selected when the Enable line is '01'. The adderperforms addition of two points on an elliptic curve.

Figure 6 shows Block Diagram of Main Controller**,** the main controller controls the functioning of the adder andmultiplier components. It has several internal signals, namely **Clock**: The internal clock; **Reset**: The reset signal is used to bring back all the components totheir initial conditions, when set to '1'.**Mx** : X -coordinate of the message to be transmitted ; **My** : Y - coordinate of the message to be transmitted; **aPx** : X - coordinate of the quantity "**xP**", (required in the encrypterpart);**aPy** : Y - coordinate of the quantity "**yP**", (required in the encrypterpart); **Enc_Dec** : Selects encryption/decryption;'0' – Encryption;'1' – Decryption;k_l : A very large integer (Private key) generated at random;**En** : Enables Multiplier/point Adder;'00' – Multiplier;'01' – Point Adder.

To Multiplier: **k**: A very large integer generated at random which is multipliedwith the point; **oPx** : X - coordinate of the point which is to be multiplied with the integer; **oPy** : Y - coordinate of the point which is to be multiplied with theinteger.

To Point Adder:**oPx**: X - coordinate of the addend; **oPy**: Y - coordinate of the addend; **oQx** : X - coordinate of the augend; **oQy** : Y - coordinate of the augend.

From Multiplier:**iPx** : X - coordinate of the result; **iPy** : Y - coordinate of the result;
From Point Adder**: *add_iPx***: X - coordinate of the result;***add_iPy***: Y - coordinate of the result;
Final Outputs:kPx: X - coordinate of the product "**xP**" (Encryption); ***kPy***: Y - coordinate of the product "**yP**" (Encryption); ***outx:*** X - coordinate of the expression "**xyP+M**"; ***outy:*** Y - coordinate of the expression "**xyP+M**".

[Potharaju, 4(11): November 2017] ISSN 2348 – 8034
DOI- 10.5281/zenodo.1067986 Impact Factor- 4.022

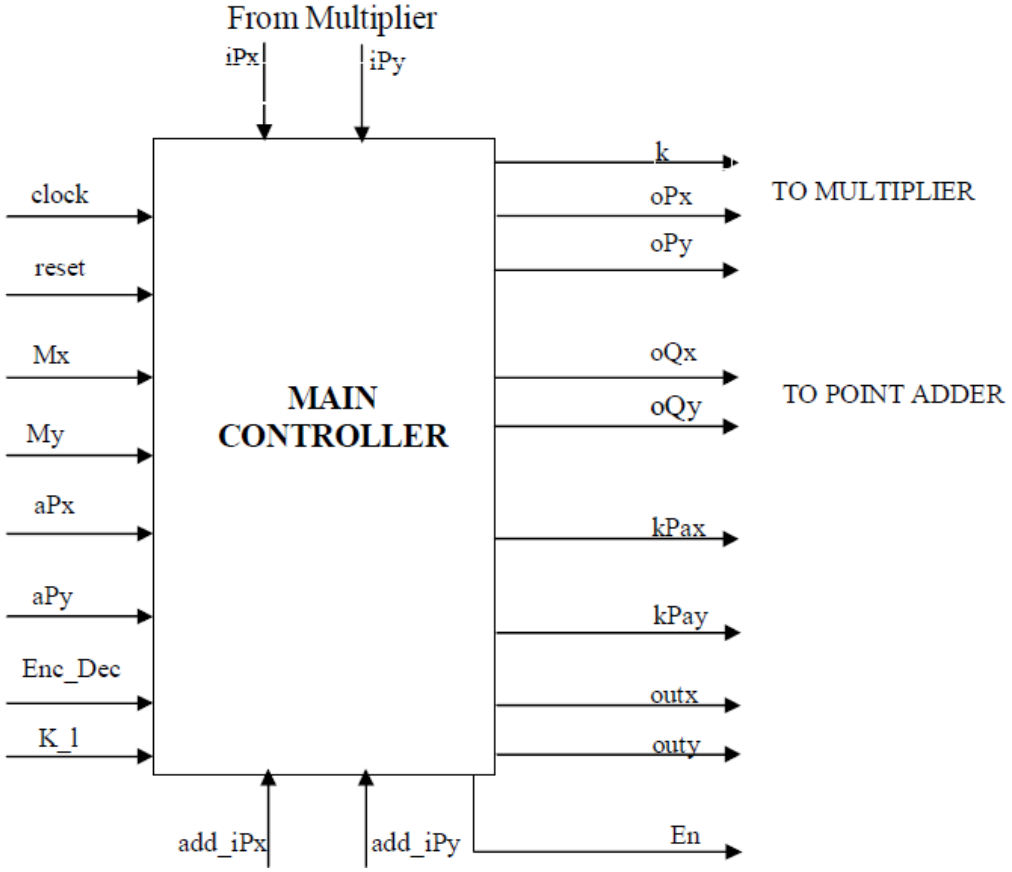


Figure 6: Block Diagram of Main Controller

The **Figure 7** shows the block diagram of multiplier. Themultiplier block is selected when the *En*line is '00'. The multiplier performsmultiplication of an integer with a point on the elliptic curve. Themultiplication is done by successive addition. This addition follows the rulesof Elliptic Curve Arithmetic, known as *Point Doubling*.

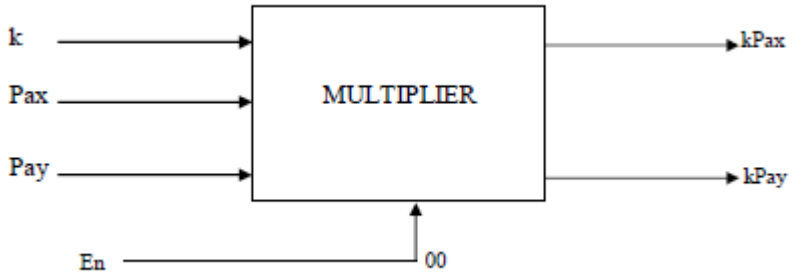


Figure 7: Block Diagram of Multiplier

From Main Controller: k: it is a randomly generated integer which is multipliedwith a point on the curve;***Pax***: x-coordinate of a point on the elliptic curve;***Pay***: y-coordinate of a point on the elliptic curve;***En***: enables the multiplier block;

To Main Controller: kPax: x-coordinate of the resultant; ***kPay***: y-coordinate of the resultant.

ISSN 2348 – 8034
Impact Factor- 4.022

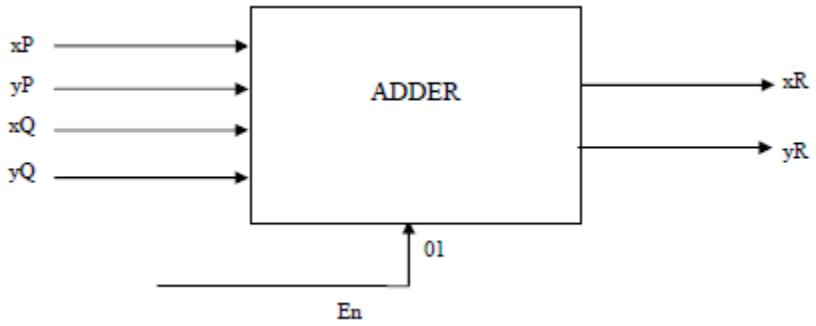


Figure 8: Block Diagram of Adder

Figure 8 shows the block diagram of adder module. The adderblock is selected when the *En*line is '01'. The adder performs addition of twopoints on an elliptic curve. Addition is based on the rules of Elliptic CurveArithmetic known as point addition. Similar to multiplication, addition isperformed on projective co-ordinates and the formulae are given in the abovetable. The result is in affine.

From Main Controller :xP: x-coordinate of a point on the elliptic curve; ***yP***: y-coordinate of a point on the elliptic curve; ***xQ***: x-coordinate .of a point on the elliptic curve; ***yQ***: y-coordinate of a point on the elliptic curve; ***En***: enables the adder block

To Main Controller :xR: x-coordinate of the resultant; ***yR***: y-coordinate of the resultant

V. RESULT & DISCUSSION

Simulation Of Encryption And Decryption

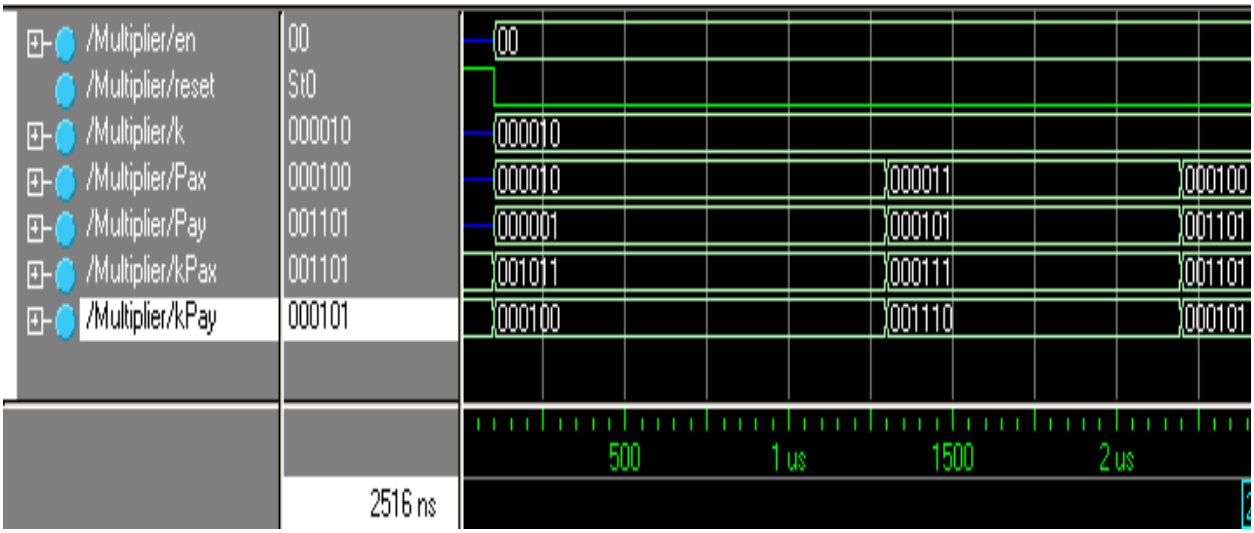
In order to implement ECC cryptographic system three modulesnamely (i) main controller (ii) point adder and (iii) multiplier are to besimulated using Xilinx ISE simulator. The program for the abovethree modules are written inVerilog HDL.There are three ways to simulate a design, software simulation, hardware acceleration and hardware emulation. Software simulation is used here, which is used to run Verilog HDL based design. They load the Verilog HDL code and simulate the behaviour in software.

When the encryption_ decryption is low (0), then the encryption partis selected. The simulation waveforms of multiplier block, adder block andmain controller are obtained. The simulation waveform for multiplier block isgiven below. When the enable line is '00' the multiplier block is selected.Then the clock is set as high and the reset as low, the system will multiply theconstant K with the combination of public and private key and gives theproduct as KPax and KPay. The resulting output for encryption anddecryption of multiplier are shown in the screen shots below in

ISSN 2348 – 8034
Impact Factor- 4.022

Figure 9.

Encryption/Multiplier:



Decryption/Multiplier :

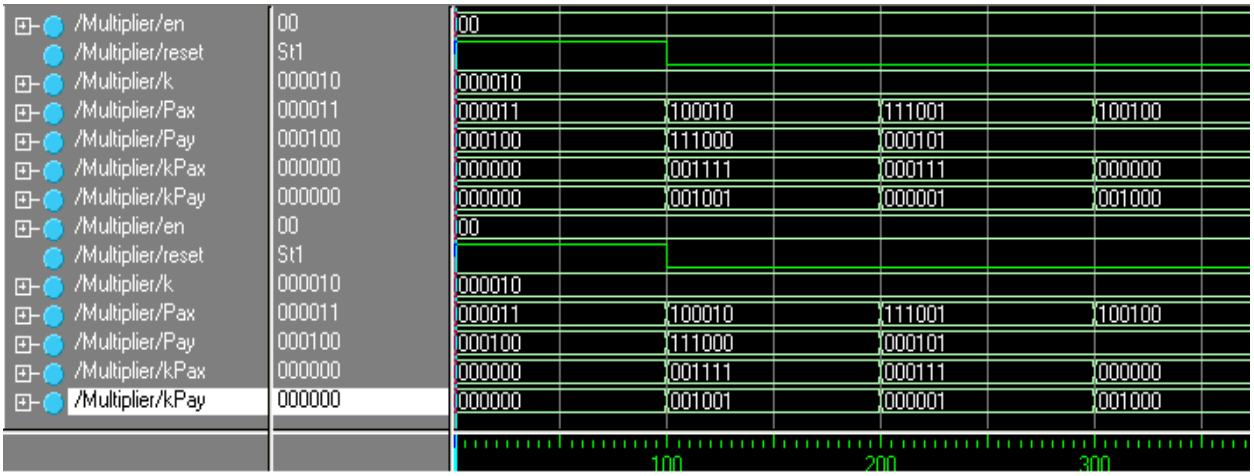
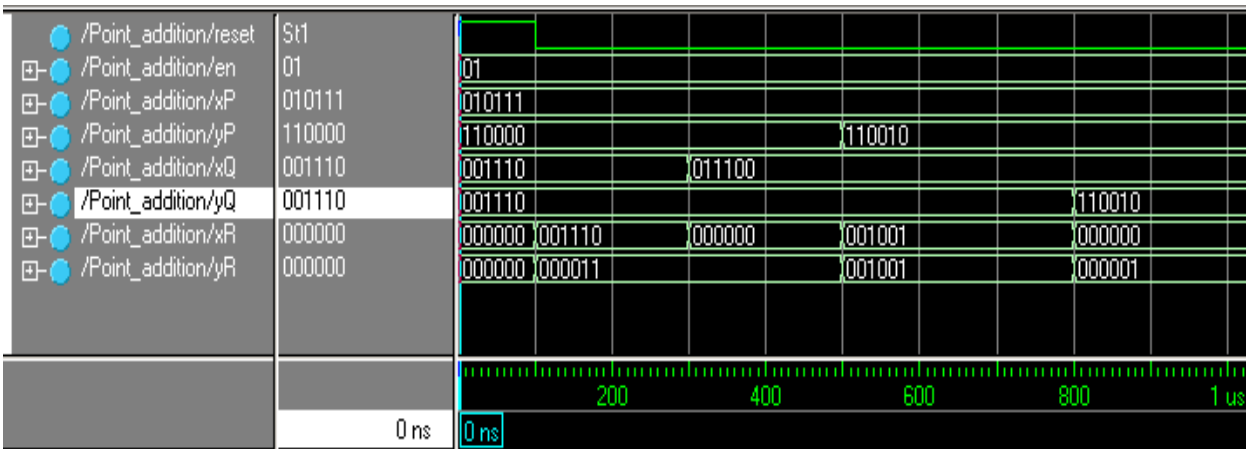


Fig.9:Screen Shots for Multiplication Operations used inEncryption and Decryption

When the enable line is '01' the adder block is selected. Then theclock is set as high and the reset as low. The adder performs the addition oftwo points on the elliptic curve. The message coordinates are added with thepublic key coordinates and the result is given to the main controller. The inputsare Mx, My, aPx and aPy and the result is Mx+aPx and My+aPy. The outputscreen shots obtained for the above operations are in **Figure 10**.

ISSN 2348 – 8034
Impact Factor- 4.022

Encryption/Adder:



Decryption/Adder:

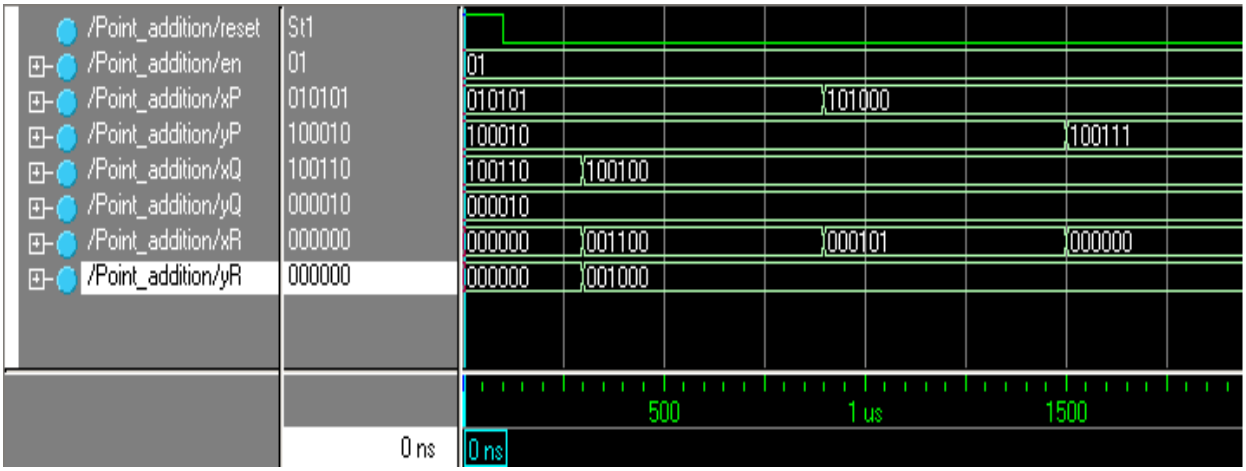


Fig.10: Output Screen Shots for Encryption and Decryption Addition Operations

The main controller controls the function of adder and multiplier. Ithas several internal signals for controlling the operation of the whole system.The clock is set as high and the reset is low. It has the input as the messagecoordinates and the public and private keys. The encrypted message coordinate is obtained as an output in the encryption process and the original message isobtained back in the decryption process. The screen shots corresponding tothe main controller's output are shown in **Figure 11&12**.

[Potharaju, 4(11): November 2017]
DOI- 10.5281/zenodo.1067986
ISSN 2348 – 8034
Impact Factor- 4.022

Encryption/Main controller:

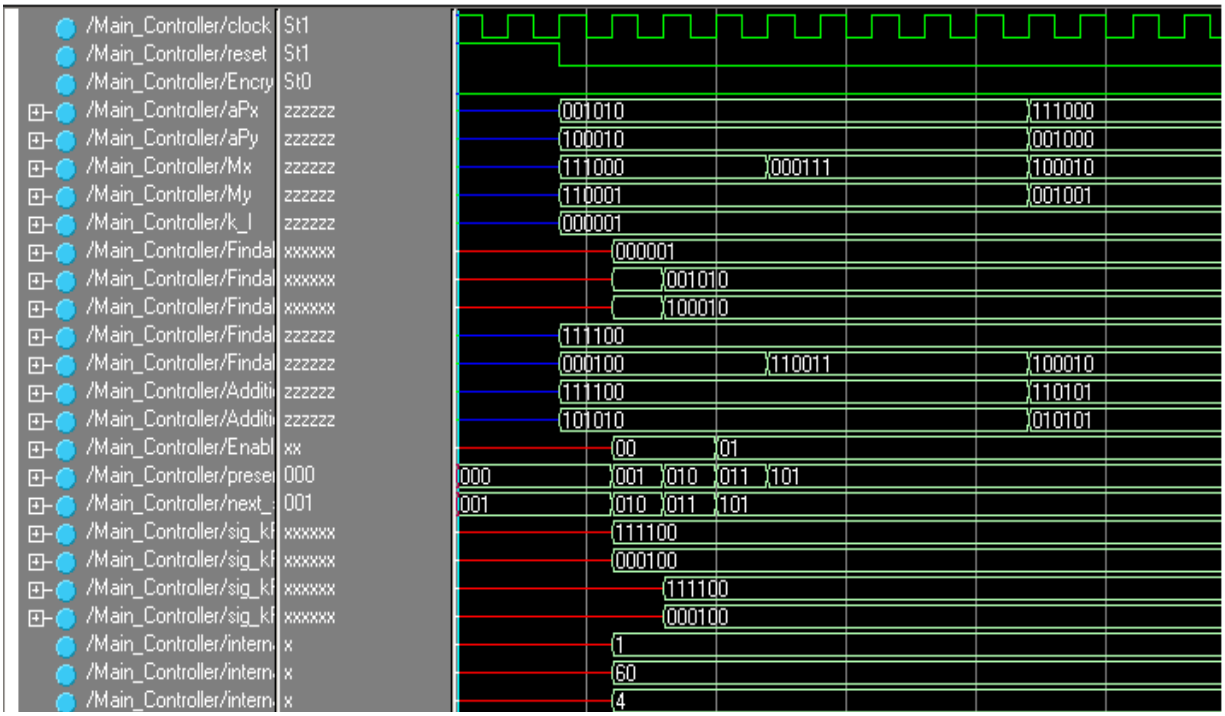


Fig.11: Main controller outputs for Encryption

Decryption/Main controller:

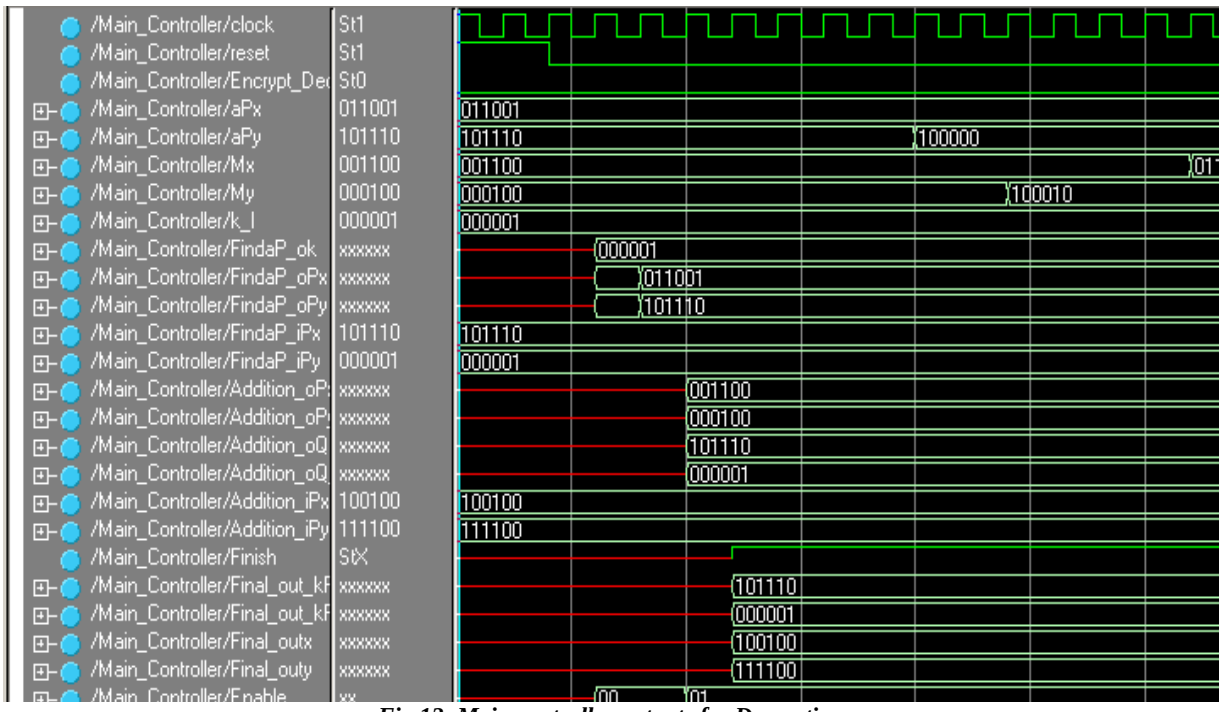


Fig.12: Main controller outputs for Decryption

[Potharaju, 4(11): November 2017] ISSN 2348 – 8034
DOI- 10.5281/zenodo.1067986 Impact Factor- 4.022

Synthesis Result

The architecture has been synthesized by using Xilinx ISE. ThePlace and Route report from ISE shows the number of logic gatesconsumed. Table 1 shows the device utilization summary of encoder anddecoder and Table 2 gives the timing summary

Table 1 Device Utilization Summary

Sl. No.	Name	Utilization
1	Multiplexer	6
2	Adder	9
3	Subtractor	9
4	Multiplier	58
5	Comparator	6
6	Latches	22
7	LUT's	43
8	IOB's	124
9	GCLK's	2
10	Flip flops	118
11	Registers	6

Table 2 Timing Summary

Sl. No.	Name	Utilization
1	Minimum period	5.043 ns
2	Maximun frequency	198.295 MHz
3	Total memory usage	269824 Kilobytes

VI. CONCLUSION

The Elliptic Curve Cryptography implemented in Verilog HDL provides a highly efficient encryption algorithm compared to existing algorithms such as RSA and DSA. Further It is achieved an equivalent strength with a key of lesser number of bits (150 – 200 bits). However, selecting a suitable curve and finite field are crucial for ECC security. In this work a method for implementing elliptic curve cryptography using Verilog HDL has been provided. Since the proposed hardware implementation of ECC algorithm has shorter key length, shorter parameters and shorter calculation time it proves to be efficient than other cryptographic algorithms. The simulation results provide further proof of the efficiency of the algorithm. The synthesis report gives the details upto the gate level for the logic blocks along with timing summary. The Place and route mapping for the system is achieved. Hardware architectures are much more dependent on the varying bit size than a software implementation.

VII. ACKNOWLEDGEMENTS

I sincerely thank FIST (Fund for Improvement of Science and Technology Infrastructure) Sponsored by Department of Science and Technology (DST)at Vignana Bharathi Institute of Technology, Hyderabad.Underwhich the present work was possible

REFERENCES

1. *M. B. I. reaz, J. jalil, H. husian, F. H. hashim,FPGA Implementation of Elliptic Curve Cryptography Engine for Personal Communication Systems.*
2. *Shishay Welay Gebregiyorgis,Algorithms for the Elliptic Curve Discrete Logarithm and theApproximate Common Divisor Problem.*
3. *Christophe Petit, Michiel Kosters, and Ange Messeng,Algebraic approaches for the Elliptic CurveDiscrete Logarithm Problem over prime fields.*
4. *W. Diffie and M. E. Hellman, "New Directions in Cryptography," IEEE Transactions onInformation Theory, vol. 22, pp. 644–654, 1976.*
5. *Lyndon V. Judge,Design Methods for Cryptanalysis.*
6. *R. Rivest, A. Shamir, and L. M. Adleman, "A Method for Obtaining Digital Signatures andPublic Key Cryptosystems," Communications of the ACM, vol. 21, pp. 120–126, 1978.*
7. *A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography. The CRC Press Series on Discrete Mathemtaics and its Applications, CRC Press, first ed., 1996.*
8. *T. Elgamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Transactions on Information Theory, vol. 31, pp. 469–472, July 1985.*
9. *N. Koblitz, "Elliptic Curve Cryptosystems," Mathematics of Computation, vol. 48, pp. 203–209, January 1987.*
10. *D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm(ECDSA)," IJIS, 2001.*
11. *W. Stallings, Cryptography and Network Security. Prentice Hall, 1999.*
12. *J. O'Connor and E. Robertson, "The MacTutor History of Mathematics Archive."http://www-gap.dcs.st-and.ac.uk/ history/Mathematicians/Khayyam.html, July 1999.*
13. *V. S. Miller, "Use of Elliptic Curves in Cryptography," in LNCS 218, Advances in Cryptology - CRYPT0 '85 (H. Williams, ed.), (Berlin Heidelberg), pp. 417–426, Exploratory Computer Science, IBM Research, P.O. Box 218, Yorktown Heights, NY 10598, Springer-Verlag, 1986.*
14. *J. B. Fraleigh, A First Course in Abstract Algebra. Pearson Education (Singapore) Pte. Ltd., 2003.*
15. *N. Koblitz, Algebraic Aspects of Cryptography. Springer, 1998.*
16. *B.Schneier. Applied Cryptography. John Wiley and Sons, second edition, 1996.*
17. *Cryptography and Elliptic Curves,http://www.tcs.hut.fi/~helger/crypto/link/public/elliptic/*
18. *Julio Lopez and Ricardo Dahab, "An overview of elliptic curve cryptography", May 2000.*
19. *V. Miller, "Uses of elliptic curves in cryptography", Advances in Cryptology - CRYPTO'85, LNCS 218, pp.417-426, 1986.*
20. *Jeffrey L. Vagle, "A Gentle Introduction to Elliptic Curve Cryptography", BBN Technologies*
21. *Mugino Saeki, "Elliptic curve cryptosystems", M.Sc. thesis, School of Computer Science, McGill University, 1996. http://citeseer.nj.nec.com/saeki97elliptic.html*
22. *J. Borst, "Public key cryptosystems using elliptic curves", Master's thesis, Eindhoven University of Technology, Feb. 1997. http://citeseer.nj.nec.com/borst97public.html*
23. *Aleksandar Jurisic and Alfred Menezes, "Elliptic Curves and Cryptography", Dr. Dobb's Journal, April 1997.*
24. *Robert Milson, "Introduction to Public Key Cryptography and Modular Arithmetic"*
25. *Aleksandar Jurisic and Alfred J. Menezes, Elliptic Curves and Cryptography*
26. *William Stallings, Cryptography and Network Security-Principles and Practice second edition, Prentice Hall publications.*
27. *R. Schroppel, H. Orman, S. O'Malley and O. Spatscheck, "Fast key exchange with elliptic key systems", Advances in Cryptography, Proc. Crypto'95, LNCS 963, pp. 43-56, Springer-Verlag, 1995.*
28. *M. Rosing, Implementing Elliptic Curve Cryptography. Manning Publications, 1999.*