International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

Verilog Based Implementation of Efficient Elliptic Curve Processor Galois fields.

R CHARITHA [1], K. PRASAD BABU [2]
[1] *192T1D3806 M.Tech DECS, Dr.KVSRECW, JNTUA,Affiliated, Kurnool, Andhra Pradesh India*
[2]*Associatet Professor, Dept of ECE, Dr.KVSRECW, JNTUA,Affiliated, Kurnool, Andhra Pradesh India*
[1]charithacherry4444@gmail.com
[2]kprasadbabuece433@gmail.com

***Abstract*— ECC is an asymmetric cryptographic system that provides an equivalent security to the well-known Rivest, Shamir and Adleman system. The basic operation in ECC is scalar point multiplication, where a point on the curve is multiplied by a scalar. A scalar point multiplication is performed by calculating series of point additions and point doublings. On the other hand GF operations consist of three operations addition, Multiplication and Inversion. In this project implementation of the elliptic curve cryptography processor is proposed. Galois fields play an important role in cryptography. As a result of their carry free arithmetic property, they are suitable to be used in hardware implementation in ECC. Here the multiplier is implemented using a double and add algorithm, to obtain an efficient elliptic curve processor over Galois fields.**

Keywords: **Cryptography, Galois fields, RSA, ECC.**

I INTRODUCTION

Elliptic curves over a field *K* are defined by the reduced Weierstrass equation in equation when the characteristic of the field is two or three. The set of solutions along with a point at infinity *O* defines the algebraic structure as a group with point addition as the basic operation

$$E : y^2 = x^3 + ax + b.$$

The control unit consists of the main controller of the processor that is an FSM. It also includes two processing units that control the procedures for the point doubling and point addition. In order to process the different coordinate systems, the other processing units can be generated to support them as an add-on feature. Hence, new instructions need to be added to the instruction set to accommodate the new processing units. Different scalar point multiplication algorithms are supported at instruction level. Different projective coordinate systems and point addition/doubling variations are configured at the control level through add-on processing units. On the other hand, elliptic curves with different finite fields are supported as a pre-synthesis process.

Cryptography is the science of information security. Cryptography includes techniques such as microdots, merging words with images and other ways to hide information in storage or transit. Cryptography is most often associated with scrambling plaintext (ordinary text, sometimes referred to as clear text) into cipher text (a process called encryption), then back again (known as decryption).A Cryptographic system that uses two keys – a public key known to everyone and a private or secret key known only to the recipient of the message. Individuals who practice this field are known as cryptographers.

II PROBLEM DEFINITION

The AU is the core unit of the processor that includes the following blocks: 1) modular addition/subtraction block; 2) modular multiplication block; and 3) modular division block. Consider the equation, Q = kP, where Q, P are points in the elliptic curve E(a,b) and k< P. It is relatively easy to calculate Q given k and P , but it is relatively hard to determine k given Q and P. This is called discrete logarithmic problem for elliptic curves. The prime number p sets the upper limits of the equation and is used for modulus arithmetic. P and Q are the points on the elliptic curve. When using ECC, there are two types of arithmetic, the cartesian coordinates for resolving the elliptic curve and modular arithmetic used for resolving the the points along the coordinate system k is a very large integer generated at random which is multiplied with the point..

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

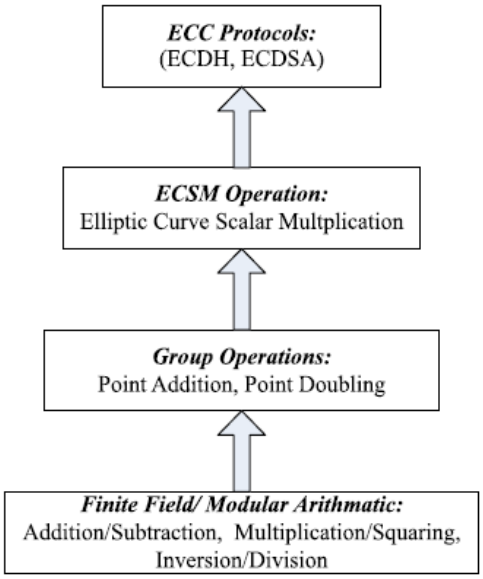

Figure1: Existing System.

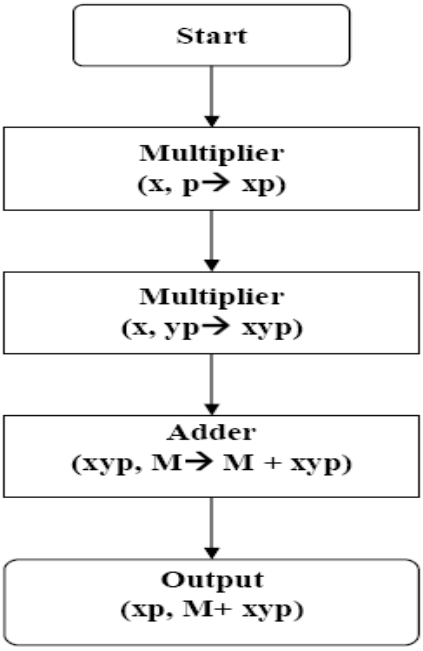

Figure2: Proposed System Encryption.
Let x, y be the private keys used by the transmitter and receiver respectively. The transmitter secret key x is multiplied with the public value of the receiver yP i.e., xyP. The message is encrypted using the formula M + xyp, where M is the plain text.

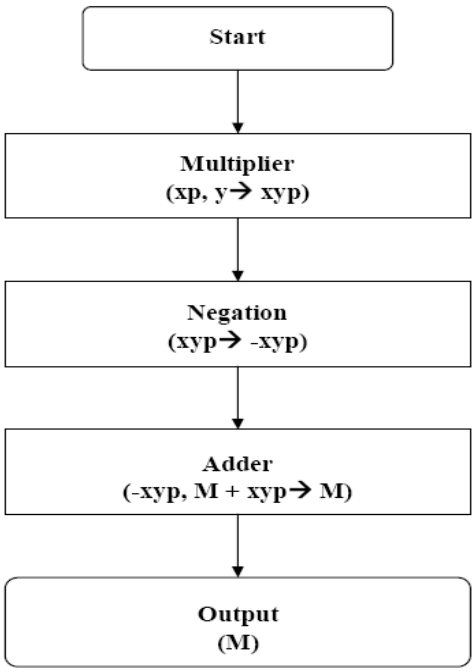

Figure3: Proposed System Encryption and Decryption.
The receiver's secret key y is multiplied with the public value of the transmitter xP i.e., yxP. The message is decrypted by subtracting the value yxP from the received message i.e., M + xyP – xyP = M.

III IMPLEMENTATION

The elements of a finite field can be represented in several different ways. For any prime power there is a single finite field, hence all representations of Galois Field, GF (2^8) are isomorphic. Despite this equivalence, the representation has an impact on the implementation complexity. Joan Daemen and Vincent Rijmen have chosen for the classical polynomial representation.A byte b, consisting of bits b7 b6 b5 b4 b3 b2 b1 b0, is considered as a polynomial with coefficient in {0,1}:

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$$

The addition of two finite field elements is achieved by adding the coefficients for corresponding powers of their polynomial representations, this addition being performed in GF (2^8), that is, modulo 2, so that 1 + 1 = 0.
Consequently, addition and subtraction are both equivalent to an exclusive-or (**XOR**) operation on the bytes that represent field elements. Addition operations for finite field elements will be denoted by the symbol ⊕.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

Finite field multiplication is more difficult than addition and is achieved by multiplying the polynomials for the two elements concerned and collecting like powers of x in the result. Since each polynomial can have powers of x up to 7, the result can have powers of x up to 14 and will no longer fit within a single byte.

IV Modules Implementation

Implementation consists of three main modules design as stated below.

i) Main Controller

ii) Multiplier and

iii) Adder

The main controller controls the functioning of the adder and multiplier components. The
multiplier block is selected when the Enable line is '00'.
The multiplier performs multiplication of an integer with a point on the elliptic curve. The multiplication is done by successive addition.
The adder block is selected when the Enable line is '01'. The adder performs addition of two points on an elliptic curve. Addition is based on the rules of Elliptic Curve Arithmetic known as point addition.

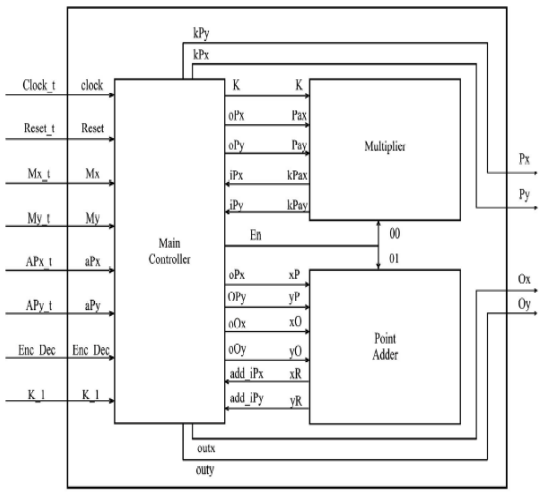

Figure 4 Main Modules in Elliptical Curve Cryptography

The main controller controls the functioning of the adder and multiplier components. It has several internal signals, the functions of which are mentioned below.
Clock : The internal clock Reset : The reset signal is used to bring back all the components to their 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 encrypter part)
aPy : Y - coordinate of the quantity "yP", (required in the encrypter part)
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 Multipler
k : A very large integer generated at random which is multiplied with 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 the integer.
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 Multipler
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"

V RESULTS

Below figures represents the individual RTL view of each module and the simulation of three main modules.

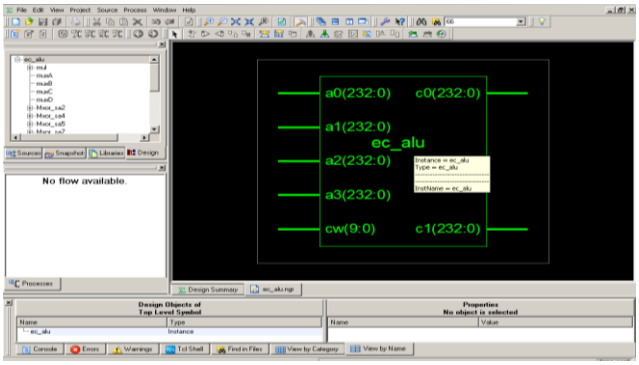

Figure 5 RTL view of EC ALU.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

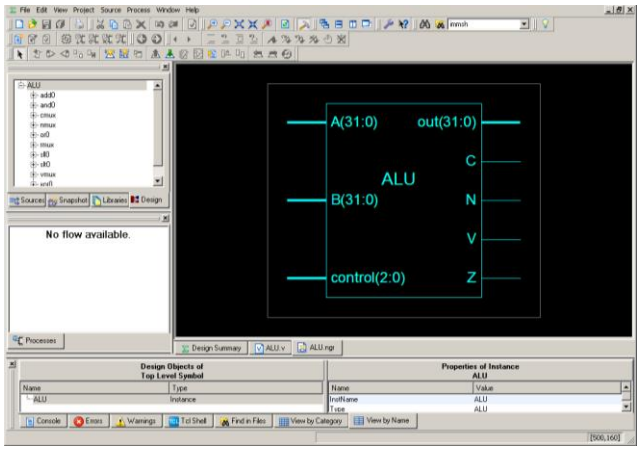

Figure 6 RTL view of ALU.

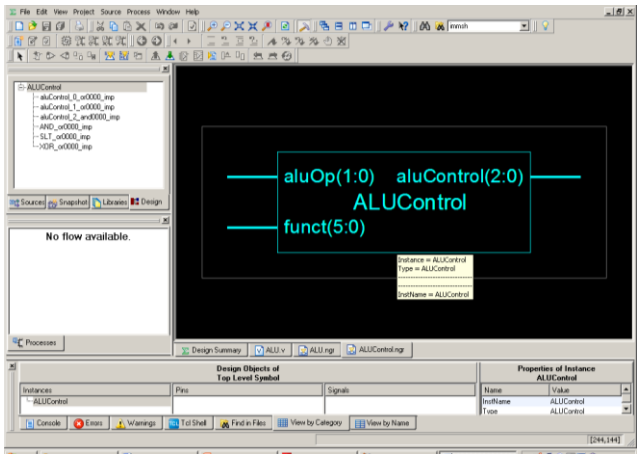

Figure 7 RTL view of ALUcontrol

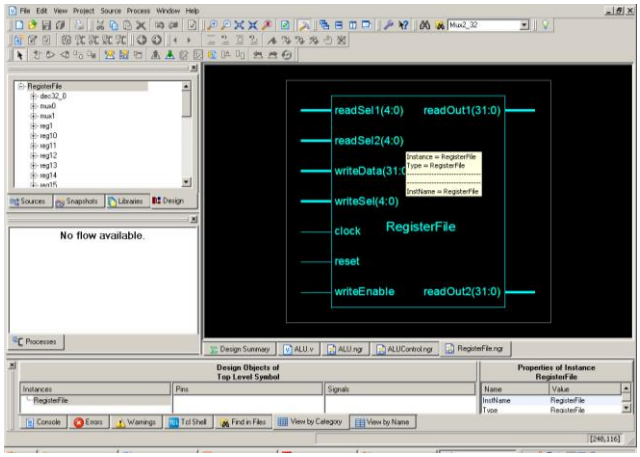

Figure 8 RTL view of REGISTER FILE.

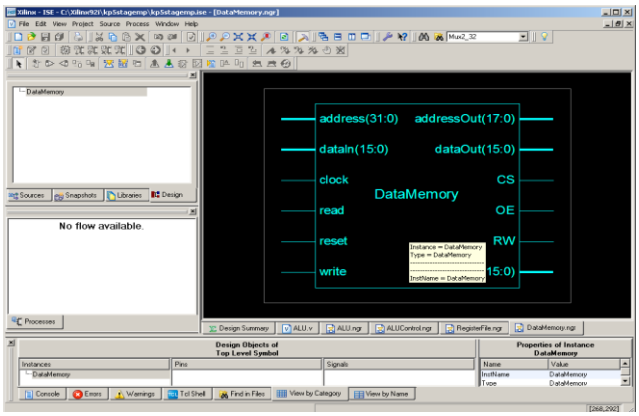

Figure 9 RTL view of DATA MEMORY

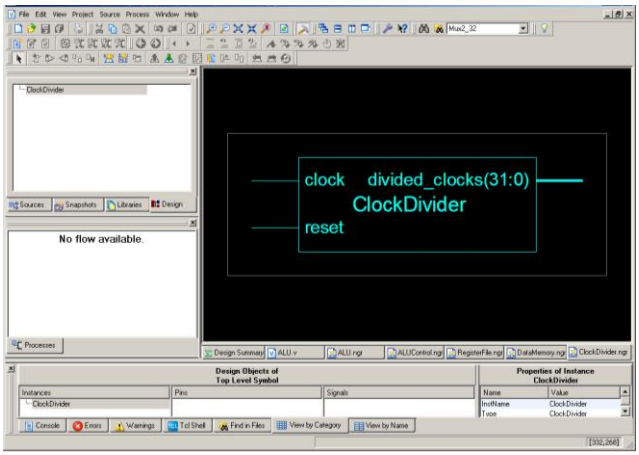

Figure 10 RTL view of clock divider circuit

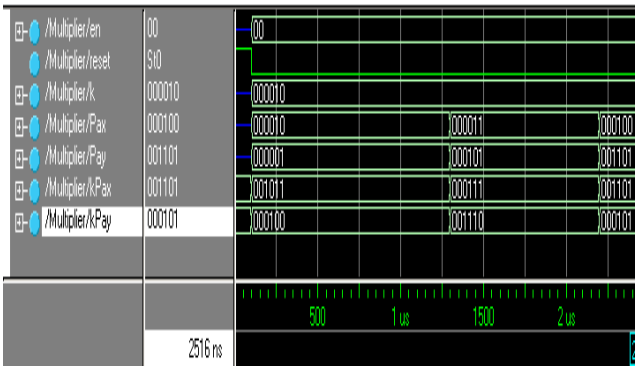

Figure 11 Multiplication Operations used in Encryption

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

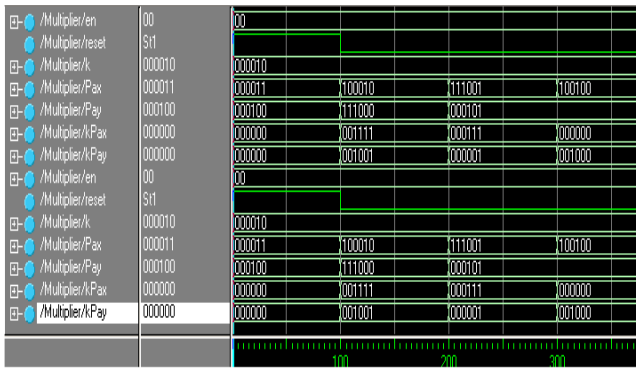

Figure 12 Multiplication Operations used in Decryption

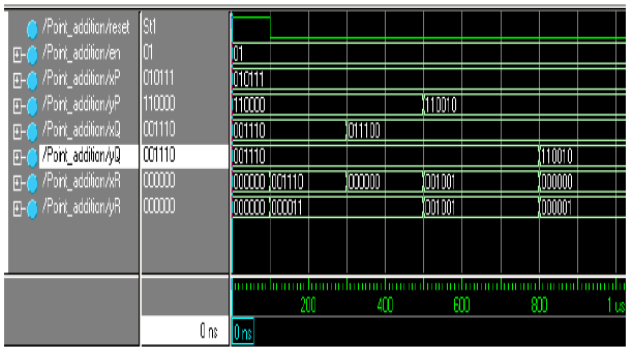

Figure 13 Addition Operations used in Encryption

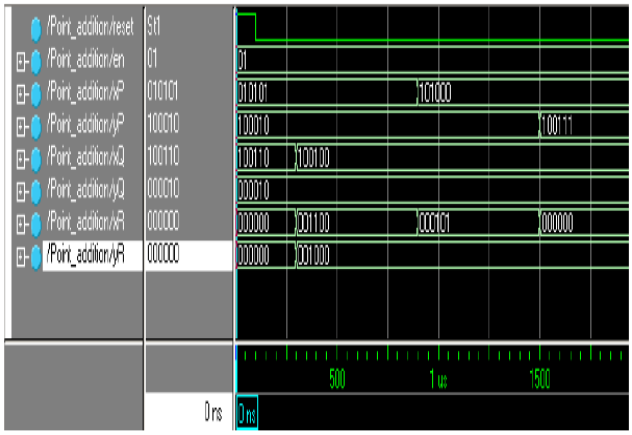

Figure 14 Addition Operations used in Decryption

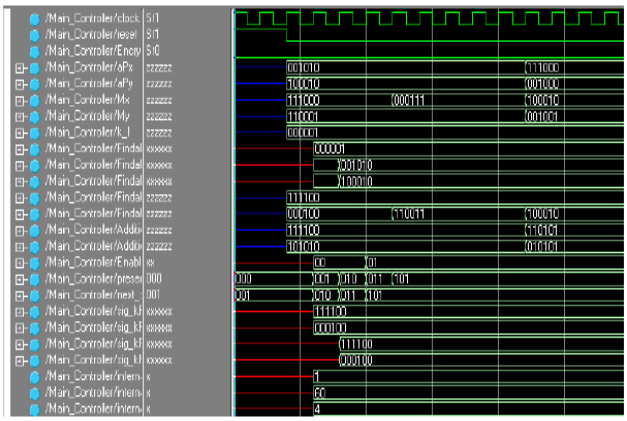

Figure 15 Main Controller outputs in Encryption

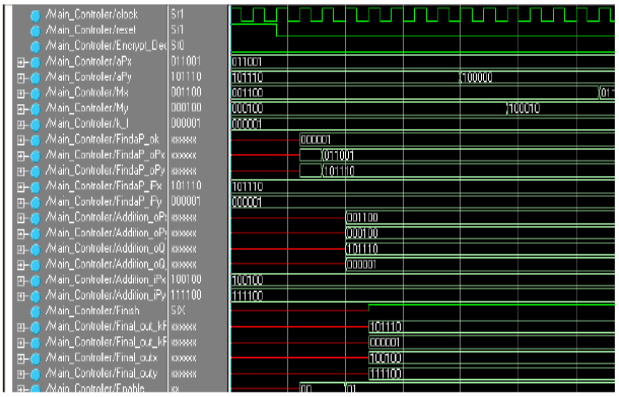

Figure 16 Main Controller outputs in Decryption.

Table 1: Existing System Results.

operations	Field Size	Area	Freq (M.Hz)
Addition	224	169	174.9
Subtraction	256	193	162.4
Multiplication	224	357	132.8
Inversion	224	1382	147.1

Table 2: Proposed System.

Operations	Field Size	Area	Freq (M.Hz)
Addition	233	154	168.6
Subtraction	233	186	157.8
Multiplication	233	321	127.3
Inversion	233	1296	138.9

Website: ijetms.in Issue: 6 Volume No.5 November – 2021 DOI: 10.46647/ijetms.2021.v05i06.005
ISSN: 2581-4621

Table 3: Existing Implementation of point operations over *GF*(*p*) using Kintex-7 FPGA

Operations size	Field	Area (slices)
point doubling (PDBL)	224	2374
point addition (PADD)	224	2260

Table 4: Proposed Implementation of point operations over *GF*(*p*) using Spartan2E FPGA

Operations size	Field	Area (slices)
point doubling (PDBL)	417	1536
point addition (PADD)	238	768

VI CONCLUSIONS

The achieved short critical path is due to the improved pipelining strategies used in Karatsuba multiplier and the efficient architecture of the divider. It can be noted that the modular multiplier is the largest block within the design due to the three recursively built Karatsuba blocks, which operate in parallel. Our modular divider performs the fastest timing of prime field dividers and competitive to binary field $GF2^{233}$ modular divider. Elliptic Curve Cryptosystems offer security comparable to that of traditional asymmetric crypto systems, such as those based on the RSA algorithm and Digital signature algorithm with smaller keys and computationally more efficient algorithms. The ability to use smaller keys and computationally more efficient algorithms than traditional asymmetric cryptographic algorithms are two main reasons for using Elliptic Curve Cryptography.

References

1) Md. Mainul Islam, "Design and Implementation of High-Performance ECC Processor with Unified Point Addition on Twisted Edwards Curve" MDPI Sensors 2020,20,5148, pp 1-19.
2) John P. Uyemura, "Introduction to VLSI circuits and systems".
3) H. Fan and M. A. Hasan, "A survey of some recent bit-parallel GF(2n) multipliers," Finite Fields Appl., vol. 32, pp. 5–43, Mar. 2015.
4) M. A. Hasan, A. H. Namin, and C. Negre, "Toeplitz matrix approach for binary field multiplication using quadrinomials," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 3, pp. 449–458, Mar. 2012.
5) B. Rashidi, R. R. Farashahi, and S. M. Sayedi, "High-speed and pipelined finite field bit-parallel multiplier over GF(2m) for elliptic curve cryptosystems," in Proc. 11th Int. ISC Conf. Inf. Secur. Cryptol. (ISCISC), Sep. 2014, pp. 15–20.
6) C. Rebeiro, S. Roy, and D. Mukhopadhyay, "Pushing the limits of highspeed GF(2m) elliptic curve scalar multiplication on FPGAs," in CHES, vol. 7428. Berlin, Germany: Springer, 2012, pp. 494–511.
7) S. Liu, L. Ju, X. Cai, Z. Jia, and Z. Zhang, "High performance FPGA implementation of elliptic curve cryptography over binary fields," in Proc. 13th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom), Sep. 2014, pp. 148–155.
8) A. P. Fournaris, J. Zafeirakis, and O. Koufopavlou, "Designing and evaluating high speed elliptic curve point multipliers," in Proc. 17th Euromicro Conf. Digit. Syst. Design (DSD), Aug. 2014, pp. 169–174.
9) Z.-U.-A. Khan and M. Benaissa, "Throughput/area-efficient ECC processor using Montgomery point multiplication on FPGA," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 62, no. 11, pp. 1078–1082, Nov. 2015.
10) Z. U. A. Khan and M. Benaissa, "High speed ECC implementation on FPGA over GF(2m)," in Proc. 25th Int. Conf. Field-Program. Logic Appl. (FPL), Sep. 2015, pp. 1–6.