Research Article

International Journal of Advanced Technology and Engineering Exploration, Vol 9(97)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2021.875995

High speed elliptic curve cryptography architecture for NIST recommended Galois field

Kirit Patel[1*], Mihir Shah[2] and Pankaj Prajapati[1]
Department of Electronics and Communication, L.D. College of Engineering, Gujarat Technological University, Ahmedabad, Gujarat, India[1]
Department of Electronics and Communication, Institute of Technology, Nirma University Gujarat, India[2]

Received: 19-May-2022; Revised: 20-December-2022; Accepted: 23-December-2022

Abstract

There is an explosive growth of confidential data transmission over the internet with resource-constrained applications. Real time applications require a high-speed and area-efficient cryptography system with higher security. Elliptic curve cryptography (ECC) offers a widely acknowledged secure environment for the exchange of confidential data in resource-constrained embedded system applications such as network servers, internet banking, smart card-based transactions, defence services, medical services, wireless sensor networks, the internet, etc. ECC provides a superior solution for enhanced security with reduced resource utilization, and it is currently the globally accepted method for transferring confidential data. This paper presents the design and implementation of a high-speed and low area-based ECC system on field-programmable gate array (FPGA). The proposed system is designed for national institute of standards and technology (NIST) recommended prime field. The proposed design has implemented and simulated on the Xilinx Vivado platform using Verilog language. We have proposed restructure point addition and point doubling modules for the Montgomery algorithm to improve the speed. The implementation of the proposed ECC architecture improved the speed of cryptographic operation by an average of 10% to 20% and optimized area by an average of 5% to 10% compared to the previous work. The simulation results confirm that this designed system has state-of-the-art architecture as well as the highest versatility for the ECC design.

Keywords

Public-key cryptography, Field programmable gate array, Scalar multiplication, Elliptic curve scalar multiplication, National institute of standards and Technology (NIST), Galois field (GF).

1.Introduction

Nowadays there is an intense increase in data communications over wireless and wired based networks. Everyday lakhs of transactions happen over the World Wide Web. These transactions have significant data that required confidentiality, validity, and authenticity in data transactions on the open-ended network [1]. Modern security requires the use of cryptographic algorithms because every transaction is associated with a cyberattack. To provide higher security, both asymmetric and symmetric cryptographic algorithms are operated in data transmission for information exchange [2]. Asymmetric algorithms are commonly employed in secure communication to exchange and manage keys, whereas symmetric cryptographic algorithms are utilised for high-throughput secure data exchange [3].

Real time and sensor-based application has limited resources and requires high speed operation. So, all the real time based applications required a cryptographic system that has high speed with utilizing fewer resources [4]. Elliptic curve cryptography (ECC) has different applications in public key cryptography such as mobile security, banking transactions, confidential data management, wireless sensor network, and other security applications [5]. The main challenges of the cryptosystem are low memory, low latency, and low power requirements of resource constrained-based internet-of-things (IoT) application. The design should be competent in terms of speed and area both. Low resource utilization based hardware implementation is the only solution for the justification of challenging applications [6]. ECC is an asymmetric type of cryptosystem and requires a small key size compared to the symmetric type of cryptosystem. Because of its small key size, ECC

[*] Author for correspondence

provides higher performance with equivalent security and less resource requirement in data transmission [7].

The ECC method can be partitioned hierarchically into four process layers based on hardware computation [8].

Finite field arithmetic operations, such as addition, subtraction, squaring, inversion, and multiplication are computed at the first layer. The second layer consists of operations such as point addition (PA) and point doubling (PD). The most computationally important aspect of the ECC is scalar point multiplication (PM), which is executed at the third layer. The encryption and decryption procedures are implemented at the architecture's fourth layer [9].

ECC can be implemented on software and hardware, but ECC based on hardware approaches usually intends to speed up critical fundamental operations with fewer resources. Hardware implementation of ECC is preferred for resource constraint devices such as the IoT, wireless sensor-based devices, and others. The most critical operation in the ECC system is PM. Two types of fields are generally utilized to implement PM, (1) The prime field Galois field (GF (p)) and (2) The binary extension field GF (2m). The national institute of standards and technology (NIST) suggests various key lengths for each of the required fields. Different types of techniques can be implemented on ECC over binary and prime field based on the requirement of the application. These techniques include performance, efficiency improvement, area optimization, and the development of customized cryptosystems.

The most time-consuming and dominant function in ECC is elliptic curve scalar multiplication (ECSM) and it is presented as Q=kP; where Q is another point on the elliptic curve, P is defined as a base point on an elliptic curve, and k is a scalar. The primary goal of the PM is to multiply the private key and fundamental point on the elliptic curve to determine the public key [10].

ECSM is comparatively faster in Jacobian coordinates compared to affine coordinates because it is not using inversion or modular division functions to perform PD and PA. PD and PA use inversion operation and it is the most costly arithmetic function over the finite prime fields [11].

The paper is organised into six sections. The related work was discussed in the section 2. The third section introduced the implementation of the proposed ECC system architecture and each submodule of the ECC system in detail. The fourth section discussed the simulation results and comparisons with the previous research work. The discussion about the article is presented in the fifth section. The conclusion and future work have been discussed in the last section.

2.Literature review

PM is the most complex and familiar operation in ECC, so the majority of reported literature targets improving the operation of PM, and the overall performance of ECC can be improved [12].

Many researchers have designed and implemented their design on hardware for ECSM in prime fields (Fp or GF(p)) or binary fields (F2n or GF(2n)) or twin fields, where ‘n’ and ‘p’ denote the number of bits in respected fields. The ECSM functions can be executed with affine, projective, or mixed coordinate systems. Various authors have utilised the reconfigurable platform as an field programmable gate array (FPGA)-based method to build ECSM because FPGA offers lower design costs, higher flexibility, rapid prototyping, and a shorter design time. The operation of ECSM generates a succession of PA and PD operations [13].

Arunachalam and Perumalsamy [14] proposed a 256-bit reformed interleaved modular multiplication (IMM) algorithm and implemented a hardware architecture on the Virtex-5 and Virtex-7 FPGA platforms. The proposed design is one of the novel architectures of IMM. This architecture provides the most efficient hardware architecture with enhanced performance parameters, including area, frequency, latency, and throughput.

Wen et al. [15] suggested a modular multiplier based on addition with multi-bit scanning. This multiplier allows a maximum length of 576 bits.

Kudithi [13] proposed a unique hardware architecture design for ECSM operations on the prime field, Fp in Jacobian coordinate system. It represents the PD and PA architecture with the implementation using resource sharing technology to attain low hardware resources and high speed. This design is implemented on FPGA and integrated into application specific integrated circuit (ASIC). On the Xilinx Virtex-7 FPGA platform, the proposed ECSM executes in

2.44ms and 1.76ms for 256 and 224 bit prime fields, respectively.

Hu et al. [16] proposed a reconfigurable modular division algorithm and reconfigurable modular multiplication algorithm to decrease power dissipation and increase reconfigurable capability. Implemented modular multiplier and modular divisor on reconfigurable platform.

To reduce the time complexity of normal IMM, a modified radix 2 interleaved approach is proposed. The recommended multiplication algorithm is created and implemented independently on hardware platforms such as the Xilinx Virtex-4, Virtex-5, Virtex-6, and Virtex-7 FPGAs. The proposed modular multiplier supports all the prime fields GF(p) reported in the literature and it is based on prime curves recommended by NIST based on different bit sizes such as 192, 224, 256, 384, and 521. This multiplier executes and offers low memory and low power.

Sajid et al. [10] proposed the reduction of complexity at the instruction level for unified PD and PA operation. The design uses the execution of multiple functions in a single instruction format. Also presented the reduction of hardware requirement by reducing the amount of required memory elements in the design. It reduces the needed clock cycle (CC) number using an integrating 32-bit finite field digit parallel multiplier in the data direction path. They have attained throughput over the area and GF (2233) on XilinxVirtex-7, Virtex-6, and Virtex-5 FPGA platforms. Zhao et al. [17] proposed a point multiplier with a binary field and a reconfigurable secure key with prime field sizes of p-233, p-283, p-409, and p-571.

Zode et al. [18] have proposed data dependency graph of PD and PA to get the optimized area-delay product. Implemented constraint-based scheduling to achieve maximum optimization.

Proposed a compact implementation algorithm of the Montgomery modular multiplier (MMM) on FPGA for embedded devices. The proposed algorithm enhanced the hardware/throughput efficiency of the MMM [19].

Bisheh et al. [20] proposed area-time efficient, lightweight, and high-performance FPGA-based versions of the Curve448 algorithm. The proposed design was implemented using a Xilinx Zynq 7020 FPGA. The proposed architecture, boosts throughput by 12% with the execution of 1,219 PM per second and clock efficiency by 40%. The hybrid Karatsuba multiplier for lightweight embedded applications that utilises less hardware resources. The proposed ECC accelerator with a 2-stage pipeline is about 1, 3 times faster than the variant with no pipeline and exceeds other solutions published in the current literature in terms of FPGA resource consumption and maximum possible clock frequency. Designed ECC processor by merging design with a synthesizable central processing unit to establish a hardware platform for facilitating the future development of ECC fourth layer applications that employ key exchange protocols [21–23].

In elliptic curve encryption, the scalar PM is the essential function that determines how efficiently the system performs in terms of speed, area, and complexity. To reduce the total number of CC involved, the Montgomery method has restructured the PA and PD functions for PM calculation [24–29].

Di et al. [30] proposed a configurable and fast ECC crypto-processor for NIST recommended P-256 and P-521 elliptic curves. They have utilised 7nm and 45nm technology for synthesis. This has been validated on a Xilinx ZCU106 board using NIST-recommended vectors. The provided processor can be utilised to accelerate various ECC-based algorithms. Imran et al. [31] presented an efficient 2-stage pipelined accelerator and implemented over GF (2163) and GF (2571). The accelerator uses a least significant digit-based multiplier with a digit size of 41 bits to execute finite field multiplication in one CC.

Our research focuses on FPGA-based hardware accelerator, elliptical curve type, Galois field, coordination system type, input option, scalar multiplication, group operation, design flexibility, and cryptographic process difficulty. The literature research on hardware accelerators revealed that hardware accelerators can improve system performance in terms of execution time for a variety of engineering applications. PA and PD are restructured for the Montgomery technique to improve its cryptographic speed and the Twisted Edward curve is adopted to speed up the multiplication and square root modules, according to an analysis of various ECC architectures. A survey of real-time applications finds that secure transmission of the secret data is hindered by resource constraints such as limited memory, fast speed, and high performance.

3.Methods

The implementation method presents the ECC architecture submodule implementation followed by ECC system implementation.

3.1ECC architecture submodule implementation

The performance of the ECC system is dependent on modules including modular multiplication, point addition, subtraction, scalar multiplication, and key generation module. Each module is crucial in determining the performance of the ECC system. Each module's design and simulation have been executed with a distinct methodology.

3.1.1Modular multiplication

Modular multiplication is the most basic and important arithmetic operation of ECC system and it is the most time-consuming module over a prime field. The entire ECC module efficiency depends on the modular multiplication design. The higher radix modular multiplier uses less computation time and less CC to process modular multiplication, but it requires more hardware resource modules, so it requires a large area of the system.

To optimize the required area for ECSM, a radix-2 IMM is used in design architecture and implemented in the ECSM modules which require less CC to achieve modular multiplication operations of two n bit integers. The additions of partial product concepts are used to implement an efficient modular multiplication process which is described in algorithm 1 where a is the multiplicand, b is a multiplier, and p is the prime number.

To accomplish the iterative addition operation of the consecutive partial products, an accumulator is doubled at the start of each iteration. Consolidate loop operation for right to left bitwise multiplication is performed using a shift-left register. A momentary variable X of n + 1 bits is utilized to decide the suitable end of the loop, with X(n down to 1) precalculated as the multiplier E and X's represent the least significant bit precomputed as 1. In the case of b0= 0, an additional bit is appended to the LSB to deal with the conclusion of the left shift function. If the most significant bit of X is 1, the multiplicand D is appended to the accumulator in every performed repetition. The prime numbers p and 2p are subtracted from F to execute this modular operation. X (n-1 down to 0) is shifted to a value of zero after n iterations.

Algorithm 1: Modular Multiplication (Radix 2 Interleaved)

```
 Input :  D= $\sum_{i=0}^{n-1} a_i\, 2^i$, E=$\sum_{i=0}^{n-1} b_i\, 2^i$, p=$\sum_{i=0}^{n-1} p_i\, 2^i$
          $a_{i,}\, b_i, p_i \in [0,1]$
Output: F=(D *E) mod p;
           X ← E&”1’;
          F ← 0;
          While  X(n-1 down to 0) ≠ 0 loop
          F ← 2F;
          if  Xn=1 then
          F ← F + D;
          end if;
          F ← F mod p;
          X ← X (n-1 down to 0) & ‘0’ ;
          end loop;
          return F;
```

The concluding modular product of the numbers D and E which is the ultimate modular product of the accumulator is accumulated in the register. The modular multiplication requires a sum of n+1 CC in the proposed architecture. In the design, n represents the CCs for n iterations of the function, and an additional CC is used to accumulate the final result in the register. To conduct modular squaring operations, the inputs of the proposed modular multiplier must be similar (D, E).

3.1.2Point addition and subtraction

Modular subtraction and modular addition are the basic operations of a cryptosystem. Modular addition and subtraction hardware architecture is shown in *Figure 1* and is also represented as algorithm 2.

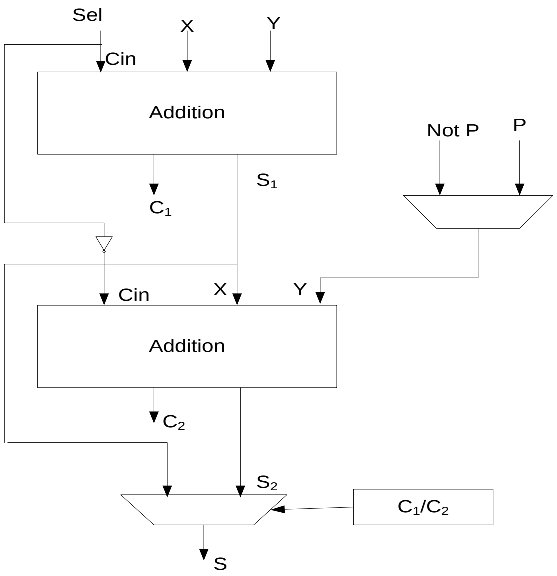


Figure 1 Modular subtraction and addition hardware architecture

Algorithm 2: Modular Addition/ Subtraction

```
Input :  a,b ∈ [0,p-1],p & Sel

Output: S=a ±b( mod p)
1.    If (sel==0) then
2.        $S_1$ = a+b;
3.        $S_2$ = $S_1$ + (~p) +(~Sel);
4.        if ($C_1$ | $C_2$ ) then
5.        S=$S_2$ ;
6.        else S=$S_1$;
7.        end if ;
8.    else
9.        $S_1$ = a+(~b) +Sel;
10.       $S_2$ = $S_1$+p;
11.       if ($C_1$) then
12.       S=$S_2$;
13.       else S=$S_1$;
14.       end if;
15.   end if;
16.   return S;
```

In algorithm 2, steps from 2 to 7 perform modular addition, and steps from 10 to 15 perform modular subtraction. The adder module performs addition between two inputs a and b providing sum S1 and carry C1. The second module executes subtraction between S1 and p with outputs S2 and C2. In the end, S1 and S2 are multiplexed according to step 4 of algorithm 2.

3.1.3Key generation module

A module is shown in *Figure 2* which generates the public key using the projective coordinate. This module is time and area efficient. k is the private key which is used in the decryption of the original message and P(x,y) is a point on the elliptic curve. k and P(x,y) are considered as input that generates a public key which is used in the encryption of the original message.

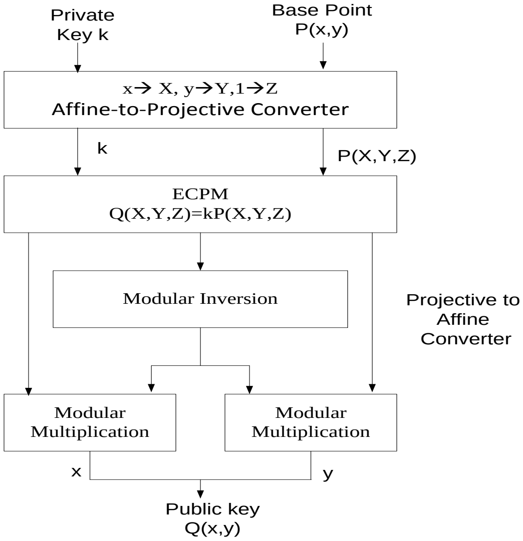


Figure 2 Key generation from private key to public key

Primarily, the affine base point which is defined as P(x, y) is converted to its respective projective from the help of an affine to projective converter module. The public key which is represented as Q (X, Y, Z) is generated by calculating the ECPM of the projective point with the private key k. At the end Q (X, Y, Z) is converted into its respective affine coordinate form Q(x,y) by the projective to affine converter.

3.1.4Elliptic curve point multiplication(ECPM)

The proposed ECPM architecture is shown in *Figure 3* which provides high speed point operation. ECPM module consists of a data path unit, registers array, and a dedicated control unit to support a finite state machine.

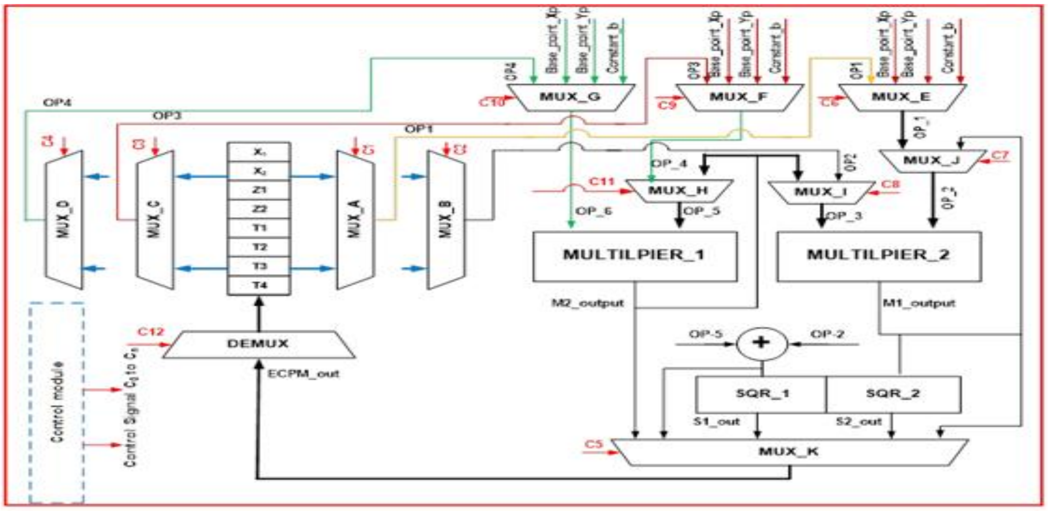


Figure 3 Proposed high-speed ECPM architecture

The basic requirements are elliptic curve parameters such as constant_b, base_point_Xp and base_point_Yp. All basic parameters are designated from NIST recommended application. The control unit generates control signals for the matching read or write operation for the memory unit and data path of the different multiplexers. The five control signals C1 to C5 are defined for memory-related functions and the remaining C6 to C11 are used for different directing purposes in the data path modules. The objective of the memory unit is to store the intermediary results during the implementation of PM operations. In the proposed architecture, there are two modular squarer and two modular multipliers submodule. The two modular multipliers operate parallel operations to boost the ECPM speed which directly improves the ECC operation speed.

3.2ECC system architecture

The proposed hardware of ECC accelerator is shown in *Figure 4*. It uses an XOR layer and a bit serial multiplier for performing kP multiplication. The Montgomery ladder concept is used for the addition and doubling process. Four-bit registers are requisite in the handing out of data and an extra m bit register is utilized for holding the inverse of the base point. The device inputs are considered as the curve constant, a base point, and scalar value. The final output is the outcome of kP and stored in the register.

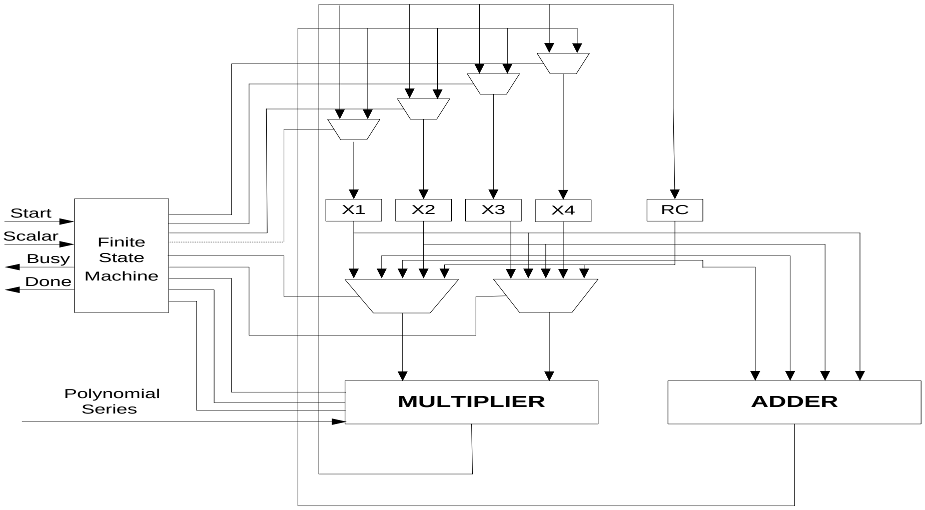


Figure 4 Proposed ECC accelerator architecture

The proposed architecture operates the inversion of the base point and fractional result is held in a specific register utilized during the inversion process. At the edge of data computation completion, the architecture design calculates a point conversion from projective to affine coordinates. The operand switch required in the Montgomery ladder is processed using multiple multiplexers to avoid data dependent register stores.

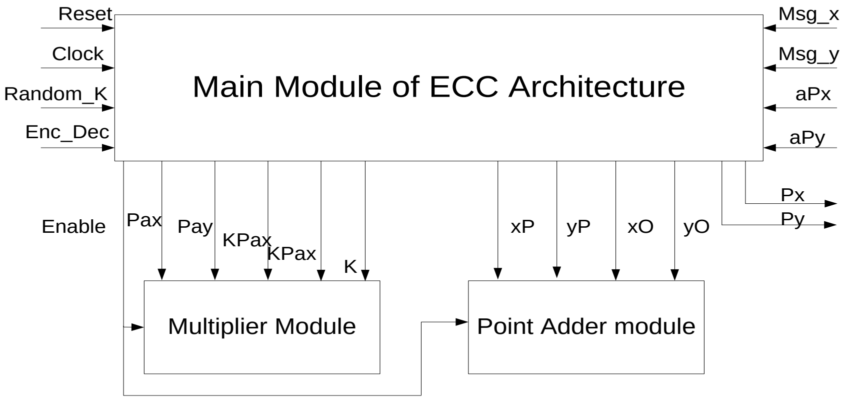


Figure 5 Main module of ECC architecture

The main module of ECC architecture is shown in *Figure 5*. The main module organizes the function of a multiplier and adder module. The adder module is processed when the Enable signal set to ‘1’. The adder executes the addition of input points on a considered elliptic curve. The multiplier module is processed when the Enable line is set to ‘0’. The multiplier executes the multiplication of an integer input with a base point on the elliptic curve. Successive addition is used to perform multiplication. The PA module performs a function as per elliptic curve arithmetic rules. The proposed accelerator architecture improves the speed of sub modules. The signal value of Enc_Dec decides encryption and decryption process of the message. The encryption, generates cipher message from the original message and decryption generates original message from cipher message. The main module has many internal signals. The function and meaning of each signal are described as below.

Reset: This signal is used to force all the components to their initial value.
Clock: The internal clock
Random_K: Random integer private key
Enc_Dec : Select encryption/decryption
Msg_x: Transmitted message with x-coordinate value
Msg_y: Transmitted message with y-coordinate value
aPx: x - coordinate of the measure “xP”
aPy: y - coordinate of the measure “yP”
kPax: Resultant with the x-coordinate
kPay: Resultant with the y-coordinate
xP: Point on the elliptic curve with the x-coordinate
yP: Point on the elliptic curve with the y-coordinate
Px: Final resultant with x-coordinate
Py: Final resultant with y-coordinate

Figure 6 shows the proposed block-level architecture of an ECC point multiplier. Double PM and PM are executed in accordance with PA and PD operation. The amount of precomputed values is taken into account, and the ECC design selects the input point P between the NIST-recommended base points. A state machine is utilised to execute PM and double PM based on standard projective coordinates, and this state machine controls the function flow required to transform the calculated point to the affine domain.

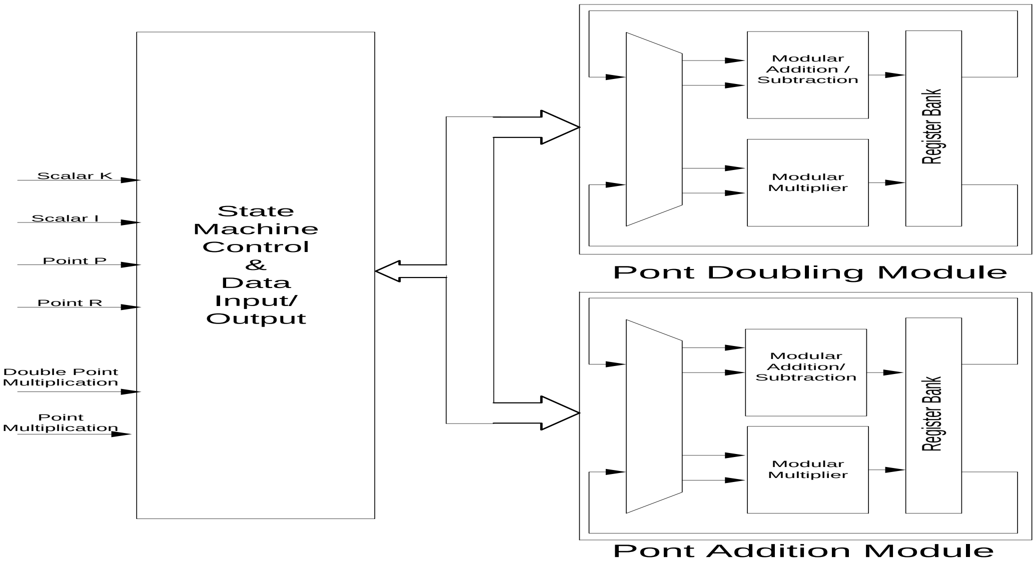


Figure 6 Proposed block level architecture of ECC point multiplier

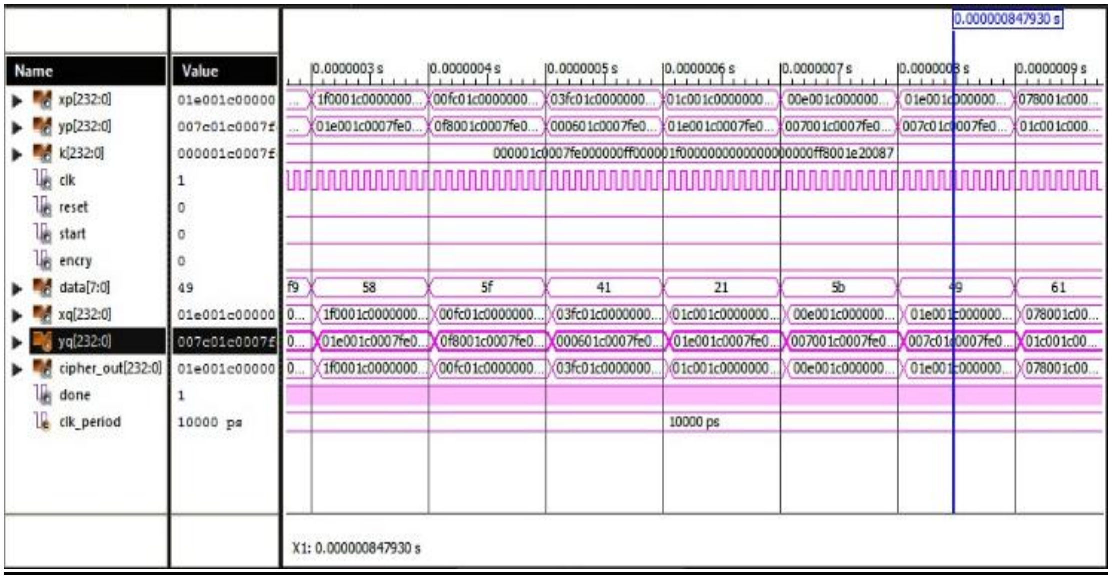
4.Results

The simulation of all the modules is performed on the Xilinx Vivado platform and all the analysis parameters such as CCs, speed, the maximum clock frequency, number of slices, and throughput are measured. The timing simulation for the encryption process and decryption process is shown in *Figure 7* and *Figure 8* respectively. The encryption process encrypts the original message using the elliptic curve and generates cipher text. The decryption process uses this cipher text and regenerates the original message. The simulated result helps to find the performance parameters. The proposed modular multiplier is simulated on Xilinx Vivado Design Suite for the Xilinx Virtex-7 FPGA device platform. For the simulation, NIST different field sizes as P-192, P-256, P-384, P-409, P-521, and P-571 have been considered. *Table 1* shows the performance parameter comparison with results reported in different literature for different NIST recommended field sizes.

There are many challenges to perform a higher processing speed with less utilization area because execution time and hardware area are two contradictory performance parameters of an FPGA based hardware implementation.

Implementation of modular multiplication has been presented in various literature. Most of the author has tried to optimize hardware area and execution time. Various literatures have a related area in terms of look up table (LUT) in place of the slice, so LUTs have been considered in this research study. The performance parameters are compared with the results reported in different literature for the various NIST recommended field sizes in *Table 2*.

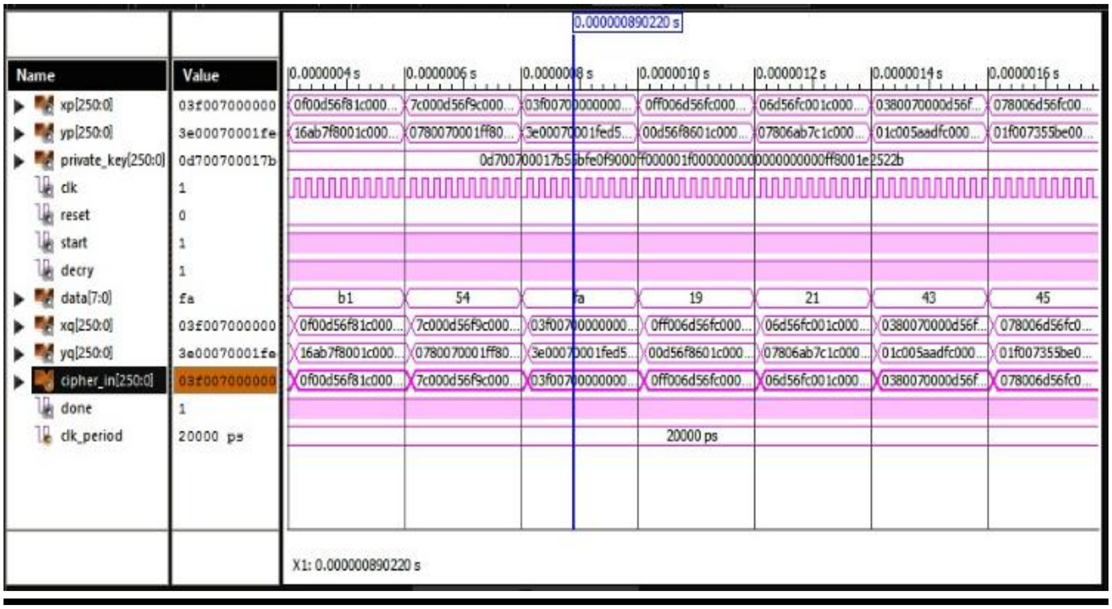
The analysis parameters are compared in *Figure 9* in terms of CC used and area x time. It shows that our proposed modular multiplication utilizes less area x time with compromising CC.



Name	Value
xp[232:0]	01e001c00000
yp[232:0]	007c01c0007f
k[232:0]	000001c0007f
clk	1
reset	0
start	0
encry	0
data[7:0]	49
xq[232:0]	01e001c00000
yq[232:0]	007c01c0007f
cipher_out[232:0]	01e001c00000
done	1
clk_period	10000 ps

X1: 0.000000847930 s

Figure 7 Simulation of the encryption module



Name	Value
xp[250:0]	03f007000000
yp[250:0]	3e00070001fe
private_key[250:0]	0d700700017b
clk	1
reset	0
start	1
decry	1
data[7:0]	fa
xq[250:0]	03f007000000
yq[250:0]	3e00070001fe
cipher_in[250:0]	03f007000000
done	1
clk_period	20000 ps

X1: 0.000000890220 s

Figure 8 Simulation of the decryption module

Table 1 Simulation results of modular multiplier design on Virtex-7

Field size	**Frequency(MHz)**	**CCs**	**Area (look up tables)**	**Time (µs)**	**Area x time (LUTs x ms)**	**Throughput (Mbps)**
192	403.5	130	880	0.55	0.48	204
256	380.6	153	1080	0.89	0.96	198.2
384	325.8	182	1350	1.05	1.42	180
409	302.3	243	1552	1.80	2.80	165
521	270.5	308	1988	2.90	5.77	152.5
571	260.8	366	2130	4.02	8.56	138.6

Table 2 Performance parameters comparison of the proposed modular multiplier

Reference work	**Platform**	**Field size**	**Clock cycles**	**Frequency (MHz)**	**Area (LUTs)**	**Time (μs)**	**Area x Time (LUTs x μs)**	**Remarks**
[32]	Virtex-7	192	135	520.2	2397	0.62	3.5	Radix-2 Montgomery multiplication architecture
		224	186	510	2505	0.79	4.02	
		256	213	501.2	2709	0.94	4.65	
		384	265	485.3	2809	1.51	5.62	
		521	232	479.5	3031	2.18	5.80	
[33]	Virtex-6	192	98	101.3	3020	0.97	2.9	Radix-4 booth encoded
		224	114	98.2	3427	1.16	4.0	
		256	130	95.2	3877	1.36	5.3	
[34]	Virtex-6	256	131	166	6300	0.79	5.0	Radix-4 interleaved
[35]	Virtex-6	192	97	92	11152	1.1	12.3	Radix-4 booth encoded interleaved
		256	129	86.6	18520	1.49	27.6	
		512	257	76.25	29916	3.37	100.8	
[36] (Design-1)	Virtex-6	192	48	101	3100	0.94	2.9	Radix-4 serial interleaved
		224	56	99	3400	1.13	3.8	
		256	64	96	3900	1.30	5.1	
[36] (Design-2)		192	48	171	4200	0.56	2.4	Radix-4 parallel interleaved
		224	56	167	4900	0.67	3.3	
		256	64	166	5300	0.77	4.1	
Proposed Design	Virtex-7	192	130	403.5	880	0.32	0.28	Radix-2 IMM
		256	153	380.6	1080	0.40	0.43	
		384	182	325.8	1350	0.56	0.75	
		409	243	302.3	1552	0.80	1.25	
		521	308	270.5	1988	1.14	2.26	
		571	366	260.8	2130	1.40	2.99	

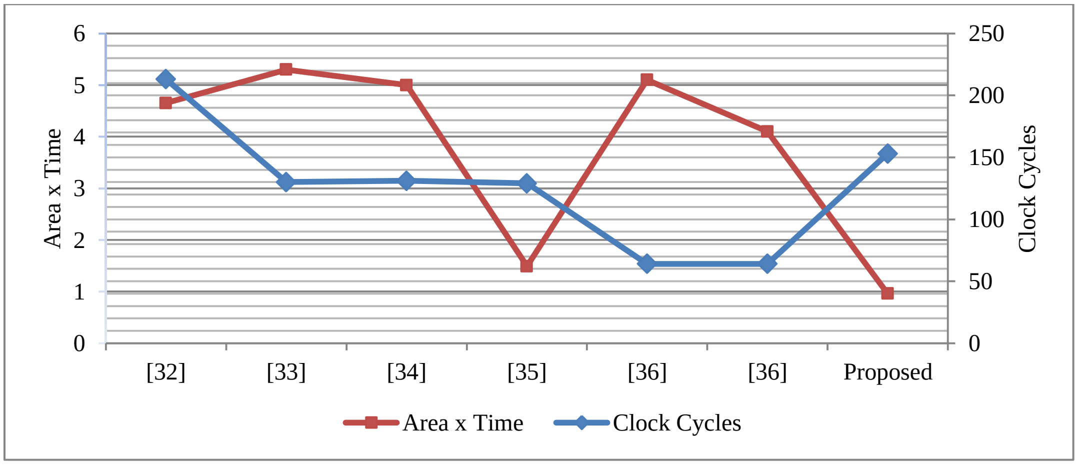


Figure 9 Comparison of design for GF (256)

The complete system achieves better performance parameters with the help of the proposed modular multiplier system. The sub modules of ECC architectures such as modular addition, modular multiplication, modular inversion, scalar multiplication, and public key generation are designed and simulated on Xilinx Virtex-6 and Virtex-7 platforms as represented as proposed design (a) and proposed design (b) respectively.

The various performance parameter analysis is presented in *Table 3.* The architecture is designed in such a way that modular multiplier and squarer modules are working with the parallel processing operation. This concept increases the modular operating speed directly, as seen by the comparison table. This gain in speed also contributes to the efficient architecture of ECCA performance parameters. *Table 4* presents a comparison between our suggested ECC system design and the findings reported in various publications over GF (256). Asif et al. [37] presented a cryptosystem based on a system of residue numbers. Shah et al. [38] suggested a redundant signed digit elliptic curve cryptography scheme based on the NIST-recommended prime field of 256. It takes 0.47ms and 65,600 slices to run the system's procedure. Asif et al. [39] developed architecture for deep pipelining to encrypt 21 keys concurrently. Hossain et al. [40] implemented ECC processor on NIST 256 prime field and performs the crypto process with less throughput. They have implemented the design on the Xilinx Virtex-5 FPGA board. Yang et al. [41] proposed the unified PA with a twisted Edwards curve and implemented it on Virtex-6 and Virtex-7 FPGA board. To understand the performance of cryptosystem, all the designs are compared in *Figure 10* with the consideration of execution time and number of slices utilized in the hardware platform. The performance parameter of the ECC components on the Xilinx Virtex-7 platform is a modest bit higher related to the Xilinx Virtex-6 platform in terms of speed, but the area utilized by the various segments on this FPGA is nearly remaining similar.

Figure 10 shows that our proposed architecture utilizes a minimum number of the slice with a compromise of execution time. The efficient implementation helps higher data security in resource constraint applications.

Table 3 Performance analysis of ECC various modules over GF (256)

Basic operation	Platform	Clock cycle (CCs)	Area (Slices)	Area (LUTs)	Maximum frequency (MHz)	Time (µs)	Throughput* Mbps)
Modular Addition	Virtex-6	52	126	405	109.2	0.48	537.60
	Virtex-7	52	112	380	112.2	0.46	552.37
Modular Multiplication	Virtex-6	121	380	1252	100.2	1.21	211.99
	Virtex-7	121	370	1211	105.6	1.15	223.42
Modular Inversion	Virtex-6	290	985	3860	100.2	2.89	88.45
	Virtex-7	290	960	3540	101.3	2.86	89.42
ECPM	Virtex-6	158203	3825	19204	98.2	1611.03	0.16
	Virtex-7	158203	3710	18506	99.1	1596.40	0.16
Public-key Generation	Virtex-6	166534	4503	20530	98.6	1646.36	0.15
	Virtex-7	164331	4380	20210	103.2	1572.97	0.16

*Throughput = (Maximum frequency÷ CCs) ×256.

Table 4 Performance parameter comparison of ECC system

Reference work	Galois field (GF)	Platform	Clock frequency (MHz)	Clock cycle (CCs)	Number of slices (K)	Time (ms)	Throughput (Mbps)	Remarks
[37]	256	Virtex-7	72.9	215.9	24.2	2.96	1816.2	Cryptosystem based on residue number system
[38]	256	Virtex-5	66.7	442.2	10.2	6.63	38.61	NIST 256 prime bit ECC processor
[41]	256	Virtex-6	93.23	198.5	6.6	2.13	120.12	Unified PA with twisted Edwards curve
[42]	256	Kintex-7	121.5	397.3	11.3	3.27	78.28	ECC over

								NIST prime field
[43]	256	Virtex-6	327	153.2	65.6	0.47	546.42	Redundant signed digit based elliptic curve cryptographic system
[44]	256	Virtex-5	160	361.6	8.7	2.26	113.27	RSD-based elliptic curve cryptographic system
[45]	256	Virtex-7	104.39	198.3	6.5	1.9	134.49	Unified PA with twisted Edwards curve
Proposed design (a)	256	Virtex-7	99.1	158.2	3.7	1.5	160.2	Radix-2 IMM and Montgomery scalar multiplication algorithm
Proposed Design (b)	256	Virtex-6	98.2	165.3	3.8	1.6	160.9	Radix-2 IMM and Montgomery scalar multiplication algorithm

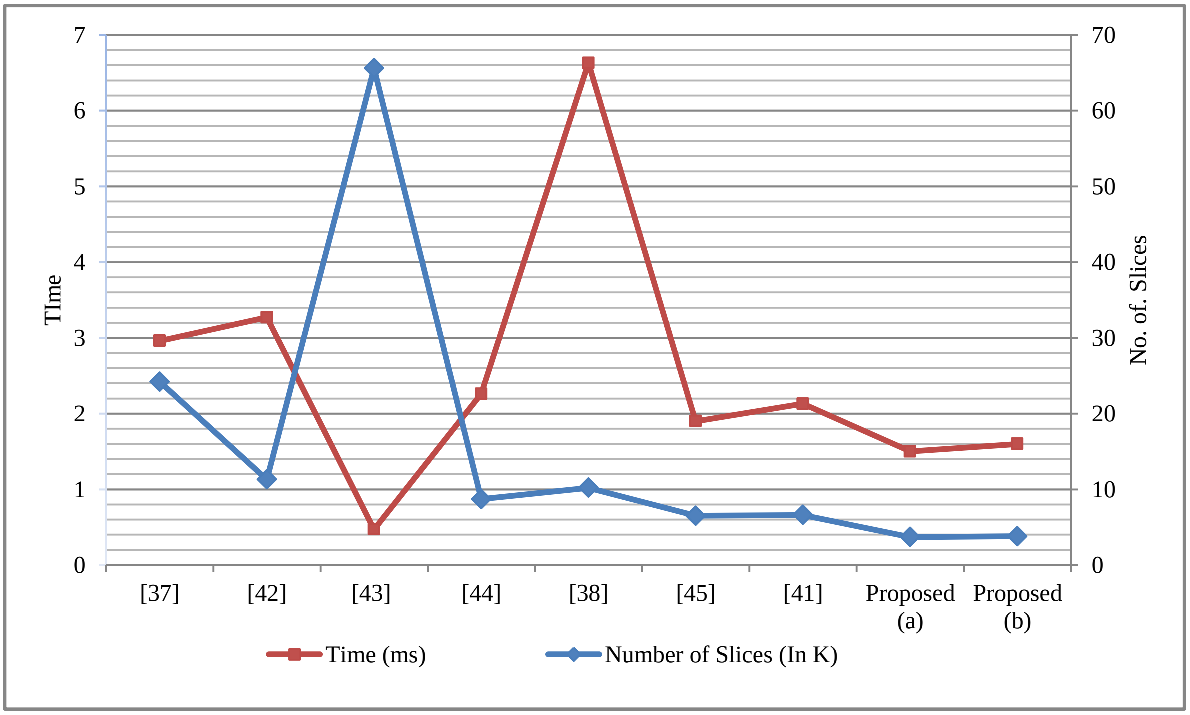


Figure 10 Comparison of ECC system design

5.Discussion

The analysis proves that the utilisation area will increase as we perform parallel modular multiplication and PM to increase speed. The trade-off leads to more devices and increased power consumption. In comparison to any previous literature implementation, the speed and slice utilization of our proposed system, implemented on a Xilinx Virtex-7, are improved. Therefore, applications considering resource constraints will adopt the proposed approach. The recommended prime field size by NIST affects ECC performance as well. The performance difficulty and processing parameter computation change when we select a higher prime field. The FPGA device technology and metal oxide semiconductor field effect (MOSFET) Transistor MOSFET size affect the ECC performance parameters. Here we used Xilinx Virtex-7 device but when we use Xilinx's latest devices such as Ultrascale and Ultrascale+, the optimization will be improved. The Xilinx provides the system on chip (SoC) device which contains FPGA device as well as a processor inside the chip. The system's adaptability can be increased by implementing the hardware-software (HW/SW) co-design approach. A higher speed for cryptography operations will be achieved by the accelerometer of the processing operation. The implemented ECC system is limited to 521 bits Galois field and real time data input size and type. To solve the limitation, the ECC system should be designed and implemented on Xilinx Ultrascale+ series board.

A complete list of abbreviations is shown in *Appendix I.*

6.Conclusion and future work

In this paper, an area efficient and a high speed elliptic curve based cryptosystem has been proposed for NIST P-256, which provides higher security with less resource utilization. The main module of the ECC is ECSM which is implemented using Montgomery scalar multiplication algorithm. All the modules of ECC are designed on the Xilinx Virtex-6 and Virtex-7 FPGA device platforms over a NIST recommended 256 bits prime field. The ECC system over GF (256) can be performed in 1.5ms at 99.1 MHz maximum frequency in Xilinx Virtex-7 FPGA platform using 3.7k slices. It is the fastest implementation outcome compared with other literature designs. It presents higher efficiency in terms of throughput without affecting the security level. This proposed module can also be simply extended to support all NIST-recommended fields by making a few changes. In the future, we can implement our design on Xilinx Ultrascale and Xilinx Ultrascale+ devices for better performance.

Acknowledgment

None.

Conflicts of interest

The authors have no conflicts of interest to declare.

Authors' contribution statement

Kirit Patel: Prepared the literature review of ECC implementation, modeling, simulation and hardware setup, implementation of design, preparing, and writing the research article. **Mihir Shah:** Proofread the article, reviewed the article and Editing. **Pankaj Prajapati:** Proofread the article, reviewed the article and Editing.

References

[1] Marzouqi H, Al-qutayri M, Salah K. Review of elliptic curve cryptography processor designs. Microprocessors and Microsystems. 2015; 39(2):97-112.

[2] Lara-nino CA, Diaz-perez A, Morales-sandoval M. Elliptic curve lightweight cryptography: a survey. IEEE Access. 2018; 6:72514-50.

[3] Islam MM, Hossain MS, Hasan MK, Shahjalal M, Jang YM. Design and implementation of high-performance ECC processor with unified point addition on twisted edwards curve. Sensors. 2020; 20(18):1-19.

[4] Islam MM, Hossain MS, Shahjalal MD, Hasan MK, Jang YM. Area-time efficient hardware implementation of modular multiplication for elliptic curve cryptography. IEEE Access. 2020; 8:73898-906.

[5] Abu KS, Abdulrahman SE, Ismail NA. Towards efficient FPGA implementation of elliptic curve crypto-processor for security in IoT and embedded devices. Menoufia Journal of Electronic Engineering Research. 2020; 29(2):106-18.

[6] Kashif M, Cicek I. Field-programmable gate array (FPGA) hardware design and implementation of a new area efficient elliptic curve crypto-processor. Turkish Journal of Electrical Engineering and Computer Sciences. 2021; 29(4):2127-39.

[7] Rashidi B. A survey on hardware implementations of elliptic curve cryptosystems. Electrical Engineering and Systems Science. 2017:1-61.

[8] Patel KV, Shah MV. Analysis of efficient implementation of elliptic curve cryptography architecture for resource constraint application. International Journal of Innovative Technology and Exploring Engineering. 2021; 10(12):28-35.

[9] Patel KV, Shah MV. Implementation of generic and efficient architecture of elliptic curve cryptography over various GF (p) for higher data security. ADBU Journal of Engineering Technology. 2020; 9(2):1-7.

[10] Sajid A, Rashid M, Imran M, Jafri AR. A low-complexity edward-curve point multiplication architecture. Electronics. 2021; 10(9):1-16.

[11] Mehrabi MA, Doche C, Jolfaei A. Elliptic curve cryptography point multiplication core for hardware security module. IEEE Transactions on Computers. 2020; 69(11):1707-18.

[12] Li J, Wang W, Zhang J, Luo Y, Ren S. Innovative dual-binary-field architecture for point multiplication of elliptic curve cryptography. IEEE Access. 2021; 9:12405-19.

[13] Kudithi T. An efficient hardware implementation of the elliptic curve cryptographic processor over prime

field. International Journal of Circuit Theory and Applications. 2020; 48(8):1256-73.

[14] Arunachalam K, Perumalsamy M. FPGA implementation of time-area-efficient elliptic curve cryptography for entity authentication. Informacije MIDEM. 2022; 52(2):89-103.

[15] Wen J, Wu N, Ge F, Zhao L. A length-scalable modular multiplier implemented with multi-bit scanning. In 4th international conference on electronics technology (ICET) 2021 (pp. 109-13). IEEE.

[16] Hu X, Huang H, Zheng X, Liu Y, Xiong X. Low-power reconfigurable architecture of elliptic curve cryptography for IoT. IEICE Transactions on Electronics. 2021; 104(11):643-50.

[17] Zhao X, Li B, Zhang L, Wang Y, Zhang Y, Chen R. FPGA implementation of high-efficiency ECC point multiplication circuit. Electronics. 2021; 10(11):1-22.

[18] Zode P, Deshmukh R. Optimization of elliptic curve scalar multiplication using constraint based scheduling. Journal of Parallel and Distributed Computing. 2022.

[19] Abd-elkader AA, Rashdan M, Hasaneen ES, Hamed HF. Efficient implementation of montgomery modular multiplier on FPGA. Computers & Electrical Engineering. 2022.

[20] Bisheh NM, Azarderakhsh R, Kermani MM. Efficient hardware implementations for elliptic curve cryptography over curve448. In international conference on cryptology in India 2020 (pp. 228-47). Springer, Cham.

[21] Kumar H, Rashid M, Alhomoud A, Khan SZ, Bahkali I, Alotaibi SS. A scalable digit-parallel polynomial multiplier architecture for NIST-standardized binary elliptic curves. Applied Sciences. 2022; 12(9):1-18.

[22] Bisheh-niasar M, Azarderakhsh R, Mozaffari-kermani M. Cryptographic accelerators for digital signature based on Ed25519. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2021; 29(7):1297-305.

[23] Alinejad M, Hassan ZS, Biranvand N. Digital signature with elliptic curves over the finite fields. Journal of Discrete Mathematical Sciences and Cryptography. 2022; 25(5):1289-1301.

[24] Guo X, Fan J, Schaumont P, Verbauwhede I. Programmable and parallel ECC coprocessor architecture: tradeoffs between area, speed and security. In international workshop on cryptographic hardware and embedded systems 2009 (pp. 289-303). Springer, Berlin, Heidelberg.

[25] Zode P, Deshmukh R. Novel fault attack resistant architecture for elliptic curve cryptography. Microprocessors and Microsystems. 2021; 84:1-7.

[26] Hemambujavalli S, Nirmal KP, Jose D, Anthoniraj S. FPGA implementation of elliptic curve point multiplication over galois field. In mobile radio communications and 5G networks 2022 (pp. 619-33). Springer, Singapore.

[27] Yang G, Kong F, Xu Q. Optimized FPGA implementation of elliptic curve cryptosystem over prime fields. In 19th international conference on trust, security and privacy in computing and communications (TrustCom) 2020 (pp. 243-9). IEEE.

[28] Gookyi DA, Ryoo K. A lightweight system-on-chip based cryptographic core for low-cost devices. Sensors. 2022; 22(8):1-28.

[29] Maimuţ D, Matei AC. Speeding-up elliptic curve cryptography algorithms. Mathematics. 2022; 10(19):1-13.

[30] Di MS, Baldanzi L, Crocetti L, Nannipieri P, Fanucci L, Saponara S. Secure elliptic curve crypto-processor for real-time IoT applications. Energies. 2021; 14(15):1-28.

[31] Imran M, Pagliarini S, Rashid M. An area aware accelerator for elliptic curve point multiplication. In 27th international conference on electronics, circuits and systems (ICECS) 2020 (pp. 1-4). IEEE.

[32] Coliban RM. Fast radix-2 montgomery modular multiplication on FPGA using ternary adder. In international conference on computing, electronics & communications engineering (iCCECE) 2022 (pp. 1-5). IEEE.

[33] Javeed K, Wang X, Scott M. High performance hardware support for elliptic curve cryptography over general prime field. Microprocessors and Microsystems. 2017; 51:331-42.

[34] El AA, Rodriguez E, Orabi M, Alarcon E. Modeling of switching frequency instabilities in buck-based DC–AC H-bridge inverters. International Journal of Circuit Theory and Applications. 2011; 39(2):175-93.

[35] Javeed K, Wang X. Radix-4 and radix-8 booth encoded interleaved modular multipliers over general Fp. In 24th international conference on field programmable logic and applications (FPL) 2014 (pp. 1-6). IEEE.

[36] Javeed K, Wang X, Scott M. Serial and parallel interleaved modular multipliers on FPGA platform. In 25th international conference on field programmable logic and applications (FPL) 2015 (pp. 1-4). IEEE.

[37] Asif S, Hossain MS, Kong Y, Abdul W. A fully RNS based ECC processor. Integration. 2018; 61:138-49.

[38] Shah YA, Javeed K, Azmat S, Wang X. Redundant-signed-digit-based high speed elliptic curve cryptographic processor. Journal of Circuits, Systems and Computers. 2019; 28(5):2-33.

[39] Asif S, Hossain MS, Kong Y. High-throughput multi-key elliptic curve cryptosystem based on residue number system. IET Computers & Digital Techniques. 2017; 11(5):165-72.

[40] Hossain MS, Kong Y, Saeedi E, Vayalil NC. High-performance elliptic curve cryptography processor over NIST prime fields. IET Computers & Digital Techniques. 2017; 11(1):33-42.

[41] Yang Y, Ng EJ, Chen Y, Flader IB, Kenny TW. A unified epi-seal process for fabrication of high-stability microelectromechanical devices. Journal of Microelectromechanical Systems. 2016; 25(3):489-97.

[42] Matutino PM, Araújo J, Sousa L, Chaves R. Pipelined FPGA coprocessor for elliptic curve cryptography based on residue number system. In international

conference on embedded computer systems: architectures, modeling, and simulation 2017 (pp. 261-8). IEEE.

[43] Benaissa M. Throughput/area-efficient ECC processor using montgomery point multiplication on FPGA. IEEE Transactions on Circuits and Systems II: Express Briefs. 2015; 62(11):1078-82.

[44] Lara-nino CA, Diaz-perez A, Morales-sandoval M. Lightweight elliptic curve cryptography accelerator for internet of things applications. Ad Hoc Networks. 2020; 103:1-9.

[45] Marzouqi H, Al-qutayri M, Salah K. An FPGA implementation of NIST 256 prime field ECC processor. In 20th international conference on electronics, circuits, and systems 2013 (pp. 493-6). IEEE.

Kirit Patel, is currently working as an assistant professor in the Electronics and Communication (EC) department at L.D. College of Engineering, Gujarat, India. He is pursuing a Ph.D. degree from Gujarat Technological University. He has received MTech. Degree in EC with a specialization in VLSI Design from the Institute of Technology, Nirma University in 2009. He has received a BE degree from VNSGU, Gujarat in 2006. He has more than 12 years of teaching experience and published 10 research papers in International /National journals/conferences. His main area of research is a cryptography and VLSI Front End design.
Email: kiritvlsi@gmail.com

Dr. Mihir Shah, is currently working as an adjunct professor, Institute of Technology, Nirma University Gujarat, India. He is awarded a Ph.D. degree from MSU, Baroda, Gujarat in 2009. He has received the M.E. degree in EC from Malaviya Regional Engineering College Jaipur, Rajasthan in 2001. He has 4 years of industry experience and more than 24 years of teaching experience. He has published more than 30 research papers in International/National Journal / Conference. His main area of research is the VLSI Front End design and CMOS analog design.
Email: mihirec@gmail.com

Dr. Pankaj Prajapati is currently working as an assistant professor in the Electronics and Communication (EC) Department at L. D. College of Engineering, Gujarat, India. He has completed his PhD in the area of Optimization of CMOS-based Analogue Circuit from the Gujarat Technological University (GTU), Ahmedabad in 2019. He haceived B.E. in EC Engineering from L. D. College of Engineering in 2001 and MTech. in EC with a spspecialisation VLSI Design from the Institute of Technology, Nirma University in 2009. He has published many research papers in International/National journals/ conferences.
Email: pankaj@ldce.ac.in

Appendix I

S. No.	Abbreviation	Description
1	ASIC	Application Specific Integrated Circuit
2	CC	Clock Cycle
3	ECC	Elliptic Curve Cryptography
4	ECPM	Elliptic Curve Point Multiplication
5	ECSM	Elliptic Curve Scalar Multiplication
6	FPGA	Field Programmable Gate Array
7	GF	Galois Field
8	HW/SW	Hardware - Software
9	IMM	Interleaved Modular Multiplication
10	IoT	Internet-of-Things
11	LUT	Look Up Table
12	MMM	Montgomery Modular Multiplier
13	MOSFET	Metal Oxide Semiconductor Field Effect Transistor
14	NIST	National Institute of Standards and Technology
15	PA	Point Addition
16	PD	Point Doubling
17	PM	Point Multiplication
18	SoC	System on Chip