Original scientific paper
https://doi.org/10.33180/InfMIDEM2022.203

Informacije MIDEM
Journal of Microelectronics,
Electronic Components and Materials
Vol. 52, No. 2(2022), 89 – 103

Programmable implementation of time-area-efficient Elliptic Curve Cryptography for entity authentication

Kamaraj Arunachalam[1], Marichamy Perumalsamy[2]

[1]Department of ECE, Mepco Schlenk Engineering College, Sivakasi, India
[2]Department of ECE, PSR Engineering College, Sivakasi, India

Abstract: The rise of wireless technologies, communications and devices, has resulted in the demand for effective security with low hardware requirements and high speed. Among the various cryptographic algorithms, the Elliptic Curve Cryptography (ECC) provides an attractive solution for this demand. In this paper, the Remote Keyless system (RKE) Authentication process using the ECC is implemented in Field Programmable Gate Array (FPGA). The designed ECC processor supports 256-bit point multiplication and point addition on the Koblitz curve secp256k1. The scalar multiplication is performed with the faster multiplier Urdhva Tiryagbhyam (UT). Additionally, pipelining is incorporated in order to speed up the multiplication process of the processor. The proposed ECC processor performs single point multiplication of 256-bit in 1.2062ms with a maximum clock frequency of 192.5MHz, which provides 212.23kbps throughput and occupies 8.23k slices in Virtex-7 FPGA. Incorporating a pipeline in scalar multiplication improves the maximum clock frequency up to 15.12%, which reduces time consumption by 22.36%, which in turn increases the throughput by 22.36%. The proposed pipelined Vedic multiplier based ECC processor outperforms the existing designs in terms of area, operating frequency, area-delay product and throughput. Also, the security evaluation and analysis of the proposed ECC processor are performed, which ensures the safety of RKE systems. Hence, the implementation of the proposed method offers time-area-efficient and fast scalar multiplication with effective hardware utilization without any compromise in security level.

Keywords: Urdhva Tiryagbhyam; Pipeline; Remote Keyless system Authentication; FPGA

Programirljivo izvajanje časovno učinkovite kriptografije eliptičnih krivulj za avtentikacijo entitet

Izvleček: Razvoj brezžičnih tehnologij, komunikacij in naprav je povzročil potrebo po učinkoviti varnosti z nizkimi strojnimi zahtevami in visoko hitrostjo. Med različnimi kriptografskimi algoritmi zagotavlja eliptična krivulja (ECC) privlačno rešitev za to. V tem članku je predstavljen postopek avtentikacije sistema brez ključa na daljavo (RKE) z uporabo ECC v FPGA (Field Programmable Gate Array). Zasnovani procesor ECC podpira 256-bitno množenje in seštevanje točk na Koblitzovi krivulji secp256k1. Skalarno množenje se izvaja s hitrejšim množiteljem Urdhva Tiryagbhyam (UT). Poleg tega je za pospešitev postopka množenja v procesorju vključena cevna povezava (pipelining). Predlagani procesor ECC izvede enotočkovno množenje 256-bitov v 1,2062 ms z največjo taktno frekvenco 192,5 MHz, kar zagotavlja prepustnost 212,23 kb/s in zasede 8,23k rezin v Virtex-7 FPGA. Vključitev cevovoda pri skalarnem množenju izboljša največjo taktno frekvenco do 15,12 %, kar zmanjša porabo časa za 22,36 %, to pa poveča prepustnost za 22,36 %. Predlagani procesor ECC, ki temelji na množitelju s cevovodi Vedic, je boljši od obstoječih modelov glede površine, delovne frekvence, produkta površine in zakasnitve ter prepustnosti. Izvedena sta tudi varnostna ocena in analiza predlaganega procesorja ECC, ki zagotavlja varnost sistemov RKE. Izvedba predlagane metode torej omogoča časovno učinkovito in hitro skalarno množenje z učinkovitim izkoristkom strojne opreme brez kompromisov na ravni varnosti.

Ključne besede: Urdhva Tiryagbhyam; Cevovod; avtentikacija sistema brez ključa na daljavo; FPGA

** Corresponding Author's e-mail: kamarajvlsi@gmail.com*

How to cite:
K. Arunachalam et al., "Programmable implementation of time-area-efficient Elliptic Curve Cryptography for entity authentication", Inf. Midem-J. Microelectron. Electron. Compon. Mater., Vol. 52, No. 2(2022), pp. 89–103

1 Introduction

Nowadays, almost all cars are equipped with a smart keyless entry system. This is an electronic lock that controls admittance to a vehicle without utilizing a manual mechanical key. Such frameworks currently have a secret touch-enacted keypad, which is as yet accessible on certain Ford and Lincoln models. This method is termed as Remote Keyless System (RKS). A distant keyless framework can incorporate both a remote keyless entry (RKE), which opens the car door and a remote keyless ignition system (RKI), which turns the engine ON.

On account of the innovative keyless technology, programmers can utilize expert systems to fool the vehicle and to make them believe that the right fob is close by, permitting them access. Such attacks are listed as below [1],

1. **Replay Attack:** A replay attack (otherwise called playback attack) is a type of organized attack in which valid information transmission is perniciously or falsely rehashed or delayed [2].
2. **Rolljam Attack:** The rolljam attack works by recording and blocking the radio signal from the key fob. Because the signal is blocked, the car doesn't unlock and the owner will naturally try again. That creates a second signal that is also recorded and blocked, but this time the attacker replays the first code to unlock the door.
3. **Brute Force Attack:** Brute force attacks are simple and reliable. Attackers let a computer do the work – trying different combinations of usernames and passwords until they find one that works.

Some other attacks are radio jamming attack, scan attack and two-thief attack, which are also major attacks in Remote Keyless Entry (RKE) and Passive Remote Keyless Entry (PRKE) systems, which are shown in Figure 1.

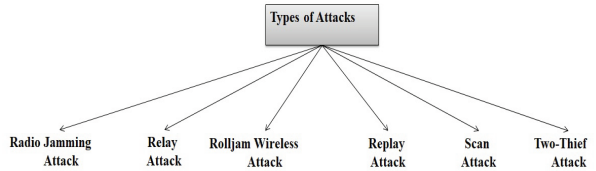


Figure 1: Types of Attacks in Remote Key Fob [1]

In order to overcome these attacks, a high level of security algorithms is needed for secure communication. The public key cryptography based authentication has no secret information to be shared between the entities. A user requesting to authenticate him has to use his private key to digitally sign a random number, which is issued by the verifying entity. This random number is a time-variant parameter and is unique to the authentication exchange. If the verifier completes the verification of the signed response of the user, then authentication would be granted. These kinds of authentication methods are widely popular in sensor networks. In such scenarios, strong encryption algorithms are required to avoid mischief [3]. This kind of entity authentication is to be initiated by the user that can be an equivalent word. This equivalent word checked by the verifier [4].

Elliptic-curve cryptography [5] is a public key cryptography based on the algebraic structure of elliptic curves over finite fields. Elliptic curves are used in key agreement, digital signatures, pseudo-random generators and also in performing other tasks. In contrast to Rivest–Shamir–Adleman (RSA), the ECC approach is based on how elliptic curves are structured algebraically over finite fields. Therefore, ECC creates keys that are more mathematically difficult to crack [6]. Hence ECC is considered to be the next generation public key cryptography and it is more secure than RSA. ECC cryptography can provide strong security with a 164-bit key as other cryptosystems realize the same level of security with a 1024-bit key or more. With the advent of mobile devices being used for highly secured private transactions, low-overhead encryption schemes are becoming highly desirable in today's applications.

The effectiveness of ECC can be improved by modifying its computations process. Vedic mathematics is a collection of procedures to solve complex mathematical functions effectively. It was invented by Sri Bharati Krishna Tirthaji from the Indian Veda scriptures. It consists of 16 sutras, among which Urdhva Tiryakbhayam, Nikhilam Sutram and Anurupye are the most widely used sutras for solving complex functions. The significant gain of the Vedic multipliers is that they have simple procedures for the resource consuming multiplications [7]. The multiplication operation can be extended to n-bits with some minor modifications.

The factual meaning of the Urdhva Tiryagbhyam Sutra is "Vertically and Crosswise". The vertical and crosswise manipulation is performed to generate the partial products; then they are summed for final product generation. 2×2 is the basic module of the Vedic multiplier. The n-bit multiplier could be derived by the repeated arrangement of 2×2 multipliers. This process makes the computation fast and the product is generated with a fewer number of steps [8].

2 Literature survey

Multiple attacks are strategized based on the technology used on the fob. A powerful attack will result in a definite loss to the user. There are many kinds of attacks

that are reported regarding the car locking mechanism. Wireless communication systems used for RKE seem to be more vulnerable to attack. Furthermore, the type of cryptographic algorithms deployed also limits the security in those systems [1].

A simple relay based passive keyless entry was constructed and tested for 10 cars from 8 different manufacturers at various physical distances. This methodology is an initiation towards the remote keyless car. These countermeasures carried against the attack itself act as a hindrance to the keyless operation [9]. In [10], a symmetric key based remote keyless secure transmission between car and fob was demonstrated, which provides secure communication against scan attack, playback attack and forward prediction attack. It requires less computation and consumes less energy with a message length of 80 bits. But, it requires frequent key updates by the user for better security. There is no safety without security in the progressively interconnected nature of a vehicle's control modules.

But, the hackers intercept the car's remote key details while the owner is using them. Then, these intercepted details are utilized to unlock the car door without the knowledge of the owner. Recently, 'Universal Remote' [11], 'EvanConnect', 'keyless repeater' [12] and power amplifier are used to hack Honda, Toyota, Volvo, Volkswagen and Jaguar cars. These hackers capitalized on the communication design flaw present in the design of the protocol. Keeloq system from Microchip has been broken by the University of Bochum or NXP's Hitag-2 system. At Fraunhofer AISEC the effectiveness of ECC in RKE applications was demonstrated. It was prototyped with the support of Field Programmable Gate Array (FPGA) [13].

A remote keyless system is widely used in automobile industries to lock or unlock the vehicle's door. But the security of the remote keyless system is threat prone since the beginning. Initially, Advanced Encryption Standard (AES) based wireless protocol with fixed and variable key length system was introduced [14, 15]. Here, a maximum of 128-bit AES is used for encryption, which could provide effective security against three types of attacks [14]. But, they are implemented in an 8-bit ATMega128L microcontroller, which has a very low speed of operation. Thereafter, FPGA implementation of secured Controller Area Network (CAN) bus communication was developed with AES for internal vehicular communication [15].

An ECC protocol developed in Python can provide security against 9 different attacks, which is economical compared to its predecessors [16]. Thereafter [17], a software protocol was developed based on ECC towards authentication of smart remote vehicle control, which could provide security against 12 kinds of attacks.

In the modern-day scenario, almost all automobiles are equipped with a remote keyless system. Hence the security of the communication system should be effective. Only a few studies focused on this issue. The presently available studies are lacking in,

- Effective Hardware implementation [14-17].
- Proper key size against the attack [14-15].

Also, according to Alan Grau [18], key fob fails due to Encryption keys generated from public data along with insufficient entropy for generating encryption keys, Discoverable encryption keys, and Deprecated key strength. It was suggested that deploying asymmetric encryption with proper key length on suitable hardware will improve the key fob encryption [18].

An effective scalar point multiplication for the elliptic curve is introduced. Then the critical path of the scalar point multiplication for the Lopez-Dahab curve is rearranged and reordered in such a way that parallel processing is enabled and the critical path operations are shifted to non-critical paths [6]. The point multiplication in ECC is a time consuming and slow process. Now, the ECC point multiplication is performed with Urdhva Tiryagbhyam Vedic multiplication [19]. The UT performs significantly better in terms of delay and logic levels compared to the conventional multiplier [20, 21]. Especially, the Vedic multiplier surpasses the performance of the Karatsuba multiplier in terms of area and delay; in addition to that, the UT has 90% less delay compared to the Booth multiplier. Even though the Booth multiplier is occupying less area, the delay for a single product generation is 287ms, which is 10 times higher than the Vedic multiplier [22] as shown in Table 1. Also, UT exhibits smaller path delay, logic delay, routing delay and dynamic power. The size of the UT may be 16-bit [19], 32-bit [23] and can be extended as desired.

Table 1: Performance of Vedic, Karatsuba and Booth multiplier [22]

Parameter	Vedic	Karatsuba	Booth
No. of slice LUTs	51761	103246	1937
No. of IOBs	640	640	643
Time Delay (ns)	27.172	34.123	287.840
Area Delay Product	0.001406	0.003523	0.000558

Hence, a more secured and high performance remote keyless system can be developed using a hybrid of ECC incorporated with Vedic multiplier, which is to be implemented in FPGA.

3 Mathematical background

In this paper, the Koblitz curve is considered with secp256k1 for ECC as shown in Figure 2. This elliptic curve has the form of $y^2 = x^3 + ax + b$, in which a = 0, b = 7, whose Prime Field (p) = $2^{256} - 2^{32} - 977$ and in the random case we have considered,

Order (n) = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFC2F

Base Point (G) = 04 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B 16F81798 483ADA77 26A3C465 5DA4FBFC 0E1108A8 FD17B448 A6855419 9C47D08F F

Addition: Let P = $(x_1, y_1) \in (K)$ and Q = $(x_2, y_2) \in E(K)$, where P = ±Q. Then P + Q = (x_3, y_3), where, $x_3 = (y_2 - y_1 / x_2 - x_1)^2 - 2x_1 - x_2$ and $y_3 = (y_2 - y_1 / x_2 - x_1)^2 - (x_1 - x_3) - y_1$

Point Doubling: Let P = $(x_1, y_1) \in E(K)$, where P = -P. Then 2P = (x_3, y_3), where: $x_3 = (3x_1^2 + a/2y_1)^2 - 2x_1$ and $y_3 = (3x_1^2 + a/2y_1)^2 - (x_1 - x_2) - y_1$

According to Hankerson, Menezes, and Vanstone [24], the primary advantage of the Koblitz lies in the possibility of implementing ECC without point doublings when performing ECC Point Multiplication (ECPM).

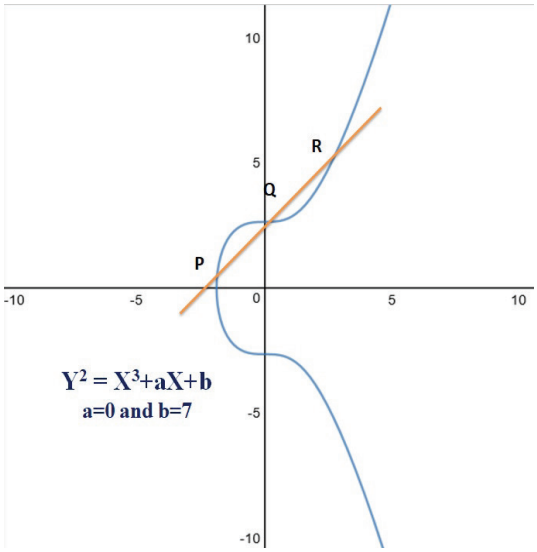


Figure 2: Diagram of Elliptic Curve

The Urdhva Tiryagbhyam (UT) Vedic multiplication is as follows,

Algorithm 1 VEDIC ALGORITHM

INPUT: n- bit Multiplicand and Multiplier
OUTPUT: 2n- bit product
$k \leftarrow 0$
S(k): 2n- bit vector initialized to 0
for $i = 0$ **to** n-1 **do**
  for $j = 0$ **to** i **do**
    $S(k) = S(k) + a(i) \times b(i - j)$
  end
  $k = k + 1$
end
for i = n-1 **to** 1 **do**
  for j = n-1 **to** i **do**
    $S(k) = S(k) + a(i) \times b(n - (i - j))$
  end
  $k = k + 1$
end
for $i = 0$ **to** $(k - 1)$ **do**
  $P = P + S(i)$
end

4 Main contribution

A time-area-efficient 256-bit ECC processor over prime field is implemented in FPGA. It is aimed to reduce the area and the delay for single point multiplication and increase the frequency and the throughput. In order to reach these objectives, the following major contributions are made in this paper,

An efficient design for ECPM on a Koblitz curve secp256k1 for the 256-bit prime field is proposed.

A faster Urdhva Tiryagbhyam multiplier is adopted for ECPM scalar multiplication, which reduces computation time.

The computing frequency is further improved by incorporating the pipeline technique in the Vedic multiplier.

Moreover, the area-delay product, throughput and efficiency of the proposed method shows improvement compared to the existing similar works in the literature.

5 Methodology

This section presents the algorithms, hardware architectures for point addition, point multiplication, modular multiplication and pipelined Vedic multiplication for ECC based remote keyless system authentication.

5.1 Remote keyless system authentication

The ECC based remote keyless entity authentication system has various processes to be carried out between the key fob and the car. In the key fob, secret key generation, public key calculation and nonce decryption

processes are performed. In the car module, random nonce encryption and verification of decrypted nonce received from the key fob takes place. The overall diagram describing the process is given in Figure 3.

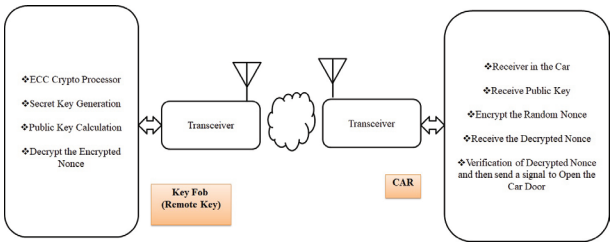


Figure 3: Block Diagram of Remote keyless system authentication Process

The above processes are accomplished in three stages as specified in Figure 4. They are described as follows,

Stage 1: The authentication process starts when the car key fob is pressed.

In Key Fob
In the first stage, when the user wishes to unlock the car door, he initiates the key fob. Two 256-bit random numbers are generated in the key fob with the help of the Linear Feedback Shift Register (LFSR). Among those, one is the Public key and another is the Private Key. The Private Key is kept confidential by the key fob. The 256-bit Public key would be transmitted to the Car through the transceiver module. Both the Public and Private keys are known to key fob alone.

In the Car
The Car receives the Public key and then, it generates a 256-bit random text (cipher text or plain text) using Linear Feedback Shift Register. Thereafter, plain text encryption takes place with the Public key. The sequence of plain text generation process is same for one complete process.

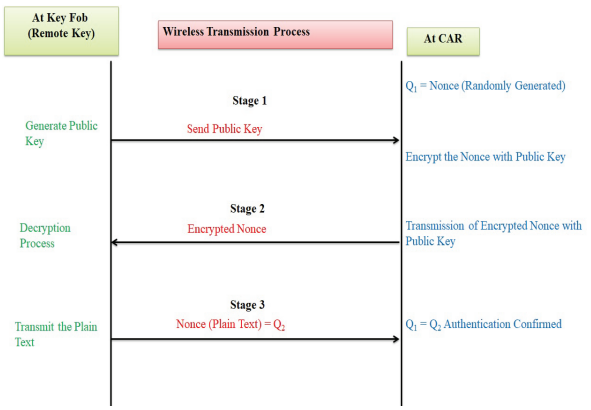


Figure 4: Three stages of Communication between Key Fob and Car

Stage 2: Encrypted Nonce.

In the Car
The Car will transmit the encrypted nonce with the plain text through the Transceiver.

In Key Fob
The key fob receives the encrypted 256-bit randomly generated plain text nonce. Then the Key fob performs the decryption of the nonce using the 256-bit Private Key.

Stage 3: Nonce Transmission.

In Key Fob
After decryption, the key fob transmits the nonce to the Car.

In the Car
The Car compares the received nonce with the original nonce generated at STAGE 1. If they are matched, then the Car door will be unlocked, otherwise not.

5.2 ECC

The ECC core chooses a point on the ECC curve in Koblitz coordinates *P(X, Y, Z)* and finds the point *Q(X, Y, Z)=k × P*. The system controller releases necessary control signals to produce the desired output *Q* as shown in Figure 5.

- Private key: nA, where nA is a 256-bit Random number.
- Public key: PA=nA × G, where nA is the Private key generated by the user (Key fob) and G is the point on the Elliptic Curve.
- Encryption: Cm = {kG, M+K PA} (cipher text), where K is the random integer chosen at the beginning and M is the Mapped point on the Elliptic curve.
- Message=(M+(K × PA)) - (K × G × nA), Because PA=nA × G, M(plaintext)

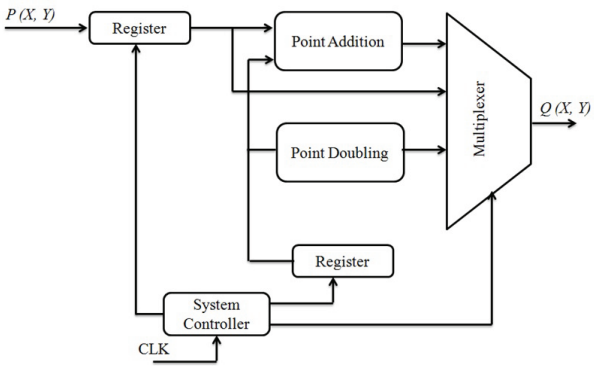


Figure 5: Overall ECC Processor architecture

Where M is the message point corresponding to the message. The Encryption operation generates a pair of points {C1, C2}.

Point addition is a computation method used to add two different points over a finite field. Here, λ is the slope of the two points. Its computation is different for point addition and point doubling as shown in equations (1) and (2). Subtraction is performed employing two's complement addition.

$$\lambda = \frac{Y_2 - Y_1}{X_2 - X_1} \tag{1}$$

$$\lambda = \frac{3x_1^2 + a}{2Y_1} \tag{2}$$

Point doubling is performed using multiplication architecture with both inputs the same. Here also division is necessary to compute point doubling. The point addition has been performed with equation (3) and the hardware architecture is shown in Figure 6. The multiplications are performed with shift and add method with modulus operation.

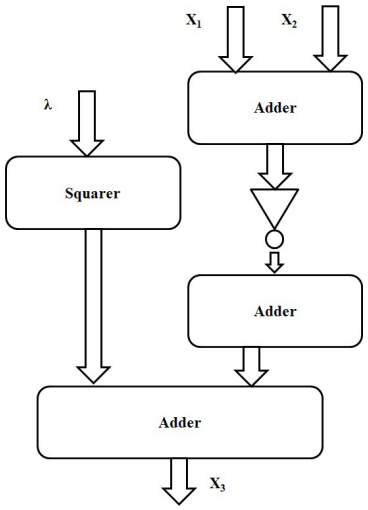


Figure 6a: Architecture for Calculating X_3

$$\begin{aligned} X_3 &= \lambda^2 - X_1 - X_2 \\ Y_3 &= (X_1 - X_3)\lambda - Y_1 \end{aligned} \tag{3}$$

Point Multiplication

Point multiplication [6] is the operation that multiplies a point with an integer. Montgomery ladder technique is used to perform point addition and point doubling in parallel. In this algorithm, two registers are used to store the intermediate results. Initially, one register loaded with an input point and another loaded with doubling

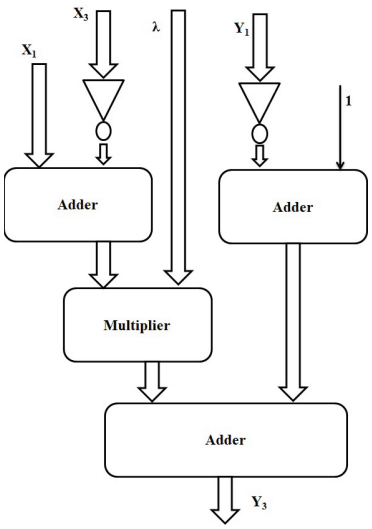


Figure 6b: Architecture for Calculating Y_3

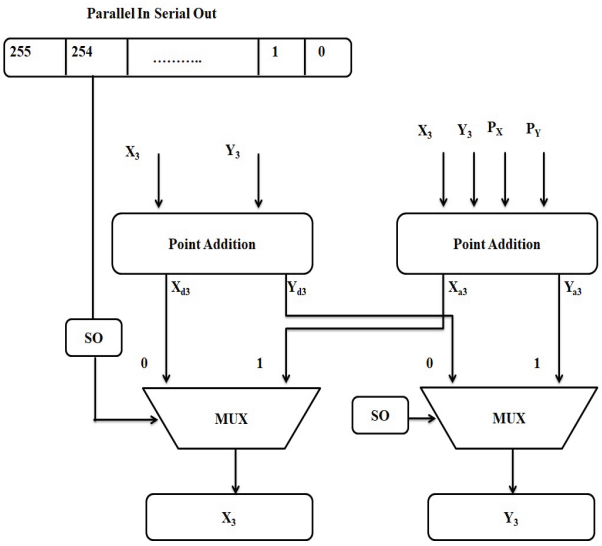


Figure 7: Architecture for Point Multiplication

of the input point. Then a loop is needed to run this algorithm. So, the serial shift register is used during every iteration. The sequence of iterations should be from n-2 to 0. The integer value which needs to be multiplied with the point is loaded into the shift register. Once it gets loaded, shifting should start from (n-2)-th bit to the 0^{th} bit. The shifted bit decides which operation will be performed and what content will be loaded into the registers. If it is 1, the first register is loaded with point addition result and the second register is loaded with point doubling result. If it is 0, then the second register loaded with point addition result and the first register is loaded with point doubling result. Once all bits get shifted out, the multiplication of the point with the integer is stored in the first register.

Point Multiplication Algorithm:

```
Q1 ←P; Q2 ←2P;
for i from n−2 down to 0
    do
    if ki =1 then
    Q1 ←Q1+Q2;     // point addition
    Q2 ←2Q2;     // point doubling
    else
    Q2 ←Q1+Q2;     // point addition
    Q1 ←2Q1;     // point doubling
end if;
end for;
return Q1;
```

Modular Multiplication Algorithm:

```
Formula : C =(A·B) mod p ;
    C ←0;
    T ←B&'1';
        while T(n−1 downto 0) != 0 loop
        C ←2C;
    If Tn =1 then              //nth bit of T
        C ←C +A;
    end if;
        C ←C mod p;
        T←T(n−1downto0)&'0';  //left-shift operation
    end loop;
    return C;
```

The above algorithm is implemented as shown in Figure 8. In order to perform the multiplication of two integers, left shift and adder are used in this architecture. In this method, a multiplier loaded with the shift register and multiplicand is given to the adder circuit as shown in Figure 8. The shift-left register is used to perform a synthesizable loop operation for the left to right bit-wise multiplication.

To determine the end of the loop, a (n+1)-bit temporary variable T is used in which T (n down to 1) is pre-computed as the multiplier B and the least significant bit (LSB) of T is pre-computed as 1. One extra bit is added at the LSB to cope with the completion of the left-shift operation in the case of $b_0 = 0$. The multiplicand A is added to the accumulator in each iteration if the most significant bit (MSB) of T is 1. The content of the accumulator is then reduced to *modulo p* after every addition. In order to perform this modular operation, C is subtracted by the prime numbers *p* and *2p*. As the content of the accumulator is always less than *3p*, subtractions by *p* and *2p* are enough to confine the content below the value of *p*. The subtractions C – p and C –2p are performed by adding the 2's complement of the subtrahends *p* and *2p* to the minuend C.

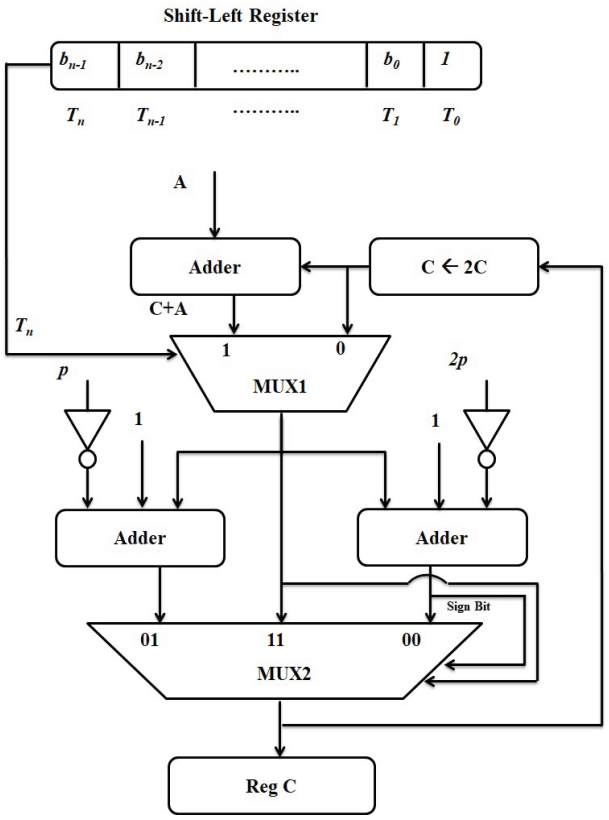


Figure 8: Architecture for Modular Multiplication

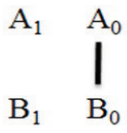
The comparisons C ≥ *p* and C ≥ *2p* are performed by checking the sign bits of the differences C–*p* and C –*2p*, respectively. At the end of every iteration, T is shifted to the left by one bit. After performing 'n' iterations, T(n–1 down to 0) is shifted to zero value and the content of the accumulator is stored in register 'Reg C', which is the final modular product of integers A and B. The module comprises two multiplexers, in which MUX1 is used to keep the content of the accumulator unchanged if Tn = 0; or add A to the accumulator if Tn = 1; and MUX2 is used for performing C mod *p*. At $(n+1)^{th}$ clock cycle, the result for multiplication of two inputs is available.

*5.3 2*2 Vedic multiplier*

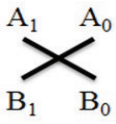
Considering two-bit numbers A (A_1A_0) and B (B_1B_0), the 2×2 Vedic multiplication is carried out as depicted in Figure 9. The logical expression of the final product is as shown in equation 4,

$$P_0 = A_0.B_0 \qquad P_1 = (A_1.B_0) \oplus (A_0.B_1)$$
$$P_2 = (A_0.A_1.B_0.B_1) \oplus (A_1.B_1) \qquad P_3 = A_0.A_1.B_0\,B_1 \tag{4}$$

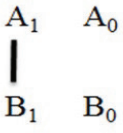
This process consumes 4 AND gates and 2 EXOR gates.



Step 1: $P_0=A_0.B_0$



Step 2: $P_1=(A_1.B_0) \oplus (A_0.B_1)$



Step 3:
$P_2=(A_0.A_1.B_0.B_1) \oplus (A_1.B_1)$; $P_3=A_0.A_1.B_0.B_1$

Figure 9: Graphical representation of 2*2 Vedic multiplication steps

*4*4 Vedic multiplier*

The 4-bit Vedic multiplier comprises four 2-bit Vedic multipliers, three 4-bit full adders & one half adder gate. The two 4-bit inputs A_i ($A_3A_2A_1A_0$) and B_i ($B_3B_2B_1B_0$) are applied to the 2-bit Vedic multiplier, and then they are forwarded to the 4-bit adder. The output from the RCA adder consists of 4-bit sum output and a 1-bit carry value. The half adder is used to sum the carry at the first two phases of the ripple carry adder. The output of the 4-bit multiplier consists of an 8-bit product term

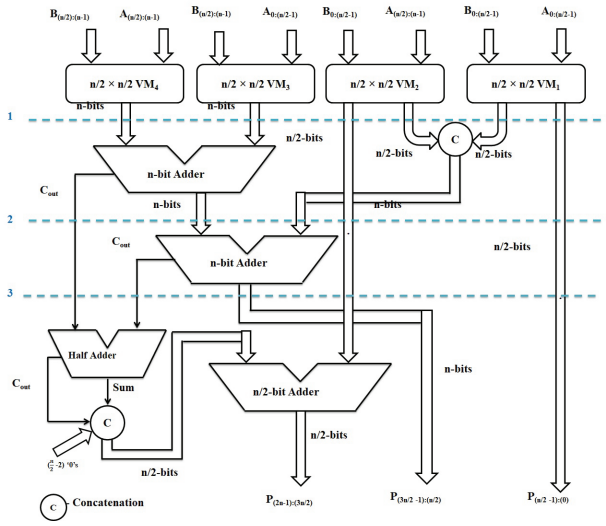


Figure 10: Generalized construction of n-bit Vedic Multiplier

(P_i - P_7…P_0) [25]. Here, the pipeline technique is introduced at 3 levels mentioned in Figure 10 (numbered 1, 2 and 3) by means of inserting registers. Similarly, higher order Vedic multipliers are constructed with the 2-bit VM_i as the base module and RCA adders for summation of the partial products. The n-bit Vedic multiplier uses four n/2-bit multipliers, two n-bit RCA adders, one n/2-bit adder, and one Half adder as shown in Figure 10.

6 Results and discussion

6.1 FPGA implementation and analysis

This section presents the FPGA implementation for the proposed ECC processor architecture. The necessary parameters such as curve order, coefficients and base point coordinates are selected based on the NIST standard. Here, we have considered a 256-bit ECC processor. The ECC processor is designed using Verilog HDL and simulated using ModelSim. It was synthesized, placed and routed using Xilinx ISE 14.6. In the proposed methodology, the used FPGA platforms were Virtex- 6 (XC6VLX240T-1FF1156) and Virtex-7 (XC7VX485T-2FFG1761C) with the goal to achieve optimal speed and area. The implementation results of the proposed 256-bit ECC module are summarized in Table 2. In which, the ECC is implemented with or without pipeline in Vedic multiplier.

The performance factors throughput and efficiency are calculated based on equation (5) [6, 22, 26, 27].

$$\left.\begin{aligned} &\textit{Cycle} = \textit{Time for one ECPM} \times \textit{maximum frequency} \\ &\textit{AT/Bit} = \textit{Area-Delay product} / \textit{Number of Bits} \\ &\textit{Throughput} = (\textit{maximum frequency} \times \\ &\textit{number of bits}) / \textit{number of clock cycles} \\ &\textit{Efficiency} = \textit{Throughput} / \textit{area} \end{aligned}\right\} \quad (5)$$

The Simulation results of point addition, point doubling, public key calculation, data encryption and decryption are shown in Figure 11 (a-e). In Figure 11a, the input points are (a,b) (c,d) and the resultant point addition is available in (x_3, y_3). It is produced in 945 clock cycles. Figure 11b shows the simulation result of point doubling, where (x_p,y_p) are input point and the result (x_3, y_3) is produced in 430 clock cycles. Figure 11c shows the simulation result of public key generation. The complete ECPM simulation is shown in Figure 11d, which is completed in 232.21k clock cycles. When the interim states are having less than 256-bits, then the ECPM gets completed in fewer cycles.

The pipelined Vedic multiplier based ECC implemented on Virtex-7 occupies 8.23k slices, takes 32.2k clock cy-

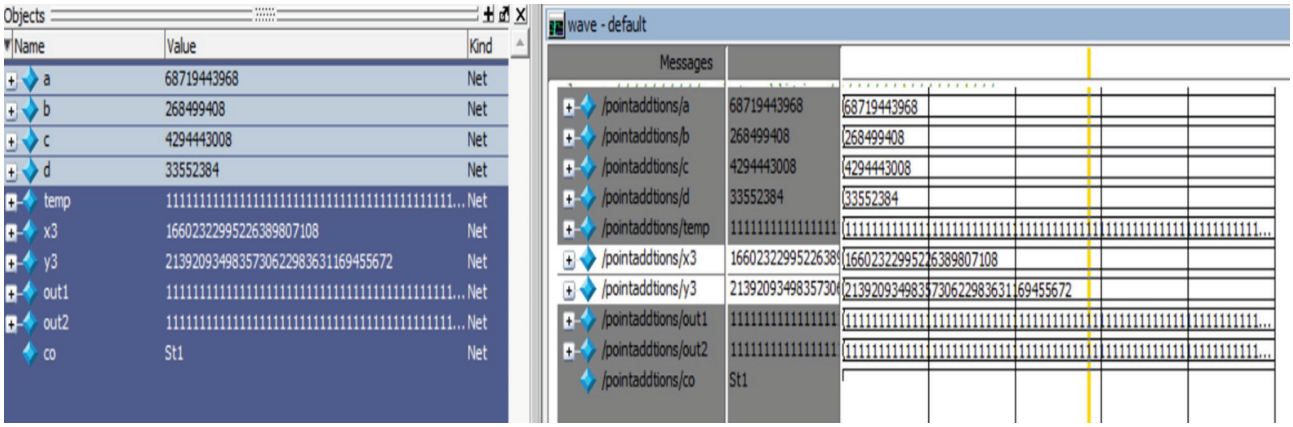


Figure 11a: Simulation Result of Point Addition

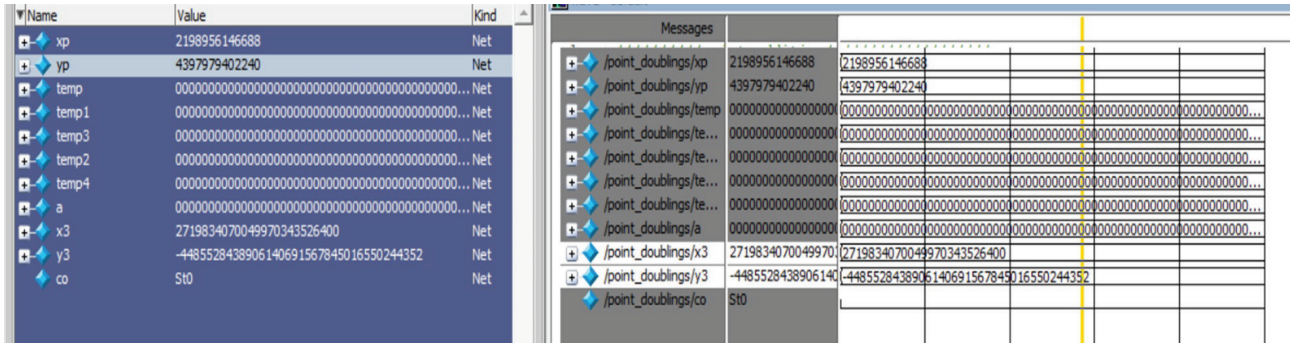


Figure 11b: Simulation Result of Point Doubling

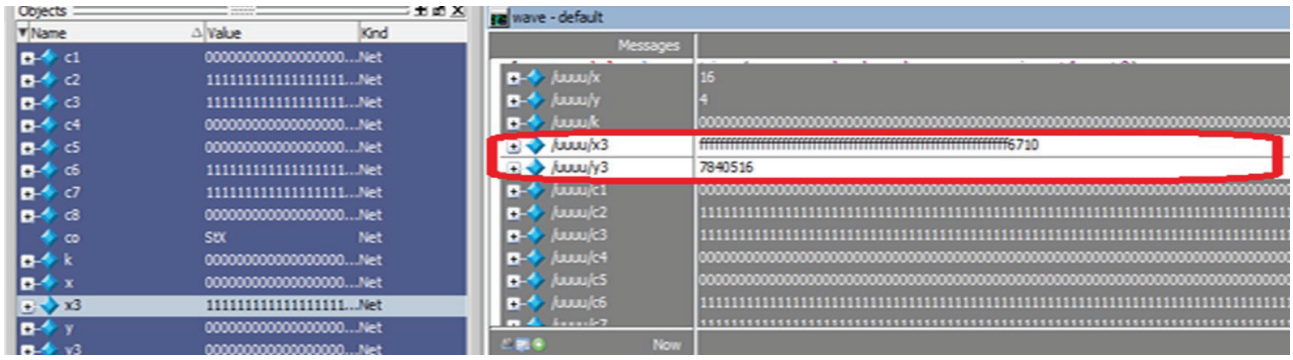


Figure 11c: Simulation Result of Public Key Calculation

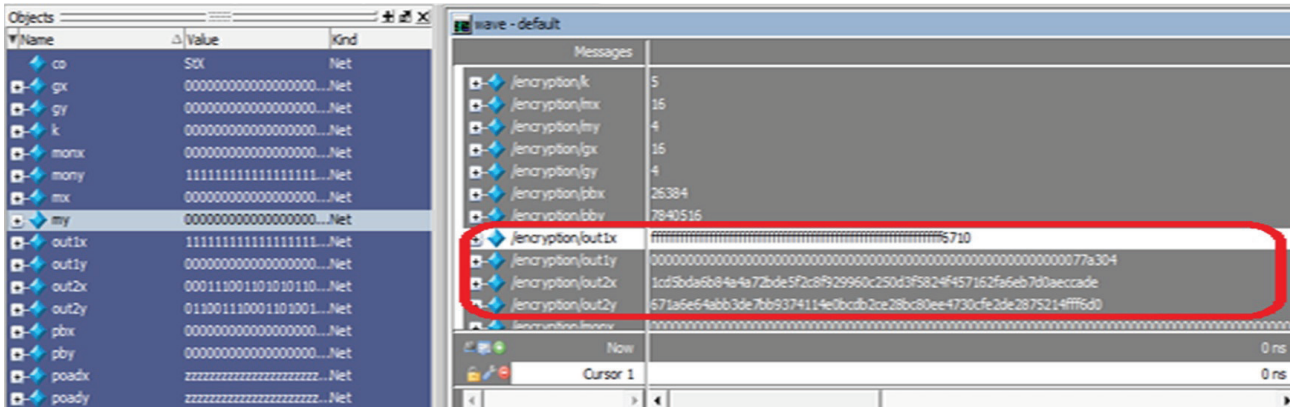


Figure 11d: Simulation Result of Encryption

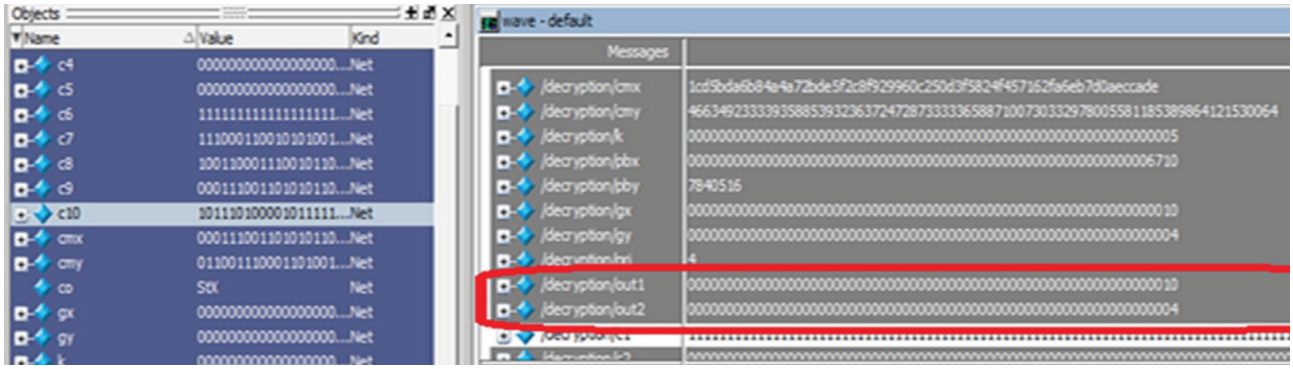


Figure 11e: Simulation Result of Decryption

cles for process completion, the maximum operating frequency is 192.5MHz, and the process takes 0.9365ms to complete. The Area-Delay product and throughput are 7.922 and 273.36kbps, respectively. The pipelined ECC has 32.31% efficiency. The pipelined Vedic ECC has 22.36% and 20.18% improvement in throughput and efficiency with respect to non-pipelined implementation. This is due to the increased frequency of operation (226.8MHz) of pipelined architecture. It leads to additional 2.72% overhead on the area; but the area-delay overhead is reduced by 20.19%.

Table 2: Performance of Proposed Vedic multiplier based ECC

	Platform	Number of bits	Number of Slices (k)	Clock Cycles (k)	Maximum Frequency (MHz)	Time (ms)	Area Delay Product	AT/B	Throughput (kbps)	Efficiency
No Pipeline	Virtex-7	256	8.23	232.2	192.5	1.2062	9.927026	0.0387774	212.23	25.79
Pipeline			8.46	212.4	226.8	0.9365	7.92279	0.0309484	273.36	32.31
No Pipeline	Virtex-6		8.82	232.2	186.2	1.247	10.99854	0.042963	205.29	23.27
Pipeline			9.12	212.4	216.8	0.98	8.9376	0.0349125	261.3	28.65
Percentage of improvement	Virtex-7		2.72	8.53	15.12	22.36	20.19	20.19	22.36	20.18
	Virtex-6		3.29	8.53	14.11	21.41	18.74	18.74	21.44	18.78

Table 3: Performance comparison of Proposed ECC

Ref	Year	Platform	Number of bits	Number of Slices (k)	Clock Cycles (k)	Maximum Frequency (MHz)	Time (ms)	Area Delay Product	AT/B	Throughput (kbps)	Efficiency
Ours	woP	Virtex-7	256	8.23	232.2	192.5	1.2062	9.927026	0.03878	212.23	25.79
	wP			8.46	212.4	226.8	0.9365	7.92279	0.03095	273.36	32.31
	woP	Virtex-6		8.82	232.2	186.2	1.247	10.99854	0.04296	205.29	23.27
	wP			9.12	212.4	216.8	0.98	8.9376	0.03491	261.30	28.65
26	2020	Virtex-7	256	6.5	198.7	104.39	1.9	12.35	0.0482	134.49	20.69
26	2020	Virtex-6	256	6.6	198.7	93.23	2.13	14.05	0.0549	120.12	18.20
28	2019	Virtex-7	256	8.9	262.7	177.7	1.48	13.17	0.051	173.2	19.46
28	2019	Virtex-6	256	9.2	262.7	161.1	1.63	15.00	0.059	157.00	17.06
29	2018	Virtex-6	256	65.6	153.2	327	0.47	30.83	0.120	546.42	8.33
30	2018	Virtex-4	256	9.4 + 14DSPs	610	20.44	29.84	280.5	1.096	8.58	0.91
31	2017	Kintex-7	256	11.3	397.3	121.5	3.27	63.95	0.144	78.28	6.92
32	2017	Virtex-7	256	24.2	215.9	72.9	2.96	71.63	0.280	1816.20	3.57
33	2017	Virtex-4	193	12	459.9	36.5	12.6	151.2	0.783	20.32	1.69
34	2017	Virtex-4	256	20.6	191.6	49	3.91	80.55	0.315	65.47	3.18
27	2016	Virtex-4	256	13.2	200	40	5	66	0.258	51	3.88
35	2016	Virtex-4	192	35.7	207.1	70	2.96	105.67	0.550	86.53	2.42
36	2016	Virtex-5	256	8.7	361.6	160	2.26	19.66	0.077	113.27	13.02

woP – Without Pipeline; wP – with Pipeline; Þ-calculated by author

The same pipelined architecture has 261.30kbps throughput and 28.65 efficiency in Virtex 6. The pipeline structure has 21.44% and 18.78% improvement in throughput and efficiency with respect to non-pipelined implementation with 3.29% area overhead. Significantly, the maximum frequency of operation and time consumption are improved by 14.11% and 21.41% respectively.

The performance characteristics of ECC implementation in FPGA are shown in Table 3. Here, varieties of FPGA families such as Virtex-4, 5, 6, 7 and Kindtex-7 are used for implementation purposes. Most of the researchers designed for the 256-bit size ECC, except in [33, 35], where 192 and 193-bit are considered respectively. The significant performance factors considered for the analysis are Area, the number of clock cycles, maximum operating frequency, area-delay product, throughput and efficiency.

From Table 3, it is observed that higher frequency of operation leads to a reduction in the required number of clock cycles, completion time and increases the throughput. The proposed pipelined Vedic multiplier has an optimized area, speed and throughput. The ECC based security systems are performing well against the attacks [37] such as algebraic attack, brute force attack and statistical attack [38, 39] and protects confidential data hiding in spatial images [40].

In [6], efficiency is calculated using throughput and area, which is represented in the equation (5). The same has been calculated and shown for comparison in Table 3. The efficiency of the proposed method is comparatively better with respect to all the previous works. Moreover, the pipelining of the multiplication process increases the efficiency by another 25%. The earlier FPGA studies using Virtex-I Pro, Virtex-E and Spartan 4 are omitted for comparison in Table 3 due to their nature of high power consumption and the limited number of Input/Outputs.

6.2 Security Evaluation and Analysis

The security evaluation criteria which are essential for the Remote Keyless Entry system are shown in Table 4. In order to illustrate the effectiveness of the proposed ECC scheme evaluation, a comparative assessment of 10 schemes for the RKE system has been done. By evaluating the 10 criteria of security attacks stated in [41, 42, 51] the performance of the proposed Vedic based ECC has been evaluated. The results are summarized in Table 5.

Table 4: Security Evaluation criteria

Short Form	Evaluation Criteria
C1	No password verifier-table
C2	Resist password guessing threat
C3	Defend replay attack
C4	Defend session key temporary information attack
C5	Accurate login and password change phase
C6	Defend user un-traceability attack
C7	Mutual authentication
C8	Facilitates user anonymity
C9	Defend insider attack
C10	Facilitates forward secrecy property

Table 5: Security comparison among the authentication schemes

Ref.	Year	Evaluation Criteria									
		C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Ours	2022	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
42	2022	✓	✓	✓	✓	×	✓	✓	✓	✓	✓
43	2020	✓	✓	✓	✓	×	×	✓	✓	✓	✓
44	2019	✓	✓	✓	✓	×	×	✓	✓	✓	✓
45	2019	✓	✓	✓	✓	×	✓	✓	✓	×	×
46	2018	✓	✓	✓	✓	×	×	✓	✓	×	✓
51	2018	×	✓	×	✓	✓	✓	✓	✓	✓	✓
47	2018	✓	✓	✓	✓	×	×	✓	✓	×	✓
48	2018	✓	×	✓	✓	×	×	✓	✓	×	✓
49	2018	✓	✓	✓	✓	×	×	✓	✓	×	×
50	2017	✓	✓	✓	✓	×	×	✓	✓	×	×

Table 6: Evaluation of computational cost

Ref.	Year	Authentication		Time Period (Sec)		
		User	Car Sensor	User	Car Sensor	Total
Ours	2022	$T_{be} + 2T_H$	$T_S + 2T_H$	0.0600	0.5733	0.6333
42	2022	$T_{be} + 3T_H$	$T_{se} + 4T_H + T_S$	0.0605	0.583	0.6435
43	2020	$T_{be} + 3T_H$	$T_{se} + TS + 4T_H$	0.0605	0.583	0.6435
44	2019	$T_{PM} + 3T_H$	$2T_{PM} + 4T_H$	1.0518	2.0523	3.1041
45	2019	$5T_H$	$5T_H$	0.0528	0.0528	0.1056
46	2018	$T_C + 3T_H$	$2T_C + 6T_H$	0.5738	1.0973	1.6711
51	2018	$5T_H$	$11T_H$	0.0528	0.0558	0.1086
47	2018	$T_{PM} + 2T_H$	$2T_{PM} + 4T_H$	1.0513	2.0523	3.1036
48	2018	$6T_H$	$5T_H$	0.0533	0.0528	0.1061
49	2018	$8T_H$	$6T_H$	0.0543	0.0533	0.1076
50	2017	$5T_H + T_S$	$7T_H + T_S$	3.0615	0.5758	3.6373

T_H: time complexity of a hash function; T_{PM}: the time complexity of ECC point multiplication operation; T_S: time complexity of a symmetric key encryption/decryption operation; T_{ME}: time complexity of a modular exponentiation operation

In order to evaluate the execution time of the proposed protocol and relevant protocols, we have assumed that the hash function, modular exponentiation operation, a symmetric key encryption/decryption operation and point multiplication operation require 0.0005 seconds, 0.522 seconds, 0.0087 seconds and 0.0503 seconds [51], respectively. The computation cost for the RKE is estimated and compared with the existing literature in Table 6. It is observed that the proposed method is competent in computation cost with the previously published works. The proposed RKE has 1.58% improvement compared to the recently available method [42]. Hence, the proposed pipelined vedic ECC can be incorporated in RKE for effective secured communication of various applications such as smart cards [52], mobile devices [53] and wireless sensor networks [42].

7 Conclusion

In this paper, a high-speed, area-efficient ECC processor is designed on the Koblitz curve secp256k1 for the Remote Keyless Authentication system. It supports 256-bit point addition and point multiplication over a prime field. A novel method of multiplication using Urdhva Tiryagbhyam is adopted for scalar multiplication. The speed of multiplication is improved by incorporating the pipeline technique. The proposed pipelined Vedic multiplier based ECC processor is implemented in the Xilinx Virtex-7 and Virtex-6 platforms for the 256-bit prime field. The implemented processor performs a single 256-bit multiplication in 1.2062ms with a maximum clock frequency of 192.5MHz, which provides 212.23kbps throughput and occupies 8.23k slices in Virtex-7 FPGA. Incorporating pipeline in scalar multiplication improves the maximum clock frequency up to 15.12%, and reduces time by 22.36%, which in turn increases the throughput by 22.36%. The pipeline has an additional area overhead of 2.72% and 3.29% in Virtex-7 and Virtex-6 respectively. Also, the computational cost of the proposed method is evaluated, which shows 1.58% improvement from the most recent literature. The pipelined Vedic multiplier based ECC processor outperforms the existing designs in terms of area, clock cycle count, operating frequency, time, area-delay product, throughput, efficiency and security. Based on the overall performance of the proposed ECC processor, it can be concluded that it is a reliable choice for wireless communication technologies as well as for resource constrained applications.

8 Conflicts of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

9 References

1. Kunal Karnik, Saurabh Kale, Manandeep, Ajinkya Medhekar, 2020. On Vehicular Security for RKE and Cryptographic Algorithms: A Survey, https://www.ijert.org/research/on-vehicular-security-for-rke-and-cryptographic-algorithms-a-survey-IJERTV9IS050693.pdf
2. Kraft, Caleb. Anatomy of the RollJam Wireless Car Hack. Make: We Are All Makers. 10/11/15. Accessed 12/10/15. Available from: http://makezine.com/2015/08/11/anatomy-of-the-rolljam-wireless-carhack/
3. Melnikov, D.A., Lavrukhin, Y.N., Durakovsky, A.P., Gorbatov, V.S. and Petrov, V.R., 2015, August. Access control mechanism based on entity authentication with IPv6 header 'flow label' field. In 2015, 3rd International Conference on Future Internet of Things and Cloud (pp. 158-164). IEEE, https://doi.org/10.1109/FiCloud.2015.41.
4. Kuhn, D.R., Hu, V.C., Polk, W.T. and Chang, S.J., 2001. Introduction to Public Key Technology and the Federal PKI Infrastructure Infrastructure, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-32.pdf.
5. Amara, M. and Siad, A., 2011, May. Elliptic curve cryptography and its applications. In International workshop on systems, signal processing and their applications, WOSSPA, pp. 247-250. IEEE, https://doi.org/10.1109/WOSSPA.2011.5931464.
6. Mahdizadeh, H. and Masoumi, M., 2013. Novel Architecture for Efficient FPGA Implementation of Elliptic Curve Cryptographic Processor Over ${\rm GF}(2^{163}) $. IEEE transactions on very large scale integration (VLSI) systems, 21(12), pp.2330-2333, https://doi.org/10.1109/TVLSI.2012.2230410.
7. Rakshith TR., Rakshith Saligram, Design of High Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach, International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013], 2013, pp.775-781, https://doi.org/10.1109/ICCPCT.2013.6528848.
8. Kamaraj, A. and Marichamy, P., 2019. Design of fault-tolerant reversible Vedic multiplier in quantum cellular automata. Journal of the National Science Foundation of Sri Lanka, 47(4), pp.371-382, https://doi.org/10.4038/jnsfsr.v47i4.9677.

9. Francillon, Aurélien, Boris Danev, and Srdjan Capkun. Relay attacks on passive keyless entry and start systems in modern cars. In Proceedings of the Network and Distributed System Security Symposium (NDSS). Eidgenössische Technische Hochschule Zürich, Department of Computer Science, 2011, https://eprint.iacr.org/2010/332.pdf.
10. Glocker, Tobias, Timo Mantere, and Mohammed Elmusrati. A protocol for a secure remote keyless entry system applicable in vehicles using symmetric-key cryptography. In 2017 8th International Conference on Information and Communication Systems (ICICS), pp. 310-315. IEEE, 2017, https://doi.org/10.1109/IACS.2017.7921990.
11. https://www.dailymail.co.uk/sciencetech/article-3187299/The-30-universal-remote-cars-hackers-access-vehicle-open-garage-door.html
12. https://auto.economictimes.indiatimes.com/news/auto-technology/hacker-creates-new-device-that-can-unlock-any-luxury-car/74151233
13. https://www.aisec.fraunhofer.de/en/fields-of-expertise/projects/ecc-rke.html
14. Ni, X., Shi, W. and Fook, V.F.S., 2007, April. AES security protocol implementation for automobile remote keyless system. In 2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring, pp. 2526-2529. IEEE, https://doi.org/10.1109/VETECS.2007.520.
15. Siddiqui, A.S., Gui, Y., Plusquellic, J. and Saqib, F., 2017. A secure communication framework for ECUs. Advances in Science, Technology and Engineering Systems Journal, 2(3), pp.1307-1313, https://doi.org/10.25046/aj0203165.
16. Shafiq, A., Altaf, I., Mahmood, K., Kumari, S. and Chen, C.M., 2020. An ECC based remote user authentication protocol. Journal of Internet Technology, 21(1), pp.285-294, https://jit.ndhu.edu.tw/article/view/2243.
17. Luo, H., Zhang, Q. and Xu, G., 2021, May. Privacy-preserving ECC-based three-factor authentication protocol for smart remote vehicle control system. In EAI International Conference on Applied Cryptography in Computer and Communications, pp. 56-72. Springer, Cham, https://doi.org/10.1007/978-3-030-80851-8_5.
18. https://sectigo.com/resource-library/why-automotive-key-fob-encryption-hacks-are-making-headlines
19. Karthikeyan, S. and Jagadeeswari, M., 2021. Performance improvement of elliptic curve cryptography system using low power, high speed 16× 16 Vedic multiplier based on reversible logic. Journal of Ambient Intelligence and Humanized Computing, 12(3), pp.4161-4170, https://doi.org/10.1007/s12652-020-01795-5.
20. Kumar, A. and Sharma, V., 2017. Comparative analysis of Vedic & array multiplier. International Journal of Electronics and Communication Engineering and Technology, ISSN: 0976-6464 (P), 0976-6472, 8(3), pp.17-27, https://ejaet.com/PDF/4-7/EJAET-4-7-524-531.pdf.
21. Sunitha, G.S. and Rakesh, H.M., 2018. Performance Comparison of Conventional Multiplier with Vedic Multiplier using ISE Simulator, International Journal of Engineering and Manufacturing Science. 8(1), pp. 95-103, https://www.ripublication.com/ijems_spl/ijemsv8n1_10.pdf.
22. Kodali, R.K., Yenamachintala, S.S. and Boppana, L., 2014, September. FPGA implementation of 160-bit Vedic multiplier. In 2014 International Conference on Devices, Circuits and Communications (ICDCCom) (pp. 1-5). IEEE, https://doi.org/10.1109/ICDCCom.2014.7024721.
23. Ahuja, P., Soni, H. and Bhavsar, K., 2018, March. Fast, Secure and Efficient Vedic Approach for Cryptographic implementations on FPGA. In 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 1706-1710). IEEE, https://doi.org/10.1109/ICECA.2018.8474569.
24. Hankerson, D.; Menezes, A.J.; Vanstone, S. Guide to elliptic curve cryptography. Comput. Rev. 2005, 46, 13, https://doi.org/10.5555/1965110.
25. Gowthami P. and R. V. S. Satyanarayana, Performance Evaluation of Reversible Vedic Multiplier, ARPN Journal of Engineering and Applied Sciences, Vol. 13, No. 3, February 2018, pp.1002-1008, http://www.arpnjournals.org/jeas/research_papers/rp_2018/jeas_0218_6767.pdf.
26. Islam, M., Hossain, M., Hasan, M., Shahjalal, M. and Jang, Y.M., 2020. Design and implementation of high-performance ECC processor with unified point addition on twisted edwards curve. Sensors, 20(18), p.5148, https://doi.org/10.3390/s20185148.
27. Javeed, K. and Wang, X., 2017. Low latency flexible FPGA implementation of point multiplication on elliptic curves over GF (p). International Journal of Circuit Theory and Applications, 45(2), pp.214-228, https://doi.org/10.1002/cta.2295.
28. Islam, M. M., Hossain, M. S., Hasan, M. K., Shahjalal, M., & Jang, Y. M. (2019). FPGA implementation of high-speed area-efficient processor for elliptic curve point multiplication over prime field. IEEE Access, 7, 178811-178826, https://doi.org/10.1109/ACCESS.2019.2958491.
29. Shah, Y.A., Javeed, K., Azmat, S. and Wang, X., 2019. Redundant-signed-digit-based high speed elliptic curve cryptographic processor. Journal of Cir-

cuits, Systems and Computers, 28(05), p.1950081, https://doi.org/10.1142/S0218126619500816.
30. Hu, X., Zheng, X., Zhang, S., Cai, S. and Xiong, X., 2018. A low hardware consumption elliptic curve cryptographic architecture over GF (p) in embedded application. Electronics, 7(7), p.104, https://doi.org/10.3390/electronics7070104.
31. Hossain, M.S., Kong, Y., Saeedi, E. and Vayalil, N.C., 2017. High-performance elliptic curve cryptography processor over NIST prime fields. IET Computers & Digital Techniques, 11(1), pp.33-42, https://doi.org/10.1049/iet-cdt.2016.0033.
32. Asif, S., Hossain, M.S. and Kong, Y., 2017. High-throughput multi-key elliptic curve cryptosystem based on residue number system. IET Computers & Digital Techniques, 11(5), pp.165-172, https://doi.org/10.1049/iet-cdt.2016.0141.
33. Liu, Z., Liu, D. and Zou, X., 2016. An efficient and flexible hardware implementation of the dual-field elliptic curve cryptographic processor. IEEE Transactions on Industrial Electronics, 64(3), pp.2353-2362, https://doi.org/10.1109/TIE.2016.2625241.
34. Javeed, K., Wang, X. and Scott, M., 2017. High performance hardware support for elliptic curve cryptography over general prime field. Microprocessors and Microsystems, 51, pp.331-342, https://doi.org/10.1016/j.micpro.2016.12.005.
35. Javeed, K. and Wang, X., 2016. FPGA based high speed SPA resistant elliptic curve scalar multiplier architecture. International Journal of Reconfigurable Computing, 2016, https://doi.org/10.1155/2016/6371403.
36. Marzouqi, H., Al-Qutayri, M., Salah, K., Schinianakis, D. and Stouraitis, T., 2015. A high-speed FPGA implementation of an RSD-based ECC processor. IEEE Transactions on very large scale integration (vlsi) systems, 24(1), pp.151-164, https://doi.org/10.1109/TVLSI.2015.2391274.
37. G. Indumathi, S.Sathyakala, 2016, FPGA Based Elliptic Curve Cryptography for LAN Security, EIJO Journal of Engineering, Technology And Innovative Research, 1(2), pp. 09 – 18, https://www.eijo.in/asset/images/uploads/14651428477877.pdf.
38. Vigila, S.M.C. and Muneeswaran, K., 2012. Key generation based on elliptic curve over finite prime field. International Journal of Electronic Security and Digital Forensics, 4(1), pp.65-81, https://doi.org/10.1504/IJESDF.2012.045391.
39. Vigila, S.M.C. and Muneeswaran, K., 2013. A new elliptic curve cryptosystem for securing sensitive data applications. International Journal of Electronic Security and Digital Forensics, 5(1), pp.11-24, https://doi.org/10.1504/IJESDF.2013.054405.
40. Vigila, S.M.C. and Muneeswaran, K., 2015. Hiding of Confidential Data in Spatial Domain Images using Image Interpolation. Int. J. Netw. Secur., 17(6), pp.722-727, http://ijns.jalaxy.com.tw/contents/ijns-v17-n6/ijns-2015-v17-n6-p722-727.pdf.
41. Wang, D., Li, W. and Wang, P., 2018. Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks. IEEE Transactions on Industrial Informatics, 14(9), pp.4081-4092.
42. Wang, C., Wang, D., Xu, G. and He, D., 2022. Efficient privacy-preserving user authentication scheme with forward secrecy for industry 4.0. Science China Information Sciences, 65(1), pp.1-15.
43. Wang, D., Wang, P. and Wang, C., 2020. Efficient multi-factor user authentication protocol with forward secrecy for real-time data access in WSNs. ACM Transactions on Cyber-Physical Systems, 4(3), pp.1-26.
44. Li, X., Peng, J., Obaidat, M.S., Wu, F., Khan, M.K. and Chen, C., 2019. A secure three-factor user authentication protocol with forward secrecy for wireless medical sensor network systems. IEEE Systems Journal, 14(1), pp.39-50.
45. Ostad-Sharif, A., Arshad, H., Nikooghadam, M. and Abbasinezhad-Mood, D., 2019. Three party secure data transmission in IoT networks through design of a lightweight authenticated key agreement scheme. Future Generation Computer Systems, 100, pp.882-892.
46. Srinivas, J., Das, A.K., Wazid, M. and Kumar, N., 2018. Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial Internet of Things. IEEE Transactions on Dependable and Secure Computing, 17(6), pp.1133-1146.
47. Lin, C., He, D., Kumar, N., Choo, K.K.R., Vinel, A. and Huang, X., 2018. Security and privacy for the internet of drones: Challenges and solutions. IEEE Communications Magazine, 56(1), pp.64-69.
48. Wu, F., Li, X., Sangaiah, A.K., Xu, L., Kumari, S., Wu, L. and Shen, J., 2018. A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks. Future Generation Computer Systems, 82, pp.727-737.
49. Amin, R., Islam, S.H., Biswas, G.P., Khan, M.K. and Kumar, N., 2018. A robust and anonymous patient monitoring system using wireless medical sensor networks. Future Generation Computer Systems, 80, pp.483-495.
50. Wazid, M., Das, A.K., Odelu, V., Kumar, N. and Susilo, W., 2017. Secure remote user authenticated key establishment protocol for smart home environment. IEEE Transactions on Dependable and Secure Computing, 17(2), pp.391-406.

51. Chandrakar, P. and Om, H., 2018. An efficient two-factor remote user authentication and session key agreement scheme using rabin cryptosystem. Arabian Journal for Science and Engineering, 43(2), pp.661-673.
52. Li, Z., Wang, D. and Morais, E., 2020. Quantum-safe round-optimal password authentication for mobile devices. IEEE Transactions on Dependable and Secure Computing.
53. He, D., Wang, D. and Wu, S., 2013. Cryptanalysis and improvement of a password-based remote user authentication scheme without smart cards. Information technology and control, 42(2), pp.105-112.

Arrived: 04. 01. 2022
Accepted: 16. 05. 2022