# THE ELLIPTIC CURVE CRYPTOGRAPHY OVER PRIME FIELDS ($F_P$) USING VEDIC MATHEMATICS AND PROJECTIVE COORDINATE SYSTEM

Sandeep Kumar Tyagi
Assistant professor, Department of Mathematics, Motherhood University, Roorkee, Uttarakhand, India, 247667
**Email:** foet.sandeeptyagi@motherhooduniversity.edu.in

**Abstract:** The Elliptic Curve Cryptography (ECC) over prime fields $F_p$ is widely adopted in modern cryptographic systems due to its high security with relatively smaller key sizes. However, the computational complexity of point addition and point doubling operations remains a critical performance bottleneck, particularly in resource-constrained environments. In this paper, techniques derived from Ancient Indian Vedic Mathematics (AIVM) are employed to accelerate the arithmetic operations involved in ECC implemented using the projective coordinate system. The proposed approach integrates Vedic mathematical algorithms into elliptic curve point addition and point doubling processes, enabling efficient parallel computation and reduced computational overhead. An improved parallel elliptic curve algorithm is designed and implemented using projective coordinates to eliminate costly inversion operations. Performance analysis demonstrates that the proposed AIVM-based design achieves superior speed and computational efficiency compared to conventional ECC implementations and existing projective coordinate-based methods. The experimental results indicate that the proposed scheme offers enhanced performance with reduced computational cost, making it a promising and attractive direction for high-speed and low-resource cryptographic applications.

[Tyagi, S.K. **THE ELLIPTIC CURVE CRYPTOGRAPHY OVER PRIME FIELDS ($F_P$) USING VEDIC MATHEMATICS AND PROJECTIVE COORDINATE SYSTEM**. *The International Journal of Interpretation, Observation and Analysis*, 2026; Volume 1, Issue 2 (January-March): Page Number: 51-70. **ISSN 2349-0713, Peer-reviewed (online/offline), Refereed, Indexed and International Journal (Since 2013), Global Impact Factor: 6.489. (INTERNATIONAL CONFERENCE ON EDUCATIONAL RESEARCH & INNOVATION, Organised by Mudo Tamo Memorial College, Ziro (Arunachal Pradesh)** Collaboration with COSMOSSKY HORIZON (OPC) PRIVATE LIMITED, Date: 8th February, 2026, Mode: Online/Offline (Hybrid).

**Key Words:** Elliptic Curve, Cryptography, Projective Coordinate System, Urdhva Tiryagbhyam multiplication, Dvandva Yoga technique, Point Addition, Point Doubling

**Mathematics Subject Classification:** 94A60, 14G50.

## INTRODUCTION

Elliptic Curve Cryptography (ECC) has emerged as one of the most efficient and secure public key cryptographic techniques due to its ability to provide strong security with significantly smaller key sizes. The security of ECC is based on the computational hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which is considered infeasible to solve for appropriately chosen curves and parameters. Because of its low computational requirements, reduced memory usage, and high security per bit, ECC has become highly suitable for modern cryptographic applications, particularly in resource-constrained environments such as smart cards, mobile devices, wireless sensor networks, and Internet-of-Things (IoT) platforms. Elliptic curves were first introduced into public key cryptography independently by Miller in 1986 and Koblitz in 1987. Since then, extensive research has been carried out to improve the computational efficiency of ECC implementations. Compared to traditional public key cryptosystems such as RSA and integer-based discrete logarithm systems, ECC achieves equivalent security with much smaller key sizes. For instance, a 160-bit ECC system is widely believed to offer a security level comparable to that of a 1024-bit RSA system. Moreover, the rate at which ECC key sizes must increase to maintain security is considerably slower than that of RSA or classical discrete logarithm schemes, making ECC a scalable and future-proof cryptographic solution. At the core of ECC operations lie elliptic curve point addition and point doubling, which together form the basis of scalar multiplication, the fundamental operation used in key generation, encryption, decryption, and digital signatures. Although ECC offers strong theoretical advantages, the practical performance of ECC-based systems is largely dominated by the computational cost of these point operations. In particular, modular inversion and modular multiplication are the most



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

time-consuming arithmetic operations involved in scalar multiplication. To mitigate the high cost of modular inversion, ECC implementations commonly employ projective coordinate systems, which replace expensive inversion operations with a higher number of modular multiplications. Since modular multiplication becomes the most frequent and computationally dominant operation in projective coordinate-based ECC, optimizing multiplication efficiency is crucial for improving overall system performance. In this context, Ancient Indian Vedic Mathematics (AIVM) offers a promising set of arithmetic techniques for accelerating complex mathematical computations. Vedic Mathematics consists of sixteen sutras and thirteen sub-sutras that provide efficient and parallelizable methods for arithmetic operations such as multiplication and division. Among these, techniques such as *Urdhva Tiryagbhyam* and *Dvandva Yoga* are particularly well-suited for high-speed multiplication, making them attractive for cryptographic arithmetic optimization. In this paper, elliptic curve cryptography over prime fields $F_p$ is studied using a projective coordinate system integrated with selected Vedic Mathematics techniques. The proposed approach focuses on accelerating ECC point addition and point doubling operations by incorporating AIVM-based multiplication methods into the projective coordinate framework. Furthermore, the projective coordinate system is remodeled by interacting point addition and point doubling operations to enable efficient parallel computation. This design significantly reduces execution time while maintaining cryptographic correctness and security. An elliptic curve over a prime field is defined by the equation: $E:y^2=x^3+ax+b \pmod p$, where $a,b \in F_p$, satisfy the non-singularity condition $4a^3+27b^2 \neq 0 \pmod p$. Together with a specially defined point at infinity, the set of points on the curve forms an abelian group under point addition. Scalar multiplication, defined as repeated point addition, constitutes the core cryptographic operation in ECC and directly relies on efficient implementations of point addition and doubling. The proposed scheme is implemented using parallel projective coordinate architectures and evaluated on Field Programmable Gate Arrays (FPGAs) with parametric configurations, including the number of parallel multipliers, adders, and operand widths. Performance analysis demonstrates that the proposed AIVM-based projective coordinate design achieves reduced execution time and improved area-time-squared ($AT^2$) cost compared to conventional projective coordinate ECC implementations. These results confirm that integrating Vedic Mathematics with projective coordinate ECC provides a viable and efficient approach for high-speed and low-power cryptographic applications.

**LITERATURE SURVEY**

Elliptic Curve Cryptography (ECC) has been widely recognized as an efficient and secure public key cryptographic technique since its independent introduction by Koblitz (1987) and Miller (1986). The mathematical foundations of elliptic curves over finite fields and their group properties have been extensively studied, establishing ECC as a practical alternative to classical cryptosystems based on integer factorization and discrete logarithms (Menezes & Vanstone, 1993; Menezes, 1993). The security of ECC relies on the intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which is computationally infeasible for suitably chosen curves (Schoof, 1985; Menezes et al., 1993). Compared to traditional public key cryptosystems such as RSA and Diffie-Hellman, ECC provides equivalent security with significantly smaller key sizes, reduced bandwidth requirements, and lower power consumption (Diffie & Hellman, 1976; Stallings, 2003; Hankerson et al., 2004). These advantages make ECC particularly suitable for constrained platforms such as smart cards, embedded systems, and portable devices. Bailey and Paar (2001) demonstrated that efficient finite field arithmetic plays a critical role in the performance of ECC, especially in hardware implementations. A significant portion of ECC computation time is consumed by scalar multiplication, which is composed of repeated point addition and point doubling operations. To improve efficiency, researchers have investigated various coordinate representations to reduce the cost of modular inversion. Projective coordinate systems have been shown to effectively eliminate costly inversion operations by replacing them with additional multiplications, leading to substantial performance gains (Bosma & Lenstra, 1995; Menezes & Vanstone, 1993). Several studies have also explored alternative elliptic curve forms and addition laws to optimize point operations (Chillali, 2011; Chillali et al., 2015). Parallelism and hardware acceleration have further contributed to ECC performance improvements. FPGA-based and VLSI implementations employing parallel multipliers and adders have been reported to achieve reduced execution time and improved area-time trade-offs (Thapliyal & Arbania, 2004; Haveliya, 2012). Karatsuba and Ofman (1963) introduced fast multiplication techniques that influenced later cryptographic hardware designs, highlighting the importance of efficient multiplication algorithms in



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

cryptosystems. In parallel with developments in ECC, Ancient Indian Vedic Mathematics (AIVM) has gained attention for its efficient arithmetic techniques. Tirthaji (1986) formalized sixteen Vedic sutras that provide fast and parallelizable methods for arithmetic operations. Among these, *Urdhva Tiryagbhyam* has been widely used for high-speed multiplication in digital systems. Several researchers demonstrated that Vedic multipliers outperform conventional multipliers in terms of speed, area, and power consumption (Kanhe et al., 2012; Kumar & Charishma, 2012; Pawar et al., 2014). Vedic Mathematics techniques have been successfully applied to cryptographic algorithms, particularly RSA and AES implementations. Bhaskar et al. (2012), Salim and Lakhotiya (2015), and Sadanandan and Anjali (2014) showed that Vedic-based arithmetic significantly improves encryption and decryption performance. Anchalya et al. (2015) further highlighted the effectiveness of Vedic sutras in accelerating complex computations in digital systems. Recent studies have extended the application of AIVM to ECC. Shylashree et al. (2015) demonstrated that Vedic Mathematics can be used to optimize scalar multiplication over prime fields GF(p)GF(p)GF(p), resulting in improved execution speed. Similar approaches employing Vedic multipliers for ECC arithmetic reported reduced computation time and enhanced hardware efficiency (Nanda & Behera, 2014; Palata et al., 2017). However, most of these works focus on standard affine or conventional projective coordinate systems without deeply exploring remodeled projective architectures or parallel point operation interactions. Research on elliptic curves over rings and alternative algebraic structures has also contributed to a deeper understanding of elliptic curve arithmetic and its generalizations (Tadmori et al., 2014; Tadmori et al., 2015; Chillali, 2011). These studies provide valuable insights into elliptic curve representations and arithmetic properties, which can be leveraged to design more efficient cryptographic schemes. From the existing literature, it is evident that while ECC efficiency has been improved through projective coordinates, parallel hardware architectures, and fast multiplication techniques, limited work has been done on the systematic integration of Vedic Mathematics techniques within remodeled projective coordinate systems. This paper addresses this research gap by combining AIVM-based multiplication with optimized projective coordinate point addition and doubling operations, aiming to achieve higher computational efficiency and lower execution cost for ECC over prime fields. From 2015 onwards, research on elliptic curve cryptography and arithmetic optimization has increasingly emphasized efficiency and lightweight implementations. In 2015, Anchalya et al. (2015) demonstrated the effectiveness of Vedic Mathematics sutras in accelerating complex arithmetic operations, while Shylashree et al. (2015) successfully applied Ancient Indian Vedic Mathematics (AIVM) to optimize elliptic curve scalar multiplication over prime fields, highlighting its potential in ECC implementations. During 2016, research primarily focused on refining low-power and area-efficient arithmetic architectures for cryptographic hardware, strengthening the foundation for fast modular multiplication. In 2017, Palata et al. (2017) and Kapse et al. (2017) proposed optimized Vedic multiplier designs that achieved improved speed and reduced hardware complexity, reinforcing the suitability of Vedic arithmetic for cryptographic applications. Research in 2018 expanded toward lightweight cryptographic solutions for embedded and IoT devices, emphasizing the importance of efficient scalar multiplication in ECC. By 2019, studies increasingly concentrated on optimizing projective coordinate systems and parallel point operations to reduce ECC computation time, although integration with Vedic Mathematics remained limited. In 2020, parallel and FPGA-based ECC implementations gained significant attention, with researchers highlighting the dominant role of modular multiplication in overall ECC performance. During 2021, the focus shifted toward energy-efficient and secure ECC implementations for IoT and edge computing, where Vedic-based arithmetic units were identified as promising candidates for accelerating modular multiplication. In 2022, algorithm-hardware co-design approaches were emphasized to improve the area-time efficiency of ECC systems, further motivating the exploration of alternative arithmetic techniques. Research trends in 2023 reflected the growing demand for high-speed cryptography in blockchain, mobile, and IoT applications, with increasing interest in parallel scalar multiplication and optimized coordinate representations. By 2024, studies acknowledged that modular multiplication remains the primary bottleneck in projective coordinate ECC, and Vedic Mathematics-based multipliers were recognized as efficient alternatives in hardware implementations. Up to 2025, existing literature indicates a clear research gap in the systematic integration of AIVM techniques within remodeled projective coordinate systems for ECC, highlighting the significance of developing high-speed, low-cost, and energy-efficient cryptographic solutions based on this combined approach.



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

## MATHEMATICAL BACKGROUND OF ELLIPTIC CURVE CRYPTOGRAPHY AND THE RING $A_4$

**Encryption System:** A cryptographic system or encryption system can be characterized by the tuple $(C, P, K, E, \text{and } D)$ with these five axioms:

1. The elements of the set $C$ (Cipher text space) are called Cipher text.
2. The elements of the set $P$ (Plaintext space) are called Plaintext.
3. The elements of the set $K$ (Key space) are called Keys.
4. The set $E = \{E_k : k \in K\}$ having functions $E_k : P \to C$ and all the elements of $E$ it is called an encryption function.
5. The set $D = \{D_k : k \in K\}$ having functions $D_k : C \to P$ and all the elements of $D$ it is called the decryption functions. For each, $e \in K$, there is $d \in K$, such that $D_d(E_e(\text{p})) = p$ for all $p \in P$

**Elliptic Curves:** Let a finite field $(F_p)$ with the prime modulo $p$ and $a, b \in F_p$ such that $(4a^3 + 27b^2) \bmod p \neq 0$. The equation

$y^2 = (x^3 + ax + b) \bmod p$ , i.e. $E(F_p) = \{(x, y) : y^2 = (x^3 + ax + b) \bmod p\} \cup \{O\}$.

Satisfied by the $E(F_p)$ fact that there $E(F_p)$ is an elliptic curve with the subject $F_p$ and its set of points $x$ and $y$. The identity element under the addition operation
, i.e. $P + O = O + P = P \quad for\ every \quad P = (x, y) \in E(F_p)$;
Where $O$ is showing the point at infinity. Literally elliptic curve is different from an ellipse. Elliptic cure called similar to an ellipse because they have the same cubic equation and the same way to get the circumference.

**For an elliptic curve, $E(F_p)$ two points are defined as:**

Let P and Q be the points of an elliptic curve, where $P, Q = (x_1, y_1), (x_2, y_2) \in E(F_p)$ then the sum of these points is $R = (x_3, y_3) \in E(F_p)$, which is explained by the following formulation:

$x_3 = (\lambda^2 - x_1 - x_2) \bmod p \ \ and \ \ y_3 = (\lambda(x_1 - x_3) - y_1) \bmod p$ where

$$\lambda = \begin{cases} \left(\dfrac{y_2 - y_1}{x_2 - x_1}\right) \bmod p \quad if \quad P \neq Q \\ \left(\dfrac{3x_1^2 + a}{2y_1}\right) \bmod p \ \ if \ \ P = Q \end{cases}$$

**For an elliptic curve $E(F_p)$ Additive inverse element of a point:**

Let P be a point over $E(F_p)$ , and the inverse of this point is $-P$ these are difined as and
$-P = (x, -y \bmod p) \in E(F_p)$ i.e $(x, y) + (x, -y \bmod p) = O$ .
It can be seen that E($F_p$) forms an abelian group under the addition operation defined as above.



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

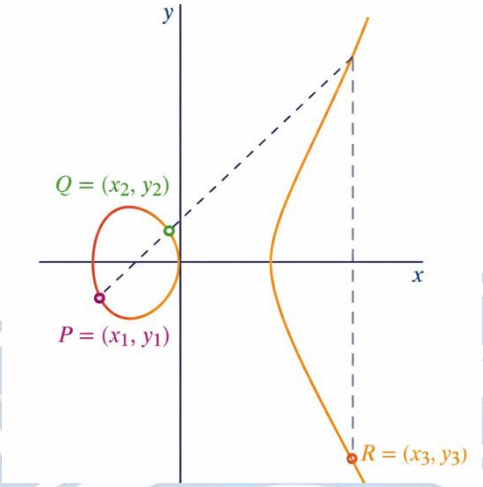


Fig. 1: Addition operation for the points P and Q (R=P+Q)

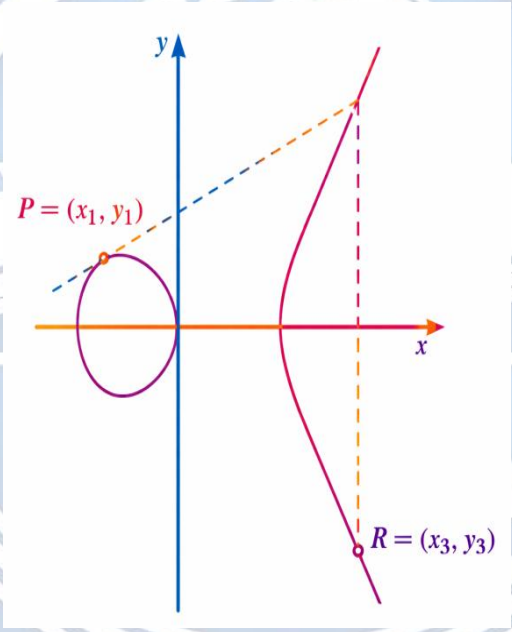


Fig. 2: Doubling operation with point P (R = 2P)

## ANCIENT INDIAN VEDIC MATHEMATICS

Across India and out of India speedily in the field of AIVM (Ancient Indian Vedic Mathematics), which contains the sixteen sutras or formulae and fourteen sub-sutras in operations of all branches of mathematics. In this AIVM section, we will discuss some useful techniques methods first one is Urdhva Triyagbhyam and second is Dvanda Yoga of Ancient Indian Vedic Mathematics which will be used in later section to improve the performance of ECC based schemes over the ring $A_4$

**Urdhva-Tiryagbhyam multiplication**

graph of research is increasing Urdhva-Tiryagbhyam multiplication technique is used for general multiplication [20]. This sutra directly explains a way in the form of vertical and crosswise, which applying on the digits. This sutra is very useful in all AIVM Sutras, and it has applied in many applications in all kinds of branches of mathematical science. In this AIVM techniques-based paper, we explain this method for two and three-digit numbers and all. This sutra has a plus point and is so beneficial because, by the use of this AIVM technique, we get a reduction in



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

multiplications of multi-bit down to one-bit. A Comparison between the Vedic multiplier and many other multipliers, like Booth and Array multiplier, shows that AIVM based multiplier do not take much time and these multiplier saves some power compared to other multipliers. Naturally, by using the standard or classical method in multiplication, we can get results, but the number of operations in the classical method is almost too high for m-bit digit integers, so it looks complicated. Another multiplication method is the Karatsuba method, its require number of operation is $m^{\log 3}$ for two integers of m bits. So it's a little bit complicated method of multiplication. Karatsuba technique (method) gives slow output for a small input in any operations comparatively classical technique (methods) of multiplication due to repetition of overhanging operations. To manipulate this type of issue best sutra of the AVIM Urdhva Tiryagbhyam technique can be applied.

**Multiplication of two-digit numbers**

$$(px+q)(rx+s) = prx^2 + (ps+qr)x + qs$$

Where $x$ is the base of the number system

*Ex. Evaluate* $42 \times 26$ *using urdhva triyagbhyam technique.*

| | | |
|---|---|---|
| $4 \times 2$ | $4 \times 6 + 2 \times 2$ | $2 \times 6$ |
| 8 | 28 | 12 |
| $8 + 2$ | $8 + 1$ | 2 |
| 10 | 9 | 2 |

$= 1092$

**Multiplication of three-digit numbers**

$\left(px^2 + qx + r\right)\left(sx^2 + tx + u\right) = psx^4 + \left(pe + qs\right)x^3 + \left(pu + qt + rs\right)x^2 + \left(qu + tr\right)x + ru$ Where $x$ is the base of the number system

*Ex.: Evaluate* $122 \times 134$ *using urdhva triyagbhyam technique.*

| | | | | |
|---|---|---|---|---|
| $1 \times 1$ | $1 \times 3 + 2 \times 1$ | $2 \times 3 + 1 \times 4 + 2 \times 1$ | $2 \times 4 + 2 \times 3$ | $2 \times 4$ |
| 1 | 5 | 12 | 14 | 8 |
| 1 | $5 + 1$ | $2 + 1$ | 4 | 8 |

$= 16348$

**Urdhva-Tiryagbhyam A lgorithm**



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

1: **Urdhva – Tiryagbhyam Algorithm**

2: $for\ i \in \mathrm{N}-1\ do$

3: $for\ j \in i\ do$

4: $sum \leftarrow sum + \alpha(\mathrm{i})\ \mathrm{and}\ \beta(\mathrm{i}-\mathrm{j})$

5: end for

6: $\mathrm{temp}(2\,\mathrm{N}+(\log_2 2)-\mathrm{i}:2N-1) \leftarrow sum$

7: $temp(\mathrm{others}) \leftarrow '0'$

8: $u \leftarrow u + temp$

9: $end\ \ for$

10: $for\ i \in N-2\ do$

11: $for\ j \in (\mathrm{i}+1) \rightarrow (\mathrm{N}-1)\ \mathrm{do}$

12: $sum \leftarrow sum + \alpha(\mathrm{i})\ \mathrm{and}\ \beta(\mathrm{N}-\mathrm{i}-\mathrm{j})$

13: end for

14: $\mathrm{temp}(\mathrm{N}+(\log_2 2)-\mathrm{i}:N-1) \leftarrow sum$

15: $temp(\mathrm{others}) \leftarrow '0'$

16: $u \leftarrow u + temp$

17: $end\ \ for$

18: $end\ \ procedure$

**Dvandva-Yoga Technique**

The Dvandva Yoga [19, 20] is the best technique of AIVM, which is used for squaring a $n$ digit number. The Dvandva Yoga technique is also known as the duplex of a number. To calculate Dvandva Yoga of a number which contain single digit Dvandva Yoga, express it as the square of that number. As follows, the Dvandva Yoga of 4 is equal to 16, and 7 is equal to 49, and all, to calculate the Dvandva Yoga of a number which contain two digit Dvandva Yoga, expressed as the double the multiplication of both digits of that number. As follows the Dvandva Yoga of number 18 is equal to $(2\times1\times8)=16$, 70 equal to $(2\times7\times0)=00$ and 45 is equal to $2\times4\times5=40$ up to and so on. The Dvandva Yoga is manipulated as the addition of a few individual Dvandva Yogas for n-digit numbers. Put together the primary digit and the $n^{th}$ digit of any number and calculate the Davndva Yoga of this two-digit figure. In the same way, put together the secondary digit along with $(n-1)^{th}$ the digit, and then calculate the Dvandva Yoga of this figure. Repeat this procedure until there are no pairs of two-digit numbers left. If there exists a digit in the center of an n-digit number that cannot be paired with any one of the other digits in the final stage, then we can determine Dvandva Yoga of this digit, as usually, after this, take the summation for all Dvandva Yogas that have been calculated as above. The above outcome summations are the sum of the Dvandva Yoga for an n-digit number. Dvandva Yoga for some digit number can be illustrated as below:



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

STEP1: $D_Y(p)= p^2$

STEP 2: $D_Y(pq)=2pq$

STEP 3: $D_Y(pqr)=2pr+q^2$

STEP 4: $D_Y(pqrs)=2ps+2qr$

STEP 5: $D_Y(pqrst)=2pt+2qs+r^2$

*And all for $n^{th}$ digit numbers …*

As we can see above, $D_Y$ (Dvandva Yoga) of any number is the sum of the square of the middle number and two times the product of the other pairs.

**A number's square is explained by**

$$(pq)^2 = D_Y(p) \mid D_Y(pq) \mid D_Y(q)$$

$$(pqr)^2 = D_Y(p) \mid D_Y(pq) \mid D_Y(pqr) \mid D_Y(qr) \mid D_Y(r)$$

$$(pqrs)^2 = D_Y(p) \mid D_Y(pq) \mid D_Y(pqr) \mid D_Y(pqrs) \mid D_Y(qrs) \mid D_Y(qr) \mid D_Y(r)$$

*Ex.: Evaluate this five digits number square $(12345)^2$ using Dvandva Yoga technique.*

STEP1: $D_Y(5)= 5^2 = 25$

STEP 2: $D_Y(45)=2(4\times5) = 40$

STEP 3: $D_Y(345)=2(3\times5)+4^2 = 46$

STEP 4: $D_Y(2345) = 2(2\times5)+2(3\times4) = 44$

STEP 5: $D_Y(12345) = 2(1\times5)+2(2\times4)+3^2 = 35$

STEP6: $D_Y(1234)= 2\times(1\times4)+2(2\times3) = 20$

STEP 7: $D_Y(123)=2(1\times3)+2^2 = 10$

STEP 8: $D_Y(12)=2(1\times2) = 4$

STEP 9: $D_Y(1) = 1^2 = 1$

Answer is 152399025.

*Ex: Evaluate this Three digits number square $(789)^2$ using Dvandva Yoga technique.*

STEP1: $D_Y(9)= 9^2 = 81$

STEP 2: $D_Y(89)=2(8\times9) = 144$

STEP 3: $D_Y(789)=2(7\times9)+8^2 = 190$

STEP 4: $D_Y(78) = 2(7\times8) = 112$

STEP 5: $D_Y(7) = 7^2 = 49$

Answer is 622521.



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

## ELLIPTIC CURVE CRYPTOSYSTEM USING URDHVA-TIRYAGBHYAM AND DVANDA-YOGA TECHNIQUE

This section describes a few imperative calculations of ECC (Elliptic Curve Cryptography) using some effective algorithms and techniques of AIVM (Ancient Indian Vedic Mathematics).

Over a field $F_p$ having finite elements, the term $E(F_p)$ elliptic curve is explained by

$$E(a,b)=\left\{(x,y):y^2=x^3+ax+b \text{ and } 4a^3+27b^2\neq 0\right\}\cup\{O\}$$

We can use Urdhva Tiryagbhyam and Dvanda Yoga technique to evaluate $\lambda^2$, and $\lambda(x_1-x_3)$ for an elliptic curve, the sum of two points (P, Q) and multiplication of two same points (P, P) is explained by:

If $P,Q=\left[(x_1,y_1),(x_2,y_2)\in E(F_p)\right]$ for case $P\neq Q$, then $R=P+Q,\ R=(x_3,y_3)\in E(F_p)$ in this formulation $x_3, y_3, \text{ and } \lambda$ is $\lambda^2-x_1-x_2$, $\lambda(x_1-x_3)-y_1$ and $\dfrac{(y_2-y_1)}{(x_2-x_1)}$ respectively.

In the elliptic curve for addition of the same two points (Doubling), we apply these expressions:

If $P=(x,\ y)$ then the doubling of the same point (P, P) is $Q=P+P=2P=(x_3,y_3), P \text{ and } Q\in E(F_p)$

In this expression $x_1, y_1, \text{ and } \lambda$ is $\lambda^2-2x$, $\lambda(x-x_1)-y$ and $\dfrac{(3x^2+a)}{2y}$ respect.

The terms $\lambda^2$, $\lambda(x-x_1)$ and $3x^2$ can be calculated using the Urdhva Tiryagbhyam and Dvanda Yoga technique.

### MATHEMATICAL BACKGROUND OF ECC PROJECTIVE COORDINATE SYSTEM OVER THE PRIME FIELD $F_P$

An elliptic curve can be represented using two primary coordinate systems: the affine coordinate system and the pure projective coordinate system. The affine coordinate system requires modular inversion during point addition and point doubling operations, which is computationally expensive in terms of execution time and hardware area. To avoid this costly inversion operation, pure projective coordinate systems are commonly employed, as they eliminate inversions at the expense of introducing a higher number of modular multiplications. To balance these trade-offs, mixed coordinate systems are often preferred, where one point is represented in projective coordinates and the other in affine coordinates during point addition. In elliptic curve arithmetic, each point addition and point doubling operation involves a fixed number of modular multiplications, squaring, additions, shifts, and other basic arithmetic operations. The exact number of operations depends on the chosen curve representation and coordinate system; however, modular multiplication and squaring operations generally dominate the overall computational cost, and consequently, the execution time scales proportionally with the number of these arithmetic operations required.

Let $E: y^2=x^3+ax+b$ be an elliptic curve defined over a finite field $F$. Then, we know that [12-14] each elliptic curve point can be described by two coordinates $x, y\in F$. In this case, we say that elliptic curve points belong to two dimensional affine plane $A_F^2=\{(x,y)\in F\times F\}$. Suppose the coordinates $(x,y)$ of the affine plane $A_F^2=\{(x,y)\in F\times F\}$ are mapped to the coordinates $(X,Y,Z)$ of the projective plane $P_F^3=\{(X,\ Y,\ Z)\in F\times F\times F\}$ as

$$(X,Y,Z)=(x.Z^c, y.Z^d, 1) \text{ or } x=X/Z^c \text{ and } y=Y/Z^d \qquad (1)$$

where $c, d$ are integers.

If $P_1=(X_1,Y_1,Z_1)$ and $P_2=(X_2,Y_2,Z_2)$ are two distinct points on the projective plane, then their point addition $(P_3=(X_3,Y_3,Z_3)=P_1+P_2)$ and point doubling $(P_3=2P_1)$ can be described as follows.

**Addition of Points on Projective Plane:**



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

**Case - I:** If $x_1 \neq x_2$ then we have $\lambda = \frac{y_2 - y_1}{x_2 - x_1} = \frac{Y_2 - \frac{Y_1}{Z_1^d}}{X_2 - \frac{X_1}{Z_1^c}}$ . It is obvious from the above expression that $\lambda$ exist because $x_1 \neq x_2$ . Now the point $P_3$ can be calculated as

$$x_3 = \lambda^2 - x_1 - x_2 = \left( \frac{Y_2 - \frac{Y_1}{Z_1^d}}{X_2 - \frac{X_1}{Z_1^c}} \right)^2 - \frac{X_1}{Z_1^c} - X_2,$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \left( \frac{Y_2 - \frac{Y_1}{Z_1^d}}{X_2 - \frac{X_1}{Z_1^c}} \right) \left( \frac{X_1}{Z_1^c} - x_3 \right) - \frac{Y_1}{Z_1^d}.$$

To eliminate denominators in the expression, $x_3$ *and* $y_3$ we set $X_3 = x_3.Z_3^c$ *and* $Y_3 = y_3.Z_3^d$ where $Z_3$ can be calculated by the denominator expression of coordinates type.

**Doubling of Points on Projective Plane:**

**Case II:** If $x_1 = x_2$ we have for point doubling, we can take $P_1 = P_2$ then $P_3 = P_1 + P_2 = 2P_1 = (X_3, Y_3, Z_3)$ .

We have

$$\lambda = \frac{3x_1^2 + a}{2y_1} = \left( \frac{3\left( \frac{X_1}{Z_1^c} \right)^2 + a}{2\frac{Y_1}{Z_1^d}} \right),$$

$$x_3 = \left( \frac{3\left( \frac{X_1}{Z_1^c} \right)^2 + a}{2\frac{Y_1}{Z_1^d}} \right)^2 - 2\frac{X_1}{Z_1^2} = \frac{(3X_1^2 + aZ_1^{2c})^2 Z_1^{2d} - 8Z_1^{3c} X_1 Y_1^2}{4Z_1^{4c} Y_1^2}$$

Evidently $\lambda$ exists if $y_1 \neq 0$ , so we get

$$x_3 = \lambda^2 - 2x_1 = \frac{(3X_1^2 + aZ_1^{2c})^2 Z_1^{2d}}{4Z_1^{4c} Y_1^2} - 2\frac{X_1}{Z_1^c}, = \frac{(3X_1^2 + aZ_1^{2c})^2 Z_1^{2d} - 8Z_1^{3c} X_1 Y_1^2}{4Z_1^{4c} Y_1^2}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \lambda(3x_1 - \lambda^2) - y_1$$

$$= \frac{(3X_1^2 + aZ_1^{2c})^2 Z_1^d}{2Z_1^{2c} Y_1} \left( 3\frac{X_1}{Z_1^c} - \frac{(3X_1^2 + aZ_1^{2c})^2 Z_1^{2d}}{4Z_1^{4c} Y_1^2} \right) - \frac{Y_1}{Z_1^d}$$

$$= \frac{12X_1 Y_1^2 (3X_1^2 + aZ_1^{2c})^2 Z_1^{3c+2d} - (3X_1^2 + aZ_1^{2c})^3 Z_1^{4d} - 8Z_1^{6c} Y_1^4}{8Z_1^{6c+d} Y_1^3}$$

There are a number of types of coordinates when c and d are set to different values. The three projective coordinate system currently used for representing the $y^2 = x^3 + ax + b$ elliptic curve defined over a finite field. *F*



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

1. Jacobian coordinates, If c=2 and d=3.
2. Standard projective coordinates, If c=1and d=1.
3. Lopez Dahab system If c=1 and d=2.

These are a projective coordinate system currently used for representing the different elliptic curves defined over a finite field. *F*

4. Chudnovsky-Jacobian Coordinate Systems.
5. The Modified Jacobian Coordinates.
6. Inverted Edwards coordinates.
7. Hessian Projective coordinates.

**ELLIPTIC CURVE CRYPTOGRAPHY OVER THE CURVE $y^2 = x^3 + ax + b$ USING DIFFERENT PROJECTIVE COORDINATE SYSTEMS**

Jacobian Coordinates are used to represent elliptic curve points on prime curves $y^2 = x^3 + ax + b$. They give a speed benefit over Affine Coordinates when the cost for field inversions is significantly higher than field multiplications. In Jacobian coordinates, the triple $(X,\ Y,\ Z)$ represents the affine point. $(X\ /\ Z^2,\ Y\ /\ Z^3)$.

After applying the Jacobian projective transformation in equation (1) with $c = 2$ and $d = 3$, the elliptic curve $E$ can be rewritten as $E : Y^2 = X^3 + aXZ^4 + bZ^6$.

If $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$ are two distinct points on the projective plane, then their point addition $(P_3 = (X_3, Y_3, Z_3) = P_1 + P_2)$ and point doubling $(P_3 = 2P_1)$ can be described as follows.

**Addition of Points on Jacobian Projective Plane: Jacobian coordinates, If c=2 and d=3.**

Using Case I, we get a Jacobian projective transformation with $c = 2$ and $d = 3$,

The addition of $P_1$ and $P_2$ is $P_3$ given by

If $x_1 \neq x_2$ then we have $\lambda = \dfrac{y_2 - y_1}{x_2 - x_1} = \dfrac{Y_2 - \frac{Y_1}{Z_1^3}}{X_2 - \frac{X_1}{Z_1^2}}$ It is obvious from the above expression that $\lambda$ exist because $x_1 \neq x_2$. Now the point $P_3$ can be calculated as

$$x_3 = \lambda^2 - x_1 - x_2 = \left( \frac{Y_2 - \frac{Y_1}{Z_1^3}}{X_2 - \frac{X_1}{Z_1^2}} \right)^2 - \frac{X_1}{Z_1^2} - X_2,$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \left( \frac{Y_2 - \frac{Y_1}{Z_1^3}}{X_2 - \frac{X_1}{Z_1^2}} \right) \left( \frac{X_1}{Z_1^2} - x_3 \right) - \frac{Y_1}{Z_1^3}.$$

To eliminate denominators in the expression, $x_3$ *and* $y_3$ we set $X_3 = x_3.Z_3^2$ *and* $Y_3 = y_3.Z_3^3$ where $Z_3$ can be calculated by the denominator expression of coordinate's type and obtain the following formulas for computing $P_1 + P_2 = (X_3 : Y_3 : Z_3)$ in Jacobian coordinates:



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

$$X_3 = (Y_2 Z_1^{\,3} - Y_1)^2 - (X_1 + X_2 Z_1^{\,2})(X_2 Z_1^{\,2} - X_1)^2$$

$$Y_3 = (Y_2 Z_1^{\,3} - Y_1)\left( X_1 \left( X_2 Z_1^{\,2} - X_1 \right)^2 - X_3 \right) - Y_1 \left( X_2 Z_1^{\,2} - X_1 \right)^3$$

$$= (Y_2 Z_1^{\,3} - Y_1)\left( 2X_1 + X_2 Z_1^{\,2} \right)\left( X_2 Z_1^{\,2} - X_1 \right)^2 - (Y_2 Z_1^{\,3} - Y_1)^3 - Y_1 \left( X_2 Z_1^{\,2} - X_1 \right)^3$$

$$Z_3 = (X_2 Z_1^{\,2} - X_1) Z_1$$

**Algorithm for the addition of points on the Jacobian projective plane**

For adding two points $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$, we use the following calculation:

**Algorithm**

**Input** : $P_1 = (X_1, Y_1, Z_1), P_2 = (X_2, Y_2, Z_2)$

**Output** : $P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$

$A = Z_1^2$

$B = Z_1 \cdot A$

$C = X_2 \cdot A$

$D = Y_2 \cdot B$

$E = C - X_1$

$F = D - Y_1$

$G = E^2$

$H = G \cdot E$

$I = X_1 \cdot G$

$X_3 = F^2 - H - 2I$

$Y_3 = F \cdot (I - X_3) - Y_1 \cdot H$

$Z_3 = Z_1 \cdot E$

Finally, we calculate to get $X_3 = F^2 - H - 2I$, $Y_3 = F \cdot (I - X_3) - Y_1 \cdot H$, *and* $Z_3 = Z_1 \cdot E$ and finally.

$P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$.

Here we can use the Urdhva-Tiryagbhyam technique in all multiplications and the Dvandwa-Yoga technique for all squares.

**Doubling of a Point on Jacobian Projective Plane: Jacobian coordinates, If c=2 and d=3.**

Using **Case - II** Jacobian projective transformation with $c = 2$ and $d = 3$,

The doubling of the point $P_1$ is given by $P_3(X_3, Y_3, Z_3)$

If $x_1 = x_2$ then we have $\lambda = \dfrac{3x_1^2 + a}{2y_1} = \left( \dfrac{3\dfrac{X_1}{Z_1^{\,2}} + a}{2\dfrac{X_1}{Z_1^{\,3}}} \right)$ It is obvious from the above expression that $\lambda$ exist because $x_1 \neq x_2$. Now the point $P_3$ can be calculated as



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

$$x_3 = \lambda^2 - 2x_1 = \left( \frac{3\frac{X_1}{Z_1^2} + a}{2\frac{X_1}{Z_1^3}} \right)^2 - 2\frac{X_1}{Z_1^2}$$

$$= \frac{(3X_1^2 + aZ_1^4)^2 Z_1^6}{4Z_1^{4c} Y_1^2} - 2\frac{X_1}{Z_1^2} = \frac{(3X_1^2 + aZ_1^4)^2 - 8X_1Y_1^2}{4Z_1^2 Y_1^2}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \left( \frac{3\frac{X_1}{Z_1^2} + a}{2\frac{X_1}{Z_1^3}} \right) \left( \frac{X_1}{Z_1^2} - x_3 \right) - \frac{Y_1}{Z_1^3}$$

$$= \frac{(3X_1^2 + aZ_1^4)}{2Y_1Z_1} \left( \frac{X_1}{Z_1^2} - x_3 \right) - \frac{Y_1}{Z_1^3},$$

To eliminate denominators in the expressions for $x_3$ and $y_3$, we set $X_3 = x_3 \cdot Z_3^2$ *and* $Y_3 = y_3 \cdot Z_3^3$ where $Z_3$ can be calculated by the denominator expression of coordinates type and obtain the following formulas for computing $2P = (X_3 : Y_3 : Z_3)$ in Jacobian coordinates:

$$X_3 = (3X_1^2 + aZ_1^4)^2 - 8X_1Y_1^2, Y_3 = (3X_1^2 + aZ_1^4)(4X_1Y_1^2 + X_3) - 8Y_1^4$$
$$= 12X_1Y_1^2(3X_1^2 + aZ_1^4) - (3X_1^2 + aZ_1^4)^3 - 8Y_1^4 \text{ and } Z_3 = 2Y_1Z_1$$

**Algorithm for Doubling of a point on the Jacobian projective plane**

For point doubling $(P_3 = 2P_1)$, we perform the following manipulations:

**Algorithm**

**Input** : $P_1 = (X_1, Y_1, Z_1), a$

**Output** : $P_3 = (X_3, Y_3, Z_3) = 2P_1$

$A = Y_1^2$

$B = 4X_1 \cdot A$

$C = 8A^2$

$D = 3X_1^2 + a.Z_1^4$

$X_3 = D^2 - 2B$

$Y_3 = D \cdot (B - X_3) - C$

$Z_3 = 2Y_1 \cdot Z_1$

Finally, we calculate to get $X_3 = D^2 - 2B$, $Y_3 = D \cdot (B - X_3) - C$, $Z_3 = 2Y_1 \cdot Z_1$ and finally $P_3 = (X_3, Y_3, Z_3) = 2P_1$.

Here $X_1^2$, $X_1.Y_1^2$, $Y_1^4$, $A^2$ *and* $Y_1Z_1$ can be calculated using the Urdhva-Tiryagbhyam Dwandwa Yoga technique.

**Table.1: No. of operations required for point addition and point doubling**

| ECC over a prime field | No. of multiplications | No. of square | No. of cube | No. of fourth power |
|---|---|---|---|---|
| Point addition | 14 | 09 | 05 | 00 |
| Point doubling | 07 | 06 | 01 | 04 |

**Table. 2: No. of operation required for point addition and point doubling in Vedic Mathematics**



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

| ECC over a prime field with Vedic Mathematics | No. of multiplications | No. of square | No. of cube | No. of fourth power |
|---|---|---|---|---|
| Point addition | 08 | 03 | 00 | 00 |
| Point doubling | 04 | 04 | 00 | 01 |

**Standard projective coordinates. If c=1and d=1:** Standard Projective Coordinates are used to represent elliptic curve points on prime curves $y^2 = x^3 + ax + b$. Their usage might give a speed benefit over Affine Coordinates when the cost for field inversions is significantly higher than field multiplications. In Standard Projective coordinates, the triple (X, Y, Z) represents the affine point (X / Z, Y / Z). After applying the Standard projective transformation in equation (1) with $c=1$ and $d=1$, the elliptic curve $E$ can be rewritten as $E: Y^2Z = X^3 + aXZ^2 + bZ^3$. Using CASE I and Standard projective coordinate with $c=1$ and $d=1$, the addition of $P_1$ and $P_2$ is $P_3$ given by. If $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$ are two distinct points on the projective plane, then their point addition $(P_3 = (X_3, Y_3, Z_3) = P_1 + P_2)$ and point doubling $(P_3 = 2P_1)$ can be described as follows.

**Algorithm Addition of points on the standard projective coordinates plane**

For adding two points $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$, we use the following calculation:

**Algorithm**

**Input**: $P_1 = (X_1, Y_1, Z_1), P_2 = (X_2, Y_2, Z_2)$

**Output**: $P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$

$A = X_1 Z_2$

$B = X_2 Z_1$

$D = Y_1 Z_2$

$E = Y_2 Z_1$

$J = Z_1 Z_2$

$K = Y_1 Z_2$

$L = X_1 Z_2$

$C = A - B$

$F = E - D$

$G = A + B$

$H = F^2 J - GC^2$

$I = CL - H$

$X_3 = CH$

$Y_3 = FI - C^3 K$

$Z_3 = JC^3$

Finally, we calculate $X_3 = CH$, $Y_3 = FI - KC^3$ and $Z_3 = JC^3$ to get $P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$.

Here we can use Urdhva Tiryagbhyam and Dwandva Yoga technique for squaring of any n-digit number to evaluate $CH$, $FI$, $KC^3$ *and* $JC^3$. where



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

$$X_3 = \left[ (X_1Z_2 - X_2Z_1)(Y_2Z_1 - Y_1Z_2)^2 Z_1Z_2 - (X_1Z_2 + X_2Z_1)(X_1Z_2 - X_2Z_1)^3 \right]$$

$$Y_3 = \begin{bmatrix} (Y_2Z_1 - Y_1Z_2)(X_2Z_2 - X_2Z_1)^2 X_1Z_2 - (Y_2Z_1 - Y_1Z_2)^3 Z_1Z_2 \\ +(X_1Z_2 + X_2Z_1)(X_1Z_2 - X_2Z_1)^2 (Y_2Z_1 - Y_1Z_2) - (X_1Z_2 - X_2Z_1)^3 Y_1Z_2 \end{bmatrix}$$

$$and\ Z_3 = \left[ (X_1Z_2 - X_2Z_1)^3 Z_1Z_2 \right]$$

**Using CASE II** and the standard projective transformation with $c = 1$ and $d = 1$, the doubling of a point $P_1$ is given by $P_3(X_3, Y_3, Z_3)$ and point doubling $(P_3 = 2P_1)$ can be described as follows.

$$X_3 = 18X_1^4 + 2a^2Z_1^4 + 12aX_1^2Z_1^2 - 8X_1Y_1^3Z_1^2,$$

$$Y_3 = 3X_1^2Y_2Z_1^3 + Y_2Z_1^5 - 8Y_1^4Z_1^2$$

$$and\ Z_3 = 8Y_1^3Z_1^3$$

**Algorithm for Doubling of a point on the standard projective coordinates plane**

For point doubling $(P_3 = 2P_1)$, we perform the following manipulations:

**Algorithm**

**Input**: $P_1 = (X_1, Y_1, Z_1),\ a$

**Output**: $P_3 = (X_3, Y_3, Z_3) = 2P_1$

$$A = 3X_1^2 + aZ_1$$

$$B = Y_1Z_1$$

$$F = Y_1X_1$$

$$G = Y_1B$$

$$C = FB$$

$$D = A^2 - 8C$$

$$E = 4C - D$$

$$X_3 = 2BD$$

$$Y_3 = AE - 8G^2$$

$$Z_3 = (2B)^3$$

Here $BD, AE, B^3,\ and\ G$ can be calculated using Urdhva Tiryagbhyam and Dwandva Yoga technique for squaring of any n-digit number.

**Table. 3: No. of operation required for point addition and point doubling**

| ECC over a prime field | No. of multiplications | No. of square | No. of cube | No. of fourth power | No. of fifth power |
|---|---|---|---|---|---|
| Point addition | 39 | 3 | 4 | 0 | 0 |
| Point doubling | 10 | 6 | 4 | 3 | 1 |

**Table. 4: No. of operation required for point addition and point doubling in Vedic Mathematics**



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

| ECC over a prime field with AIVM | No. of multiplications | No. of square | No. of cube | No. of fourth power | No. of fifth power |
|---|---|---|---|---|---|
| Point Addition | 14 | 2 | 2 | 0 | 0 |
| Point doubling | 6 | 3 | 1 | 0 | 0 |

**Lopez Dahab Projective Coordinate. If c=1 and d=2:** Lopez Dahab Projective Coordinates (LDPC). Here $c=1$ and $d=2$. The projective point $(X:Y:Z), Z\neq 0$ corresponds to the affine point $(X/Z, Y/Z^2)$. The projective equation of the elliptic curve is $E: Y^2 = XZ + aXZ^3 + bZ^4$. Lopez Dahab Projective Coordinates are used to represent elliptic curve points on prime curves $y^2 = x^3 + ax + b$. Their usage might give a speed benefit over Affine Coordinates when the cost for field inversions is significantly higher than field multiplications. In the Lopez Dahab Projective coordinates, the triple $(X, Y, Z)$ represents the affine point $(X/Z, Y/Z^2)$. After applying the Lopez Dahab Projective Coordinates in equation (1) with $c=1$ and $d=2$, the elliptic curve $E$ can be rewritten as $E: Y^2 = XZ + aXZ^3 + bZ^4$. If $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$ are two distinct points on the projective plane, then their point addition $(P_3 = (X_3, Y_3, Z_3) = P_1 + P_2)$ and point doubling $(P_3 = 2P_1)$ can be described as follows.

**Addition of points on the Lopez Dahab projective coordinates plane**

For adding two points $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$, we use the following calculation:

**Addition of Points on Lopez Dahab Projective Plane: Lopez Dahab coordinates: If $c=1$ and $d=1$.**

Using **Case - I,** we get Lopez Dahab projective transformation with $c=1$ and $d=2$,
The addition of $P_1$ and $P_2$ is $P_3$ given by

If $x_1 \neq x_2$ then we have $\lambda = \dfrac{y_2 - y_1}{x_2 - x_1} = \dfrac{Y_2 - \frac{Y_1}{Z_1^2}}{X_2 - \frac{X_1}{Z_1}}$ It is obvious from the above expression that $\lambda$ exist because $x_1 \neq x_2$. Now the point $P_3$ can be calculated as

$$x_3 = \lambda^2 - x_1 - x_2 = \left(\frac{Y_2 - \frac{Y_1}{Z_1^2}}{X_2 - \frac{X_1}{Z_1}}\right)^2 - \frac{X_1}{Z_1} - X_2,$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \left(\frac{Y_2 - \frac{Y_1}{Z_1^2}}{X_2 - \frac{X_1}{Z_1}}\right)\left(\frac{X_1}{Z_1^2} - x_3\right) - \frac{Y_1}{Z_1^2}.$$

To eliminate denominators in the expression, $x_3$ *and* $y_3$ we set $X_3 = x_3.Z_3$ *and* $Y_3 = y_3.Z_3^2$ where $Z_3$ can be calculated by the denominator expression of the coordinate's type and obtain the following. Formulas for computing the addition of two points $P_1 + P_2 = (X_3 : Y_3 : Z_3)$ in Lopez Dahab Projective coordinates:



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

$$X_3 = (Y_2 Z_1^{\,3} - Y_1)^2 - Z_1(X_1 + X_2 Z_1)(X_2 Z_1^{\,2} - X_1)^2$$

$$Y_3 = Z_1^2(Y_2 Z_1^{\,2} - Y_1)\left(2X_1 + X_2 Z_1\right)\left(X_2 Z_1^{\,2} - X_1\right)^3$$

$$-Z_1\left(X_2 Z_1 - X_1\right)(Y_2 Z_1^{\,2} - Y_1)^3 - Y_1 Z_1^2\left(X_2 Z_1^{\,2} - X_1\right)^4$$

$$Z_3 = Z_1^2(X_2 Z_1 - X_1)^2$$

**Algorithm (Addition of points on Lopez Dahab Projective coordinates plane)**

For adding two points $P_1 = (X_1, Y_1, Z_1)$ and $P_2 = (X_2, Y_2, Z_2)$, we use the following calculation:

**Algorithm**

**Input**: $P_1 = (X_1, Y_1, Z_1), P_2 = (X_2, Y_2, Z_2)$

**Output**: $P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$

$A = Z_1^2$

$B = A \cdot Y_2$

$C = B - Y_1$

$D = X_2 \cdot Z_1$

$E = D - X_1$

$F = X_1 + D$

$G = \left(E^2 \cdot F\right) \cdot Z_1$

$H = 2X_1 + D$

$I = C \cdot H - Y_1 \cdot E$

$X_3 = C^2 - G$

$Y_3 = E\left(Z_3 \cdot I - Z_1 \cdot C^3\right)$

$Z_3 = A \cdot E^2$

Finally, we calculate $X_3 = C^2 - G$ $Y_3 = E\left(Z_3 \cdot I - Z_1 \cdot C^3\right)$ and $Z_3 = A \cdot E^2$ get $P_3 = (X_3, Y_3, Z_3) = P_1 + P_2$.

Here we can use Urdhva Tiryagbhyam and Dwandva Yoga technique for squaring of any n-digit number to evaluate $C^2$, $A \cdot E^2$, $Z_3 \cdot I$, $Z_1 \cdot C^3$ *and* $\left(E^2 \cdot F\right) Z_1$.

**Using Case II:** Lopez Dahab Projective transformation with $c = 1$ and $d = 2$,

The doubling of the point $P_1$ is given by $P_3(X_3, Y_3, Z_3)$

If $x_1 = x_2$ then we have $\lambda = \dfrac{3x_1^2 + a}{2y_1} = \left(\dfrac{3\dfrac{X_1}{Z_1} + a}{2\dfrac{X_1}{Z_1^{\,2}}}\right)$ It is obvious from the above expression that $\lambda$ exist because $x_1 \neq x_2$. Now the point $P_3$ can be calculated as



https://ijioa.com/ cosmospub2023@gmail.com

Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

$$x_3 = \lambda^2 - 2x_1 = \left(\frac{3\frac{X_1^2}{Z_1^2} + a}{2\frac{Y_1}{Z_1^2}}\right)^2 - 2\frac{X_1}{Z_1} = \frac{(3X_1{}^2 + aZ_1^2)^2}{4Y_1} - 2\frac{X_1}{Z_1} = \frac{(3X_1{}^2 + aZ_1{}^2)^2 - 8X_1Y_1}{4Y_1Z_1}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = \left(\frac{3\frac{X_1}{Z_1} + a}{2\frac{Y_1}{Z_1^2}}\right)\left(\frac{X_1}{Z_1} - x_3\right) - \frac{Y_1}{Z_1^2} = \frac{(3X_1{}^2 + aZ_1{}^4)}{2Y_1Z_1}\left(\frac{X_1}{Z_1^2} - x_3\right) - \frac{Y_1}{Z_1{}^3}.$$

To eliminate denominators in the expressions for $x_3$ and $y_3$, we set $X_3 = x_3 \cdot Z_3$ *and* $Y_3 = y_3 \cdot Z_3^2$ where $Z_3$ can be calculated by the denominator expression of coordinates type, and obtain the following formulas for computing $2P = (X_3 : Y_3 : Z_3)$ in Lopez Dahab Projective coordinates:

$X_3 = (3X_1{}^2 + aZ_1{}^2)^2 - 8X_1Y_1{}^2,$

$Y_3 = 24X_1Y_1{}^3Z_1(3X_1{}^2 + aZ_1{}^2) - 2Y_1Z_1(3X_1{}^2 + aZ_1{}^2)^3 - 16Y_1^5$

$Z_3 = 4Y_1^2Z_1$

**Algorithm (Doubling of a point on the Lopez Dahab Projective plane)**

For point doubling $(P_3 = 2P_1)$, we perform the following manipulations:

**Algorithm**

**Input** : $P_1 = (X_1, Y_1, Z_1), a$

**Output** : $P_3 = (X_3, Y_3, Z_3) = 2P_1$

$A = 4Y_1^2$

$B = 2X_1 \cdot A$

$C = 3X_1^2 + a.Z_1^4$

$D = X_1 - Z_1x_3$

$E = 2D \cdot E - A$

$X_3 = C^2 - B$

$Y_3 = Y_1 \cdot A \cdot E$

$Z_3 = A \cdot Z_1$

Here $D, A \cdot E, C^2,$ *and* $E$ can be calculated using Urdhva Tiryagbhyam and Dwandva Yoga technique for squaring of any n-digit number.

**Table 5: No. of operation required for point addition and point doubling**

| ECC over a prime field | No. of multiplications | No. of square | No. of cube | No. of fourth power | No. of fifth power |
|---|---|---|---|---|---|
| Point addition | 20 | 9 | 2 | 1 | 0 |
| Point doubling | 7 | 8 | 2 | 0 | 1 |

**Table 6: No. of operation required for point addition and point doubling in Vedic Mathematics**

| ECC over a prime field with AIVM | No. of multiplications | No. of square | No. of cube | No. of fourth power | No. of fifth power |
|---|---|---|---|---|---|
| Point Addition | 10 | 3 | 1 | 0 | 0 |
| Point doubling | 6 | 4 | 0 | 0 | 0 |



Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

## CONCLUSION

This work presented a high-speed and area-efficient implementation of Elliptic Curve Cryptography (ECC) over a prime field $F_p$ using the projective coordinate system integrated with Ancient Indian Vedic Mathematics (AVIM). ECC is well known for providing stronger security with significantly smaller key sizes compared to traditional public-key cryptosystems, resulting in reduced computational complexity, compact hardware/software implementations, and faster cryptographic operations. These characteristics make ECC highly suitable for modern security-critical and resource-constrained environments. In the proposed approach, the core ECC operations, point addition and point doubling, were optimized using Vedic mathematical techniques for multiplication and squaring. Since scalar multiplication is the most computationally intensive operation in ECC, the application of Vedic multiplication significantly improves the overall performance. The use of projective coordinates further eliminates expensive modular inversion operations, enhancing computational efficiency. The results indicate that the proposed Vedic-based ECC architecture achieves improved speed and reduced area compared to conventional multiplication-based ECC implementations. The AVIM techniques simplify complex arithmetic operations, minimize intermediate computations, and maintain a systematic correlation throughout the cryptographic process rather than relying on multiple unrelated optimization methods. This leads to better performance consistency and reduced hardware complexity. Overall, the proposed scheme demonstrates that integrating Vedic Mathematics with ECC over $F_p$ is a promising approach for achieving high-performance, low-area, and energy-efficient cryptographic systems. Therefore, this work is well-suited for security-oriented applications such as embedded systems, smart cards, IoT devices, and lightweight authentication protocols, where both speed and resource efficiency are critical.

## REFERENCES


1. Anchalya R., Chiranjeevi G. N., Kulkarni S., Efficient Computing Techniques using Vedic Mathematics Sutras, International Journal of Innovative Research in Electrical, Electronic Instrumentation and Control Engineering, 3(5), 24-27, May 2015.
2. Bailey D. V. and Paar C., Efficient Arithmetic in Finite Field Extensions with Application in Elliptic Curve Cryptography, Journal of Cryptology, 14(3), 153-176, (2001).
3. Berlekamp E Algebraic Coding Theory. McGraw-Hill, New York, NY,(1968).
4. Bhaskar R., Hegde G., and Vaya P. R., An efficient hardware model for RSA Encryption system using Vedic mathematics, Procedia Engineering, 30, 124-128, Jan. 2012.
5. Bhattacharya P.B., Jain S.K., and Nagpaul S.R., Basic abstract algebra, Cambridge University Press, United Kingdom, (1995)
6. Bosma W., Lenstra H., Complete Systems of Two Addition Laws for Elliptic Curves. J. Number Theory 53, 229-240 (1995).
7. Chillali A., Elliptic Curve over Ring, International Mathematical Forum, 6(31), 1501-1505, (2011).
8. Chillali A., Hassib M.H. and Elomary M.A., Elliptic curves over a chain ring of characteristic 3, Journal of Taibah University for Science, 9(3), 276-287, (2015).
9. Diffie W. and Hellman M., New directions in Cryptography, IEEE Transactions on Information Theory, 22(6), 644-654, Nov. (1976).
10. Hankerson D., Menzes A., and Vanstone S., Guide to Elliptic Curve Cryptography, Springer-Verlag, New York, 2004.
11. Haveliya A., FPGA implementation of a Vedic convolution algorithm, International Journal of Engineering Research and Applications, 2(1), 678-884, Feb. (2012).
12. Kanhe A., Das S.K. and Singh A.K., Design and implementation of low power multiplier using Vedic multiplication technique, International Journal of Computer Science and Communication, 3(1), 131-132, June (2012).
13. Kapse Y. D., Sarangpure P. R., and Lokhande K. M., Review on a Compressor Design and Implementation of Multiplier using Vedic Mathematics, International Journal of Advanced Research in Computer and Communication Engineering, 6(2), Feb. (2017).
14. Karatsuba A. and Ofman Y., Multiplication of multidigit numbers by automata, Soviet Physics-Doklady, 7, 595-596, (1963).
15. Koblitz N., Elliptic Curve Cryptosystem. Journal of Mathematics Computation, 48(177), 203-209, (1987).
16. Kumar G. G. and Charishma V., Design of high-speed Vedic multiplier using Vedic mathematics techniques, International Journal of Scientific and Research Publications, 2(3), 1-8, March (2012).




Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026

17. Menezes A. And Vanstone S., Elliptic curve cryptosystems and their implementation, Journal of Cryptology, 6, 209-224, (1993).
18. Menezes A. J., Elliptic Curve Public Key Cryptosystems, Kluwer Academic Publishers, Springer, July (1993).
19. Menezes A., Okamoto T. And Vanstone S., Reducing elliptic curve logarithms to logarithms in a finite field, IEEE Transactions on Information Theory, 39, 1639-1646, (1993).
20. Nanda A. and Behera S., Design and Implementation of Urdhva-Tiryakbhyam Based Fast 8×8 Vedic Binary Multiplier, International Journal of Engineering Research & Technology, 3(3), 1856-1859, March (2014).
21. Palata K. N., Nadar V. K., Jethawa J. S., Surwadkar T. J., and Deshmukh R. S., Implementation of an Efficient Multiplier based on Vedic Mathematics, International Research Journal of Engineering and Technology, 4(4), 494-497, April (2017).
22. Pawar A., Sahu A. K., and Sinha G. R., Implementation of High Speed Vedic Multiplier, International Journal of Innovative Research in Advanced Engineering, 1(10), 396-401, Nov. (2014).
23. Poornima M., Patil S. K., Kumar S., Shridhar K. P., and Sanjay H., Implementation of multiplier using Vedic algorithm, International Journal of Innovative Technology and Exploring Engineering, 2(6), 219-223, May.(2013).
24. Sadanandan S. and Anjali V., Design of Advanced Encryption Standard using Vedic Mathematics, International Journal of Innovative Research in Advanced Engineering, 1(6), 322-325, July (2014).
25. Salim S. M. and Lakhotiya S. A., Implementation of RSA Cryptosystem Using Ancient Indian Vedic Mathematics, International Journal of Science and Research 4(5), 3221-3230, May (2015).
26. Sameer G., Sumana M., and Kumar S., Novel High Speed Vedic Mathematics Multiplier using Compressors, International Journal of Advanced Technology and Innovative Research, 7(2), 0244-0248, Feb. (2015).
27. Schoof R., Elliptic curves over finite fields and the computation of square roots mod $p$, Mathematics Computing, 44, 483-494, (1985).
28. Shylashree N., Reddy D. V. N., and Sridhar V., Efficient Implementation of Scalar Multiplication for Elliptic Curve Cryptography using Ancient Indian Vedic Mathematics over GF (p), 49(7), 46-50, July (2015).
29. Stallings W., Cryptography and Network Security: Principles and Practices, Prentice Hall, India, (2003)
30. Tadmori A., Chillali A., and M. Ziane, Elliptic Curve over Ring $A_4$; Applied Mathematics Science, 9, 1721-1733 (2015).
31. Tadmori A., Chillali A., and M. Ziane, Normal Form of the elliptic curves over the finite ring; Journal of Mathematics and system Sience 4, 194-196 (2014).
32. Tadmori A., Chillali A., and Ziane M., Coding over elliptic curves in the ring of characteristic two; International Journal of Applied Mathematics and Informatics, 8, 65-67 (2014).
33. Thapliyal H. and Arbania H.R., A Time-Area-Power Efficient Multiplier and Square Architecture Based On Ancient Indian Vedic Mathematics", Proceedings of the 2004 International Conference on VLSI (VLSI'04), Las Vegas, Nevada, pp. 434-439, June ( 2004).
34. Tirthaji J. S. S. B. K., Vedic Mathematics or Sixteen Simple Sutras from Vedas, Motilal Banarsidas, Varanasi, India, (1986).




Paper received: 02.02.2026 Reviewed: 03.02.2026 Accepted: 07.02.2026 Published: 08.02.2026