International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

# A Low-Area and Time-Efficient Hardware Architecture for Secure Comb Scalar Multiplication Using Recoding

**N.Karthik Kumar Reddy[1], O.Sireesha[2]**

[1]PG Scholar, Dept.of ECE, Srinivasa Institute of Technology & Science, Kadapa.
[2]Assistant Professor, Dept.of ECE, Srinivasa Institute of Technology & Science, Kadapa.
Email: nkarthik358@gmail.com, obulapuramsrieesha@gmail.com

***Abstract –***

Elliptic Curve Cryptography (ECC) is widely used for secure communications, but hardware implementations often face a trade-off between area, speed, and side-channel resistance. This paper proposes a novel low-area, time-efficient hardware architecture for secure comb-based scalar multiplication using advanced recoding techniques. The proposed system introduces three key innovations: 1. Adaptive Comb Recoding (ACR): A dynamic recoding strategy that optimizes the window size and precomputation table based on the input scalar, reducing both the number of point additions and memory footprint. 2. Hybrid Modular Arithmetic Unit (HMAU): Combines interleaved modular multiplication and a low-latency modular inverse algorithm to accelerate point operations while minimizing hardware resources. 3. Dual Randomization Security Layer (DRSL): Integrates Z-coordinate randomization and point blinding to resist simple and differential power analysis (SPA/DPA) without introducing performance overhead. The architecture is fully pipelined and designed for FPGA and ASIC platforms, achieving a significant reduction in area-time product (ATP) while maintaining high security. Simulation and synthesis results for a 256-bit prime field demonstrate up to 30% reduction in hardware slices and 25% lower latency compared to existing comb scalar multiplication architectures. The proposed design is suitable for resource-constrained embedded systems requiring secure an efficient ECC operations.

***Keywords –*** Elliptic Curve Cryptography (ECC), Scalar Multiplication, Recoding Techniques, Low-Area Hardware Design, Side-Channel Attack Resistance (SPA/DPA), Modular Arithmetic.

## 1. INTRODUCTION

Cryptography is the study and design of methods to protect secret information over an insecure channel against adversaries. Cryptographic protocols are imperative to protect files and other information due to the rapid growth of security requirements on the Internet being used as a channel for communication and business in today's society. Billions of people are using the Internet as a tool for communication, e-commerce, internet banking, storage and retrieval of sensitive data from cloud servers, wireless sensors networks, mobile commerce, and many others. Successful deployment of a data communication network depends largely on the network's ability to counter against different unwanted attackers (users), that is, how secure the network system is in the presence of many fraudulent users. Therefore, Suitable cryptographic schemes are essential to meet the growing demands for data security in many different systems, ranging from large servers to small hand-held devices. Many constraints such as computation time, area consumption, flexibility, and security must be considered by network system designers. Different systems have different computing powers, resource limitations, and security requirements. For example, a server needs to complete a large number of tasks in a short duration of time, whereas, a more compact design is required to meet security demands in hand-held devices such as smartphones and smartcards because of their resource limitations. On the other hand cryptanalysis, a reverse operation of cryptography, is the study of methods to break cryptographic systems either by solving the underlying mathematical problem or by exploiting the algorithmic and implementation weaknesses of the crypto-system. With the rapid advancement in technology, cryptanalysis has also flourished. Many new efficient cryptanalysis algorithms and procedures have been figured out to attack cryptographic systems either to reveal sensitive data, to alter sensitive data, or to hack a system to perform a task for which it is not designed for. Thus, security of a system can be compromised at any time. Therefore, the underlying implementation platform must be flexible to adopt new algorithms and security parameters regularly to avoid any security breach. Dedicated hardware architectures are essential to meet the speed requirements of many real-time applications. Dedicated hardware processors have many advantages over the general purpose processor (GPP). For example,

implementation of a crypto scheme on dedicated hardware yields higher performance and lower power consumption results compared to an implementation of the crypto scheme on a GPP. Therefore, in many applications the most computationally intensive tasks are performed on dedicated hardware to boost the overall performance of the systems. However, an implementation on GPP is more flexible than dedicated hardware. Field programmable gate array (FPGA) is a hardware platform that can offer the performance of a dedicated hardware as well as the flexibility of a GPP. FPGA is a hardware platform which provides the flexibility for users to replace a current design with a new one in-house. FPGA has established itself as a suitable platform for implementation of security algorithms. This is due to its short design cycle time, low cost and re-usability which make it a more attractive choice compared to application specific integrated circuits (ASICs). It should be pointed out FPGA offers flexibility at the cost of lower performance and higher power consumption compared to ASIC. Therefore for applications demanding a balance of performance and flexibility, FPGA implementation is recommended. ASIC implementation is preferable in applications where high performance is the only major requirement.

## 2. EC Scalar Multiplier Architectures

Public-key cryptography (PKC) has solved many problems that were previously considered impractical such as key exchange, digital signatures, etc. Most of the PKC protocols are based on two efficient schemes: RSA and elliptic curve cryptography (ECC). Recommendations by different standards indicate that 256-bit ECC implementation is capable of providing an equivalent security in comparison to 3072-bit RSA. This gap of the required number of bits in ECC and RSA is expected to increase further in future due to higher security demands. Therefore, due to the much smaller key sizes for same level of security, the ECC based crypto-systems are better in terms of bandwidth utilization, power consumption, and implementation cost as compared to the traditional RSA based crypto-systems. ECC is particularly useful in resource constrained devices because ECC requires lower implementation and transmission cost and thus lower power consumption. ECC will find applications in the Internet of thing, where more and more resource constrained devices will be connected to the Internet.

***EC Scalar Multiplier Architecture in Affine Coordinates***

This section presents an architecture to compute the EC scalar multiplication operation using affine coordinates. It consists of three modular arithmetic units, a modular adder/subtractor (A/S), a modular multiplier (Mul) and a modular divider (Div). A/S unit performs either modular addition or subtraction operation at a time in a single clock cycle, while the Div unit takes 2n clock cycles to compute an n-bit modular division operation. The EC scalar multiplier performance results are shown for different field sizes. The architecture in Figure 1 also consists of instruction memory, register file and a control unit. The instruction memory is loaded with micro-coded instructions, the register file is responsible for storing intermediate and final results while the control unit generates the necessary control signals to execute the required operations. The register file consists of two separate sets of registers, dedicated purpose registers (DPR) and general purpose registers (GPR). The scheduling of finite field operations to perform EC PD and PA operations in affine coordinates are given in Table 1 and 2 respectively. It is worth mentioning using the DA method EC PD and PA operations cannot be executed concurrently. It is also visible from Table 1 that scope of parallelism inside these operations are also very limited, therefore, the EC scalar multiplier architecture incorporated a single A/S, Mul and Div units as shown in Figure 1.

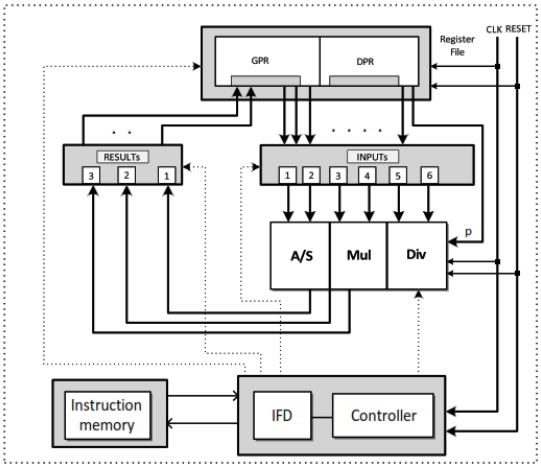


Fig.1: EC scalar multiplier architecture using affine coordinates

Table.1: EC point operations using affine coordinates

| Point addition, **(PA)** | Point doubling, **(PD)** | No of field operations **(FOP)** |
|---|---|---|
| $x_3=\lambda^2-(x_1+x_2)$<br>$Y_3=\lambda(x_1-x_3)-y_1$<br>$\lambda=(y_2-y_1)/(x_2-x_1)$ | $x_3=\lambda^2-(x_1+x_1)$<br>$y_3=\lambda(x_1-x_3)-y_1$<br>$\lambda=(3x_1^2+a)/2y_1$ | **PA** = 6A+2M+1D<br><br>**PD** = 8A+3M+1D |

Normally, to achieve a better performance of EC point multiplication on dedicated hardware, multiple copies of modular adder, subtractor, multiplier and divider units are integrated. These multiple copies can help to execute several operations in parallel at the expense of extra area and cost. As mentioned before when using the DA method, EC PD and PA operations can not be computed in parallel. The scope of parallelism in affine coordinates is also very limited, therefore integration of multiple arithmetic units can not be fully exploited to achieve a significant performance increase.

Table 2: Scheduling of PD operation in affine coordinates

| A/S | Mul | Div |
|---|---|---|
| $A_1 = y_1 + y_1$<br>$A_2 = x_1 + x_1$ | $M_1 = x_1 \times x_1$ | |
| $A_3 = M_1 + M_1$<br>$A_4 = A_3 + M_1$<br>$A_5 = A_4 + a$ | | |
| | | $D_1 = A_5 \; \Xi A_1$ |
| | $M_2 = D_1 \times D_1$ | |
| $A_6(x_3) = M_2 - A_2$<br>$A_7 = x_1 - A_6$ | | |
| | $M_3 = A_7 \times D_1$ | |
| $A_8(y_3) = M_3 - y_1$ | | |

However, DAA provides a flexibility to compute EC PD and PA operations concurrently. Therefore, this work integrates two instances of the arithmetic unit (AU1 and AU2) to execute EC PD and PA operations as shown in Figure 2.

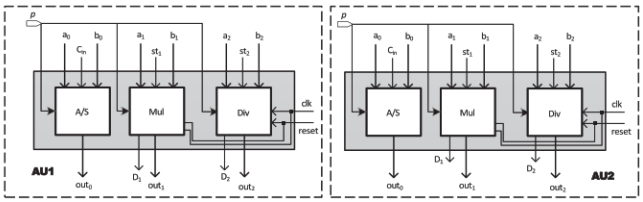


Fig.2: Arithmetic units for parallel execution of PD and PA operations

## 3. IMPLEMENTATION AND RESULTS

The EC scalar multiplier architectures are coded in Verilog (HDL). Xilinx ISE 14.1 design suite is used for synthesis, mapping, placement, and routing purposes and Xilinx ISim simulator is used for simulation purposes. The R4PIM modular multiplier described and implemented is used. The EC points are represented in standard projective coordinates and the EC group operations are computed over standard form of an EC. It depicts latencies of EC PA, PD and scalar multiplication operation against different field sizes when computed using DA and DAA methods by incorporating different number of parallel multiplier units. During synthesis Xilinx Virtex-6 device is selected as the target implementation platform. The available fast carry chains of Xilinx FPGA are utilized to perform addition and subtraction operations. On the top level, the EC scalar multiplier is based

Table.3: Implementation results of EC scalar multiplier in projective coordinates on the DA and DAA algorithms.

| Method | Multipliers | $n$ | Area (slices) | ATB | TP (ops) |
|---|---|---|---|---|---|
| DA | 2 | 192 | 5251 | 29.5 | 925.9 |
| | | 224 | 6048 | 40.5 | 666.6 |
| | | 256 | 7089 | 55.6 | 497.5 |
| | 3 | 192 | 6952 | 32.9 | 1098.9 |
| | | 224 | 8108 | 45.6 | 793.6 |
| | | 256 | 9372 | 61.8 | 591.7 |
| | 4 | 192 | 8731 | 35.9 | 1265.8 |
| | | 224 | 10, 115 | 49.6 | 909 |
| | | 256 | 11, 655 | 66.9 | 680.3 |
| DAA | 2 | 192 | 5432 | 42.7 | 666.6 |
| | | 224 | 6341 | 59.4 | 476.2 |
| | | 256 | 7235 | 79.1 | 357.1 |
| | 3 | 192 | 7113 | 38.15 | 970.8 |
| | | 224 | 8341 | 53.6 | 694.4 |
| | | 256 | 9588 | 71.9 | 520.8 |
| | 4 | 192 | 8897 | 36.14 | 1282 |
| | | 224 | 10, 291 | 50.07 | 917.4 |
| | | 256 | 11, 791 | 67.24 | 684.9 |

The DAA technique offers inherent protection against timing and simple power analysis attacks. The presented designs are programmable for various field sizes of p $\leqslant$ 256-bit. It is obvious that incorporating more multiplier units reduces the computation time of an EC scalar multiplication operation at the cost of more hardware resources. Note that in the case of using four parallel multiplier units computation time of an EC scalar multiplication operation using DA and DAA methods are comparable.

## CONCLUSION

The contribution of this research work is mainly comprises of efficient hardware architectures for finite arithmetic operations including addition, subtraction, multiplication, and division. Based on these optimized arithmetic primitives, high performance hardware architectures for elliptic curve scalar multiplication operation are proposed. As the computational complexity of a modular division operation is more than a modular multiplication, the standard projective coordinates are used to remove this division operation from the computation of EC group operations. At the end of an EC scalar multiplication operation in projective coordinates, two division operations are needed to convert the result back to affine coordinates. This work uses the standard projective coordinate system to perform these point operations. In projective coordinates, there are many opportunities to execute the underlying field operations in parallel. Therefore, the EC scalar multiplier using standard projective coordinates employs different numbers of parallel multipliers. A performance evaluation is presented based on computation time, resource consumption and throughput. Using the DA method, performance is evaluated by employing three and four parallel multipliers and for the DAA method, performance is shown for the architectures using four modular multipliers. Finally, comparison with state-of-the-art designs in the literature is presented which shows that the

architectures proposed in this work are good trade- offs between performance and flexibility. The presented designs provide the flexibility for the user to select a prime number and the EC parameters. Therefore, the user can update these parameters to avoid any security breach.

## REFERENCES

[1] V. S. Miller, "Use of elliptic curves in cryptography," in Advances in Cryptology–CRYPTO85 Proceedings. Springer, 1986, pp. 417–426.

[2] N. Koblitz, "Elliptic curve cryptosystems," Mathematics of computation, vol. 48, no. 177, pp. 203–209, 1987.

[3] Mahimalur, R. K., Vasgam, M., & Manoharan, D. Devops Lifecycle Management And Cloud Migration Assessments: A Security-Driven CICD Perspective.

[4] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and Public-Key cryptosystems," Communications of the ACM, vol. 21, pp. 120–126, 1978.

[5] Maturi, S. Y. (2025). Vulnerabilities in the 802.11 Wireless Client Selection Mechanis.

[6] Doragacharla, V. R. (2023). Comprehensive Benchmarking Analysis of Auto Scaling Approaches in Cloud Native Streaming Pipelines During Flash Sales and Holiday Traffic Peaks. Available at SSRN 6566479.

[7] NIST, "Data encryption standard (DES), FIPS (46-3)," National Institute of Standards and Technology, vol. 25, no. 10, pp. 1–22, 1999.

[8] Manoharan, D. (2024). Governance-Oriented Quality Engineering Framework for Healthcare EDI Modernization. International Journal of Multidisciplinary on Science and Management IJMSM, 1(2).

[9] Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.

[10] "Advanced encryption standard (AES), FIPS (197)," National Institute of Standards and Technology, vol. 197, pp. 441–0311, 2001.

[11] Sikder, M. Z., Shakil, M. A. I., Ahad, A., Karim, M. F., Intakhab, B., & Islam, D. A. (2025, June). Microwave-Based Detection of Early-Stage Renal Cell Carcinoma Using UHF Range Antenna. In 2025 International Conference on Computer Systems and Technologies (CompSysTech) (pp. 1-6). IEEE.

[12] J. Daemen and V. Rijmen, The design of Rijndael: AES-the advanced encryption standard. Springer Science & Business Media, 2013.

[13] Purmani, S. S. R. (2024). Aligning IT investment decisions with overall business strategy from an enterprise program management perspective, focusing on the integration of IT leadership in strategic decision-making processes. International Journal of Communication Networks and Information Security, 16(5), 1213–1219.

[14] Manoharan, D. (2026). Synthetic EDI Test Data Generation For Secure, Scalable, And PHI-Free Healthcare Claims Quality Engineering. Journal of International Crisis and Risk Communication Research, 9(1).

[15] Maturi, S. Y. Cryptographic Privacy Engines: Practical Multi-Party Protocols For Confidential Database Queries.

[16] Poojari, R. (2024). Empirical Analysis of Context Window Enhancement Methods in Retrieval-Augmented Generation Models. Journal of Computational Analysis and Applications, 33(2).

[17] Poojari, R. (2024). Assessing Clinical Natural Language Processing (NLP) Models for Interpreting Electronic Health Records (EHR): Focus on Accuracy, Bias, and Generalizability.

[18] Ravishankara, M. (2026, February). PlotChain: Deterministic Checkpointed Evaluation of Multimodal LLMs on Engineering Plot Reading. In SoutheastCon 2026 (pp. 1-8). IEEE.

[19] Purmani, S. S. R., & Kotte, G. Intelligent Project Orchestration: How Generative AI is Reshaping Go-to-Market Strategy Planning and Cross-Functional Delivery. environments, 4, 5.

[20] B. Schneier, Applied cryptography: protocols, algorithms, and source code in C. john wiley & sons, 2007.

[21] W. Diffie and M. E. Hellman, "New directions in cryptography," Information Theory, IEEE Transactions on, vol. 22, no. 6, pp. 644–654, 1976.

[22] Mudusu, S. K. (2024, August). Designing self-healing data pipelines for autonomous and continuous AI operations. Journal of Computational Analysis and Applications, 33(2), 1238–1247.

[23] J. Katz and Y. Lindell, Introduction to modern cryptography. CRC Press, 2014.

[24] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, "NIST special publication 800-57," NIST Special Publication, vol. 800, no. 57, pp. 1–142.