

VLSI Implementation Of A Chua Chaotic System-Based Secure Image Encryption Architecture

Sarala B¹, Kousalya R², Jeyaprabha TJ³,

^{1,2,3}*Department of Electronics and Communication Engineering, Sri Venkateswara College of Engineering, Chennai, India*

Abstract—This paper presents a high-performance FPGA-based architecture for secure image encryption employing a chaos-driven cryptographic technique. Traditional image encryption approaches often encounter limitations, including restricted key space, vulnerability to statistical and differential attacks, and reduced processing efficiency for high-resolution images. To address these challenges, the proposed architecture integrates the nonlinear dynamics of Chua's chaotic system with a hardware-efficient stream cipher to enhance encryption security and computational performance. Initially, chaotic sequences are generated using Chua's circuit, numerically implemented through the fourth-order Runge-Kutta method with Q12.12 fixed-point precision.

The resulting chaotic sequences are subsequently combined with a Linear Feedback Shift Register (LFSR) to produce pseudorandom keystreams suitable for efficient hardware realization. A reversible XOR-based diffusion mechanism is incorporated to strengthen the encryption process while ensuring lossless recovery of the original image during decryption. The proposed architecture employs a fully pipelined design capable of processing one pixel per clock cycle, making it suitable for real-time image encryption applications. The cryptographic strength of the system is assessed using statistical and differential security measures, including information entropy, adjacent-pixel correlation, Number of Pixels Change Rate (NPCR), and Unified Average Changing Intensity (UACI). Experimental observations indicate favorable entropy characteristics, minimal correlation between neighboring encrypted pixels, and strong differential sensitivity. Furthermore, the decryption process achieves exact reconstruction of the original image without pixel-level errors. These characteristics demonstrate the potential of the proposed FPGA architecture for secure, high-throughput image processing applications.

Index Terms—Chaotic encryption, Chua's circuit, RK4 solver, FPGA implementation, image security, LFSR

I. INTRODUCTION

The increasing use of digital images in communication systems has created a strong demand for secure and efficient image encryption techniques. Conventional cryptographic algorithms such as AES are not well suited for image data due to high pixel correlation, redundancy, and large data size, leading to performance inefficiencies in real-time applications. Chaos-based encryption has emerged as a promising alternative because of its sensitivity to initial conditions and pseudo-random behaviour. In particular, Chua's circuit provides strong nonlinear dynamics suitable for secure key generation.

However, practical hardware implementation of chaotic systems faces challenges such as computational complexity and limited throughput. Numerical methods like the Runge-Kutta method improve precision but introduce latency, making direct use of chaotic sequences inefficient for high-speed image processing. Additionally, many existing designs lack proper synchronization and efficient diffusion mechanisms. To address these issues, this paper proposes a hardware-efficient image encryption architecture that combines chaotic key generation with a high-speed Linear Feedback Shift Register. The system employs a fixed-point RK4 implementation of Chua's system and a reversible diffusion scheme, enabling real-time processing with one pixel per clock cycle. Experimental results demonstrate strong security and perfect reconstruction.

II. RELATED WORK

A. Existing methods

Several image encryption techniques have been proposed using both conventional cryptographic

algorithms and chaos-based systems. Traditional methods such as AES provide strong security but are not optimized for image-specific characteristics like high redundancy and spatial correlation. To address this, researchers have explored chaos-based encryption techniques using systems such as Logistic maps, Lorenz systems, and Chua's circuit.

Recent works have also incorporated numerical solvers such as the Runge-Kutta method to improve the precision of chaotic sequence generation. Additionally, pseudo-random generators like Linear Feedback Shift Register have been used to enhance throughput in hardware implementations.

B. Limitations

Despite these advancements, existing approaches suffer from several limitations:

- 1) High computational complexity of chaotic systems limits real-time hardware implementation.
- 2) Low throughput due to direct use of chaotic sequences without efficient expansion.
- 3) Weak resistance to statistical and differential attacks in some designs.
- 4) Lack of proper pipeline synchronization in FPGA implementations.
- 5) Increased hardware resource utilization due to floating-point computations.

C. Research Gap

From the literature, it is evident that there is a need for an efficient architecture that balances security, speed, and hardware complexity.

Specifically, there is a gap in designing a system that:

- Uses high-precision chaotic systems while maintaining hardware efficiency.
- Achieves real-time performance through pipelined processing.
- Provides strong resistance to statistical and differential attacks.
- Integrates chaos-based key generation with high-speed expansion techniques.

This paper addresses these gaps by proposing a hybrid FPGA-based architecture that combines chaotic dynamics with efficient hardware design techniques to achieve secure and real-time image encryption.

III. PROPOSED SYSTEM

The proposed system implements a fully pipelined image encryption architecture designed for real-time FPGA operation. It consists of four major modules: chaotic key generation, key expansion, diffusion-based encryption, and memory interface. The input grayscale image is stored in ROM and processed sequentially in a streaming manner. The chaotic key generation unit produces high-precision states using a nonlinear system, which are then expanded into a high-speed key stream. This key stream is synchronized with incoming pixel data and fed into the diffusion module. The encrypted pixels are stored in dual-port RAM, enabling simultaneous write (encryption) and read (display) operations. Careful pipeline alignment ensures that data and key signals are synchronized, achieving a throughput of one pixel per clock cycle.

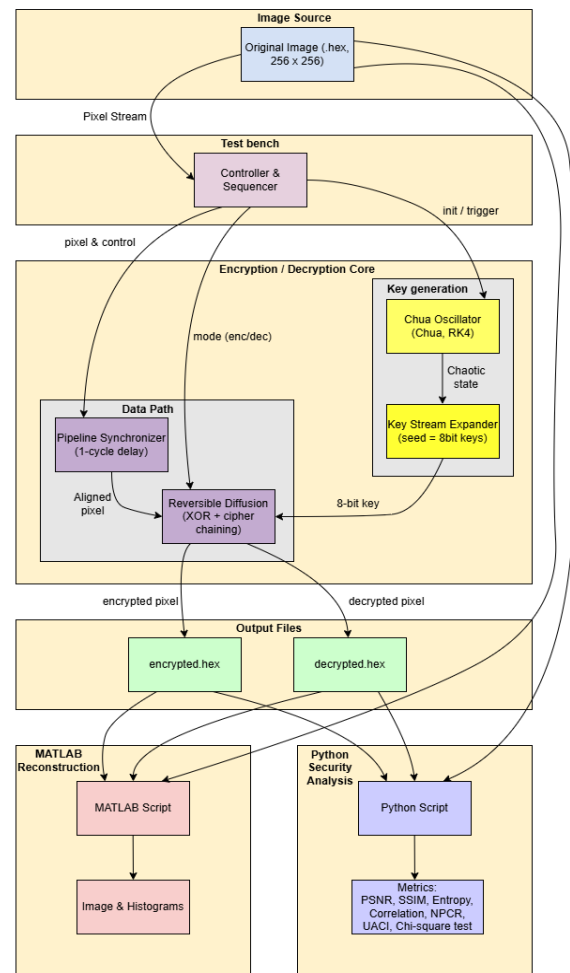


Fig. 1. Top-Level Architecture of the Proposed Chaotic Image Encryption System and Analytical Workflow

A. Chua Chaotic System

The system utilizes Chua's circuit as the entropy source due to its strong nonlinear dynamics and sensitivity to initial conditions.

The governing equations are:

$$\frac{dx}{dt} = \alpha(y - x - f(x)) \quad (1)$$

$$\frac{dy}{dt} = x - y + z \quad (2)$$

$$\frac{dz}{dt} = -\beta y \quad (3)$$

where $f(x)$ is a piecewise nonlinear function defined as:

$$f(x) = m_1 x + \frac{1}{2}(m_0 - m_1)(|x + 1| - |x - 1|) \quad (4)$$

This nonlinear characteristic introduces chaotic behavior, making the generated sequences highly unpredictable and suitable for cryptographic applications.

B. RK4 Solver Implementation

To numerically solve the continuous-time chaotic system, the Runge-Kutta method is employed. The RK4 method computes intermediate slopes k_1, k_2, k_3, k_4 and updates the state as:

$$x_{n+1} = x_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (5)$$

A similar formulation is used for y and z . The solver is implemented using fixed-point Q12.12 representation to balance precision and hardware efficiency. A finite state machine (FSM) controls the sequential computation of RK4 stages, completing one iteration in multiple clock cycles. This design ensures deterministic operation while maintaining compatibility with FPGA resources.

C. Key Generation Module

Since the RK4 solver produces chaotic values at a lower rate, a key expansion mechanism is required to match the pixel processing speed. A Linear Feedback Shift Register is used to expand each chaotic seed into multiple 8-bit keys.

The chaotic outputs x, y, z are combined and mixed to initialize a 32-bit LFSR. The LFSR then generates a sequence of pseudo-random keys using a primitive polynomial, enabling continuous key generation at one key per clock cycle. A batch-based approach is used, where one chaotic seed generates multiple keys, ensuring high throughput while preserving randomness.

D. Encryption Process (Confusion + Diffusion)

The encryption process is based on a combination of confusion and diffusion principles. Confusion is achieved through the use of chaotic keys, while diffusion is implemented using a feedback-based XOR operation.

For each pixel $p(i)$, the encrypted pixel $c(i)$ is computed as:

$$c(i) = p(i) \oplus k(i) \oplus c(i-1) \quad (6)$$

where $k(i)$ is the key generated by the key expander and $c(i-1)$ is the previous cipher pixel. This feedback mechanism ensures that a small change in input propagates across the entire image, enhancing security against differential attacks.

E. Reversible Logic Design

The proposed encryption scheme is inherently reversible, allowing the same hardware architecture to perform both encryption and decryption.

The decryption process is defined as:

$$p(i) = c(i) \oplus k(i) \oplus c(i-1) \quad (7)$$

By controlling the feedback path, the system switches between encryption and decryption modes without requiring additional hardware. This reversible design reduces resource utilization and ensures lossless recovery of the original image. The correctness of the design is validated through simulation, demonstrating zero-pixel mismatch after decryption.

IV. HARDWARE IMPLEMENTATION

A. Target FPGA Platform

The proposed architecture is designed targeting the Altera DE10-Lite FPGA development board, which is based on the Intel MAX 10 series device. The platform provides a suitable environment for implementing real-time image processing systems due to its embedded memory resources, logic elements, and on-chip peripherals.

Although the current work focuses on RTL design and functional verification through simulation, the architecture is optimized considering FPGA constraints such as limited hardware resources, memory bandwidth, and clock frequency. The modular design ensures compatibility with the DE10-Lite platform for future hardware deployment.

B. Fixed-Point Representation

To achieve hardware efficiency, the chaotic system is implemented using fixed-point arithmetic instead of floating-point representation. A Q12.12 format (24-bit signed) is used, consisting of 1 sign bit, 11 integer bits, and 12 fractional bits.

This representation provides a good balance between numerical precision and hardware complexity. Fixed-point operations significantly reduce resource utilization and power consumption while maintaining sufficient accuracy for chaotic behaviour. Bit slicing is used after multiplication operations to preserve the required precision and avoid overflow.

C. RTL Design

The entire system is described at the Register Transfer Level (RTL) using Verilog.

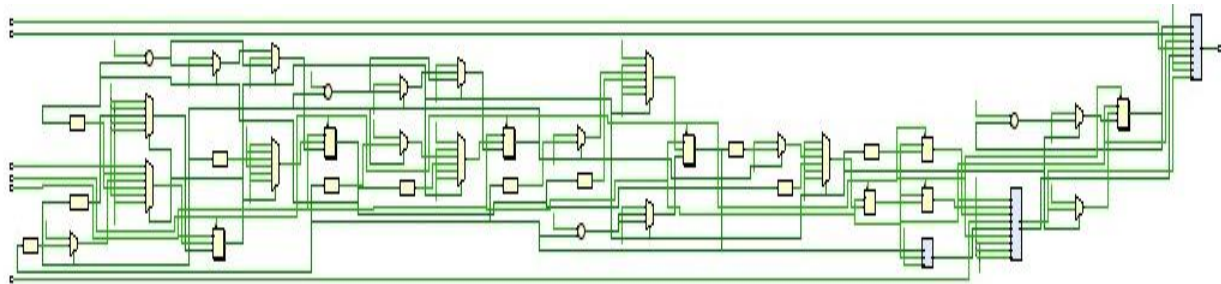


Fig. 2. RTL Schematic of Chaos-Based Encryption Architecture

D. FSM Design

Finite State Machines (FSMs) are used to control sequential operations within the system, particularly in the RK4 solver and image pipeline.

The RK4 core uses a multi-state FSM to compute intermediate values k_1, k_2, k_3, k_4 across multiple clock cycles.

Each state corresponds to a specific computation stage, ensuring deterministic execution.

Similarly, the top-level control unit manages image reading, key generation synchronization, and processing flow. The FSM-based design enables efficient scheduling of operations while maintaining pipeline consistency.

E. Pipelining and Parallelism

The proposed architecture is fully pipelined to achieve high throughput. Pipeline registers are introduced to align data paths and control signals, particularly between ROM output, key generation, and diffusion stages.

The design is modular and consists of the following key components:

- Chaotic RK4 computation core
- Key expansion module
- Diffusion (encryption/decryption) core
- Image ROM and dual-port RAM interface

Each module is designed independently and integrated using well-defined interfaces. The design follows synchronous logic principles with a global clock and active-low reset.

Handshaking signals such as valid, start, and busy are used to ensure proper data flow and synchronization between modules.

A key design consideration is the synchronization of pixel data with the generated key stream. Since the key expander introduces a one-cycle delay, the input pixel is also delayed by one clock cycle to ensure proper alignment. This technique eliminates timing mismatches and enables continuous data streaming.

Parallelism is exploited at the system level through independent module operation, allowing simultaneous key generation, pixel processing, and memory access. As a result, the architecture achieves a throughput of one pixel per clock cycle after initial latency.

F. Resource Utilization (Estimated)

The proposed design is optimized for efficient hardware utilization by leveraging fixed-point arithmetic and modular design principles. Multipliers are primarily used in the RK4 computation, while XOR-based operations dominate the diffusion stage, resulting in low logic overhead.

Although synthesis results are not yet reported, the design is expected to fit within the available logic

elements and embedded memory resources of the target FPGA. The use of LFSR-based key expansion minimizes computational complexity compared to direct chaotic sequence generation.

Future work will include synthesis and implementation on the target FPGA to obtain detailed metrics such as logic utilization, power consumption, and maximum operating frequency.

V. EXPERIMENTAL RESULTS

A. Simulation Setup (Verilog testbench / MATLAB)

The proposed encryption architecture is validated using a Verilog testbench and MATLAB-based post-processing. A 256×256 grayscale image is converted into hexadecimal format and stored in a ROM module. The RTL design processes the image in a streaming manner, generating encrypted and decrypted outputs as encrypted.hex and decrypted.hex files.

The simulation is carried out with proper synchronization between the chaotic key generator, key expansion module, and diffusion core. The generated output files are imported into MATLAB for image reconstruction and evaluation. Various security and quality metrics are computed to assess the effectiveness of the proposed system.

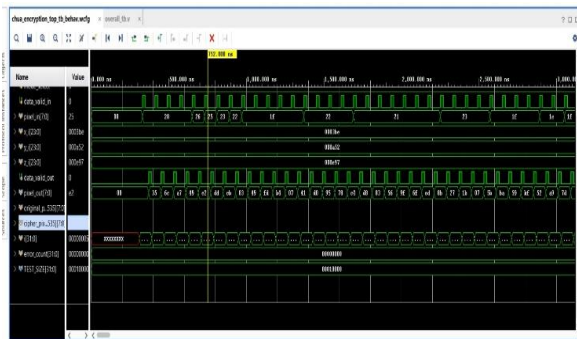
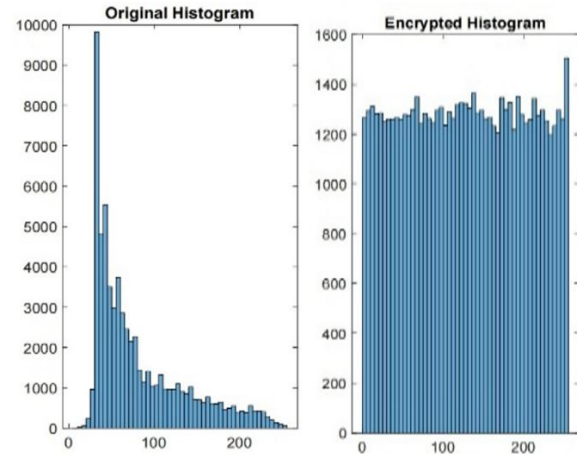
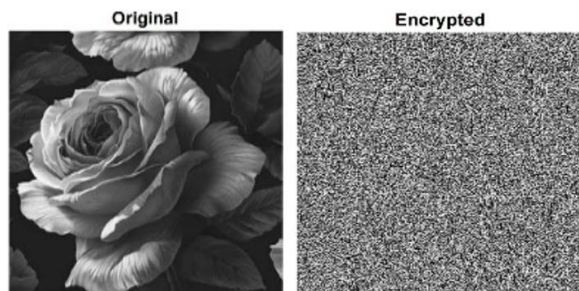


Fig. 3. Waveform results & Validation

B. Encryption Results



The encrypted image appears as random noise with no visible patterns from the original image. This indicates that the proposed system effectively eliminates pixel correlation and structural information.

The histogram of the encrypted image is uniformly distributed, demonstrating strong resistance against statistical analysis attacks.

C. Decryption Results



The decrypted image is identical to the original input image, confirming the correctness of the reversible encryption scheme.

The system achieves zero-pixel mismatch, ensuring lossless recovery and validating the accuracy of the proposed design.

D. Performance Metrics

The performance of the proposed system is evaluated using standard image encryption metrics, including Peak Signal-to-Noise Ratio, entropy, correlation, NPCR, and UACI.

TABLE I. PERFORMANCE EVALUATION METRICS

Metric	Value	Interpretation
PSNR (Encryption)	27.901 dB	Low similarity to original → effective encryption
PSNR (Decryption)	100 dB	Perfect reconstruction
SSIM (Decryption)	1.000	No distortion (lossless recovery)
Entropy (Encrypted)	7.9972	Very close to ideal (8) → excellent randomness
Correlation (Horizontal)	- 0.00109	$\approx 0 \rightarrow$ no pixel correlation
Correlation (Vertical)	- 0.00187	$\approx 0 \rightarrow$ no pixel correlation
Correlation (Diagonal)	0.00054	$\approx 0 \rightarrow$ no pixel correlation
NPCR	100.0 %	Strong avalanche effect
UACI	50.24 %	High intensity variation (strong diffusion)
Chi-Square	253.17	Close to uniform distribution

VI. CONCLUSION

This paper presented a hardware-efficient image encryption architecture based on chaotic dynamics and reversible logic. The proposed system integrates a fixed-point implementation of Chua's circuit using the Runge-Kutta method with a high-speed Linear Feedback Shift Register to generate a robust and continuous key stream. A feedback-based diffusion mechanism was employed to achieve strong confusion and diffusion while enabling lossless decryption using the same hardware architecture.

The proposed design achieves one-pixel-per-clock throughput through efficient pipelining and synchronization. Experimental results demonstrate strong security performance, including high entropy (7.9972), near-zero correlation among adjacent pixels, and a strong avalanche effect with NPCR reaching 100% and UACI approximately 50.24%, indicating enhanced sensitivity to slight changes in input. The encrypted image exhibits low similarity to the original, while the decrypted image achieves perfect reconstruction with a PSNR of 100 dB and SSIM equal to 1, validating the correctness and reversibility of the proposed design.

The key achievements of this work include the development of a hybrid chaotic-LFSR key generation scheme, an efficient fixed-point hardware

implementation, and a fully pipelined architecture suitable for real-time applications. Future work will focus on complete FPGA implementation on the target platform, detailed resource and power analysis, and extension of the design to color image encryption and video stream processing.

REFERENCES

- [1] M. Maazouz *et al.*, "FPGA implementation of a chaos-based image encryption algorithm," *Alexandria Eng. J.*, vol. 61, no. 2, pp. 1201–1214, 2022.
- [2] T. Umar *et al.*, "Chaos based image encryption scheme using modified skew tent map," *Expert Syst. Appl.*, vol. 234, 2024.
- [3] T. M. Hoang *et al.*, "Novel structures of chaos-based parallel multiple image encryption and FPGA implementation," *Sci. Rep.*, vol. 16, 2026.
- [4] C. H. Yang *et al.*, "FPGA implementation of image encryption using chaotic system," *Electronics*, vol. 14, no. 4, 2025.
- [5] X. Wang and Y. Li, "Chaotic image encryption algorithm based on hybrid optimization and DNA sequence," *Opt. Lasers Eng.*, vol. 137, 2021.
- [6] M. Gafsi *et al.*, "Optimized FPGA implementation of a chaos-based cryptosystem for real-time image encryption," *Int. J. Bifurcation Chaos*, vol. 31, no. 14, 2021.
- [7] F. Ciylan *et al.*, "FPGA-based chaotic image encryption using systolic array architecture," *Electronics*, vol. 12, no. 12, 2023.
- [8] S. S. Moghimi Moghaddam *et al.*, "Parallel chaos-based image encryption algorithm for high-speed applications," *J. Supercomput.*, 2024.
- [9] N. H. Abdelzaher *et al.*, "Efficient FPGA-based multi-image encryption using hyperchaotic systems," *Microelectron. J.*, 2026.
- [10] S. Jing-yu *et al.*, "FPGA image encryption and steganography using a novel chaotic system," *Digit. Signal Process.*, vol. 134, 2023.