International Journal of Innovative Research and Scientific Studies, 8(5) 2025, pages: 2140-2150

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

Hybrid software-hardware image encryption techniques for network security using multiple chaotic maps

Fadwa Al Azzo[1], Aamer T. Suhail[2*], Harith G. Ayoub[3], Zaid.A. Abdulrazzaq[4]

[1,2,3,4]*Norther Technical University (NTU) Iraq.*

Corresponding author: Aamer T. Suhail (*Email: aamir@ntu.edu.iq*)

Abstract

The security of digital data transmission has become a primary concern in this century, particularly in the field of information data communication. Encryption is the process of transforming information to prevent unauthorized access and plays a critical role in ensuring data security. The two proposed techniques support four distinct Chaos Pseudo Random Bit Generators (PRBGs): Lozi map, Tent map, Logistic map, and Quad map. The image is split into four parts, each encrypted using one of the mentioned PRBG maps, thereby enhancing the complexity and security of the encryption system. The randomness of the generated cryptographic keys was tested using the National Institute of Standards and Technology (NIST) Statistical Test Suite for evaluating Pseudorandom Number Generators for Cryptographic Applications. The overall encryption system was implemented on a Field Programmable Gate Array (FPGA) ZYNQ702 evaluation board hardware device. The results were compared with those of other researchers to evaluate improvements in the NIST Statistical Test Suite outcomes relative to existing methods, aiming to identify the most effective encryption approach. The main contributions of this paper include two systems offering more secure and effective methods, supported by comparisons with other studies, thereby advancing secure data transmission using FPGA technology at a frequency of 667 MHz and a throughput of 5.3 Gbps.

Keywords: Chaos Theory, FPGA, NIST, PRBG, Security, XSG.

DOI: 10.53894/ijirss.v8i5.9436
Funding: This study received no specific financial support.
History: Received: 26 June 2025 / **Revised:** 30 July 2025 / **Accepted:** 1 August 2025 / **Published:** 22 August 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

In recent years, the rapid evolution of communication technologies such as mobile devices and internet networks has broadened the scope of information transmission. However, this expansion has also introduced new challenges in

protecting multimedia messages from unauthorized interception during transmission. As such, it has been important to cipher data such as images and videos to block unauthorized entry and ensure reliable communication over the internet. For secure transmission objectives, many methods have been suggested to achieve the required security goals [1-3]. The proposed algorithm provides more security efficiency and performance in various tests [4, 5]. The proposed image cryptography in the Zhang and Wang [6] scheme-based zigzag.

Transform and bit-level encryption enhance a complex design of the ciphered image, making it more resistant to cryptanalysis attacks. The proposed algorithm Shi, et al. [6] introduced an image encryption method with a chaotic system-based Boson sampling, presenting more valuable security efficiency than providing suitable, reliable image communication. In Chai, et al. [7] proposed a novel image cryptography technique based on DNA sequences with chaotic systems, enhancing the efficiency of image encryption security performance. The proposed [8] introduced a new image cryptography technique based on chaotic systems supported with DNA mapping. The proposed method provides more secure image transmission.

The proposed method involves dividing the image into multiple segments, each of which is encrypted independently using a distinct chaotic map type with Pseudo-random number techniques (PRNG), thereby enhancing security through differential encryption across segments. The security analysis, compared with other recent research, demonstrates the strength of the proposed approach.

FPGA, an abbreviation for field-programmable gate array, is an efficient hardware technique used for multiple applications, including embedded image processing devices such as cameras. Supported by the Xilinx System Generator (XSG) tool within the MATLAB-SIMULINK environment, the image encryption model is executed in real-time, allowing for immediate access to the results.

2. Material and Methods

Proposed image encryption system is illustrated in Figure 1. Starting by splitting the original image into four images: X1, X2, X3, and X4. Each one of these images goes through a particular, unique chaotic system method key. The results of these processes were four ciphered images denoted by C1, C2, C3, and C4, corresponding to their respective parts of the original image. Then, the final ciphered image is produced by assembling the four encrypted images C1, C2, C3, and C4.

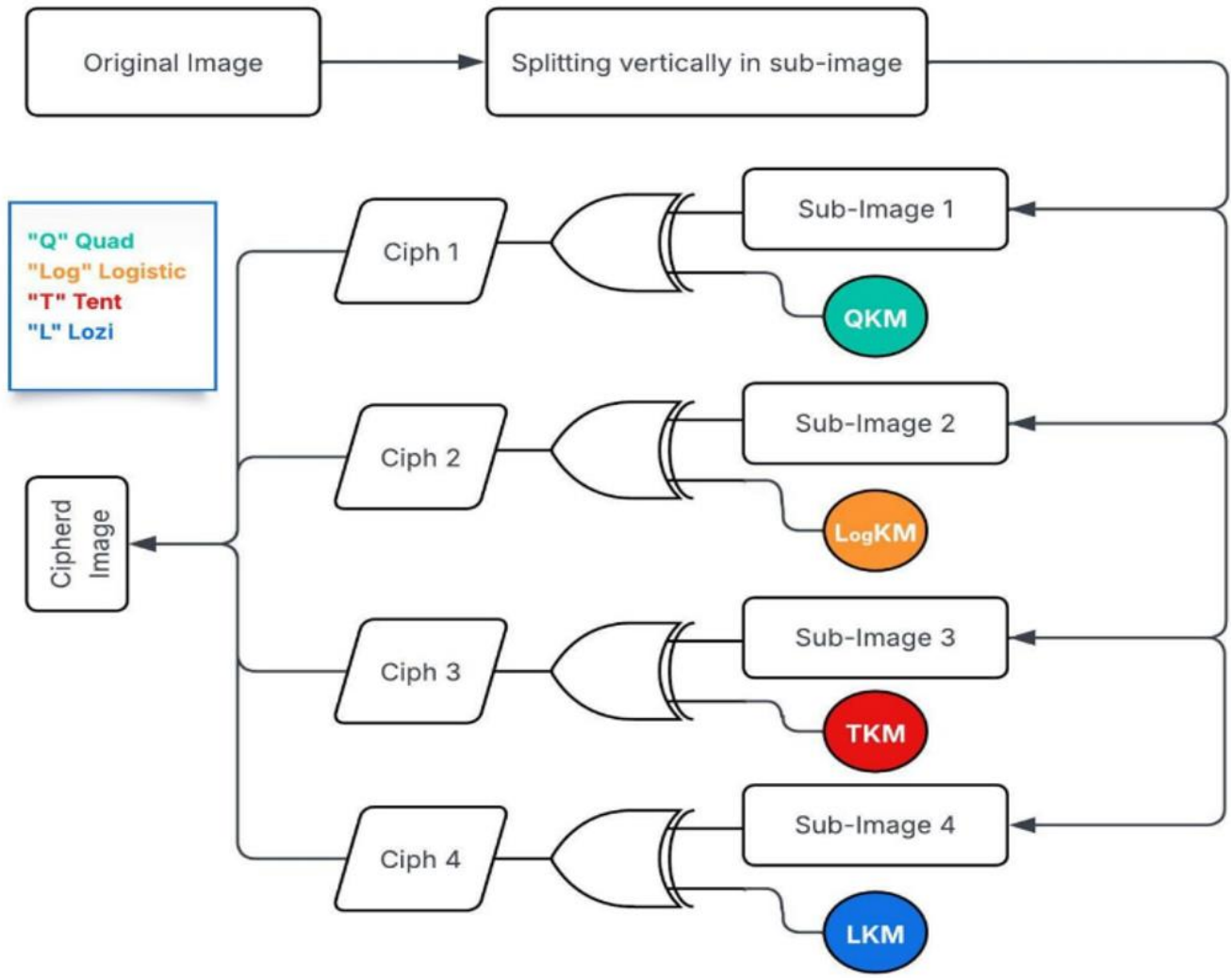

Figure 1.
Proposed image encryption.

Splitting of the original image into four segments (X1, X2, X3, and X4) was achieved using two methods: the first one illustrated in Figure 2 and Equation 1. The splitting process started with adding the first pixel of the original image and assigning it to the X1 image, then the second one to the X2 image, then the third one to the X3 image, then the fourth one to the X4 block, and so on, all in a vertical manner.

$$x(k, i) = Ima(k, (x(i - 1) * 4) + n) \quad (1)$$

Where:

$1 \leq i \leq 64$: Index variable for the column.

$1 \leq k \leq 256$: Index variable for the row.

$1 \leq n \leq 4$: indication for a new image.

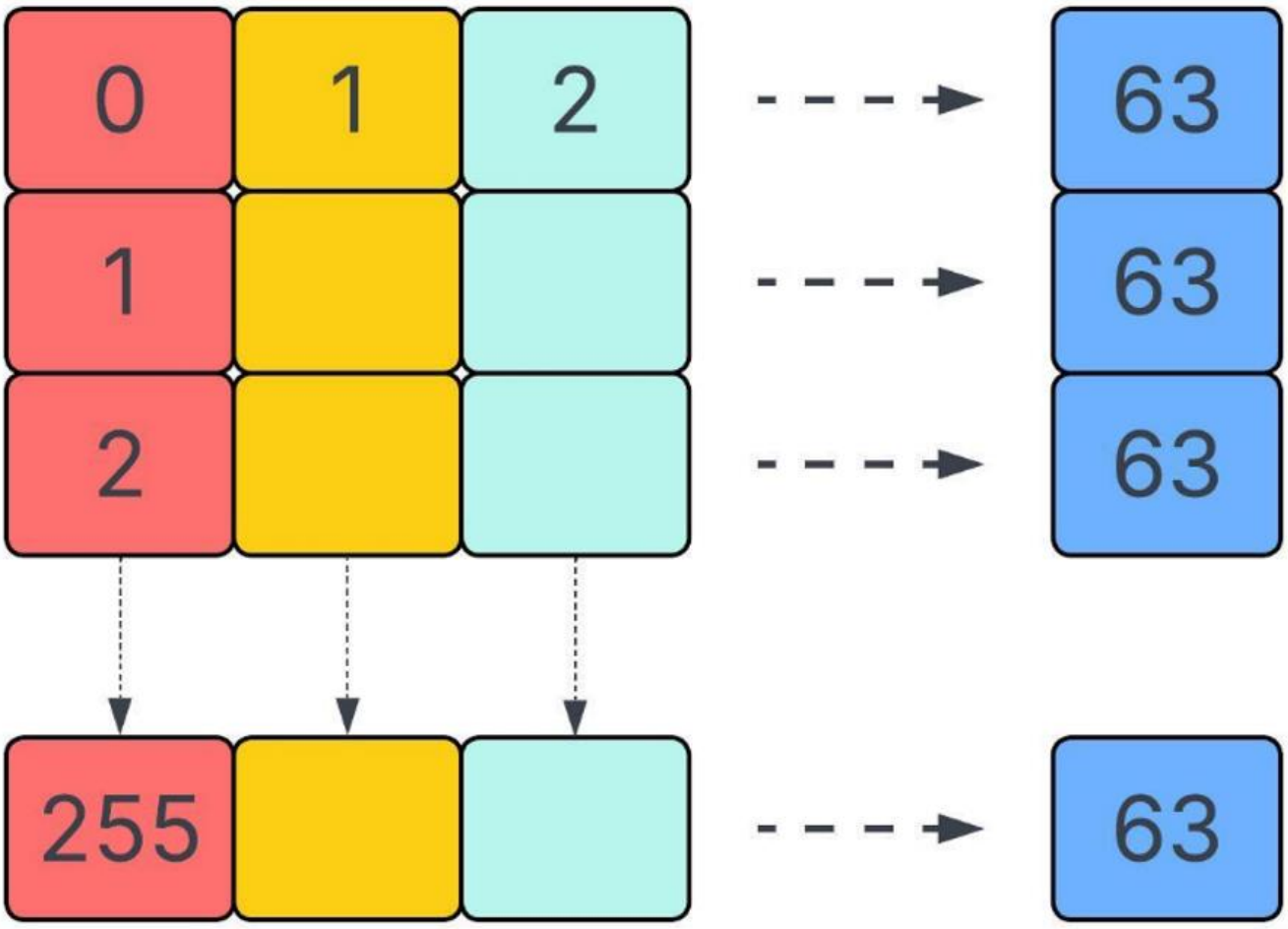

Figure 2.
The first method of image partitioning (proposed1).

The second method illustrated in Figure 3 and Equation 2, the image X1 pixels start from pixel 1 to pixel 64, the X2 image starts from pixel 65 to pixel 128, the X3 image starts.

From pixel 129 to pixel 192, and the X4 image starts from pixel 193 to pixel 256, as depicted in Fig.3, the splitting process is executed horizontally.

$$x(k, i) = Ima(k, i + (x(n - 1) * 64)) \quad (2)$$

Where:

$1 \leq i \leq 64$: Index variable for the column.

$1 \leq k \leq 256$: Index10 variable for the row.

$1 \leq n \leq 4$: indication for a new image.

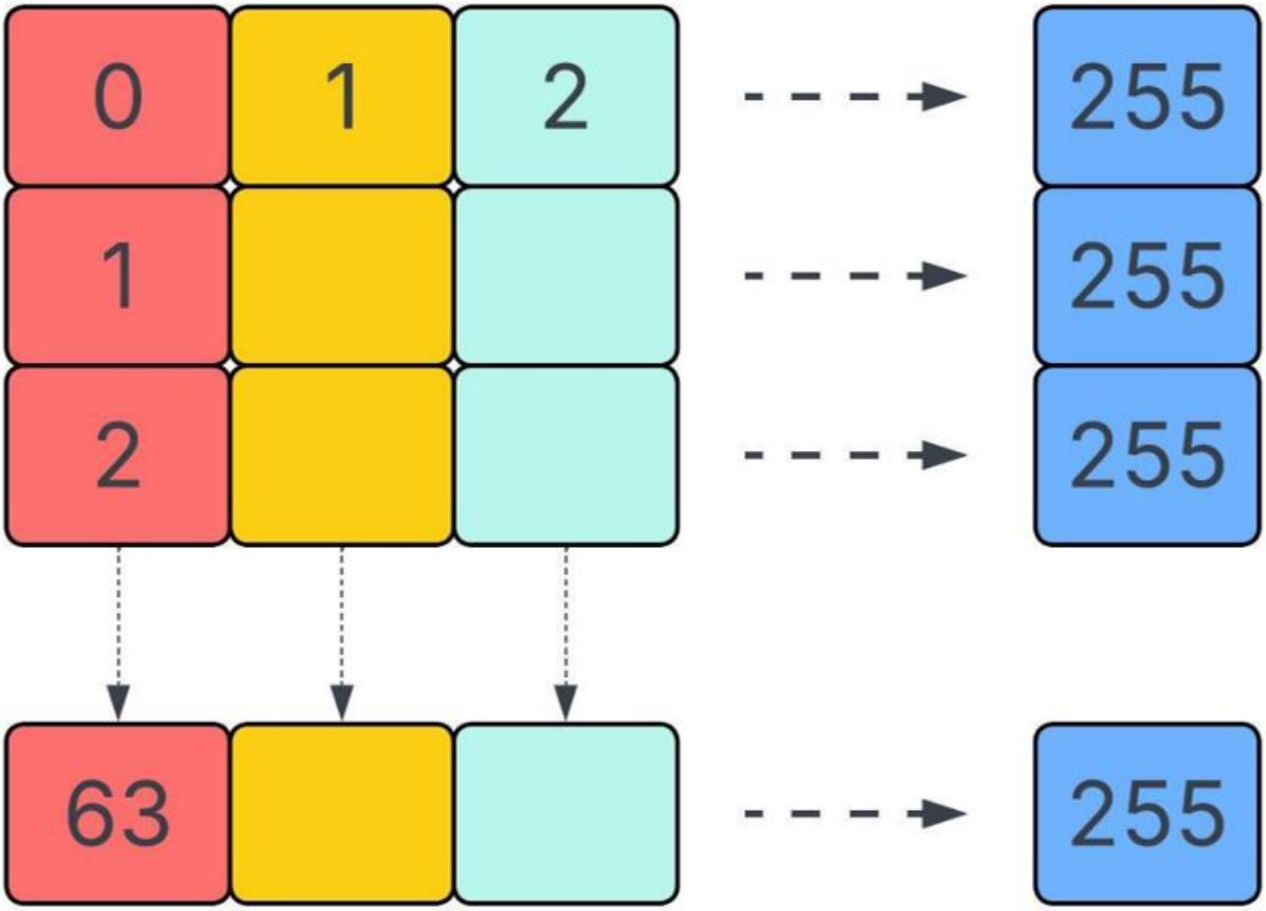

Figure 3.
The second method of image partitioning (proposed2).

The segment image sized (256 × 64) pixels, is encrypted in parallel using one of the four Chaos Pseudo-Random Bit.

Generators (PRBGs) chaotic systems [9]: the Lozi map [10], Tent map [11], Logistic map [12], and Quad map [13]. Specifically, X1 sub-image processed with quad map keys, X2 sub-image processed with logistic map keys, X3 sub-image processed with Lozi map key, and X4 sub-image processed with Tent map key. The key width of each PRBG's parameters is 32 bits, but the pixels in the image consist of 8 bits, so the 8 LSBs will be used due to their high randomness characteristic compared to other bits as shown in Figure 4.

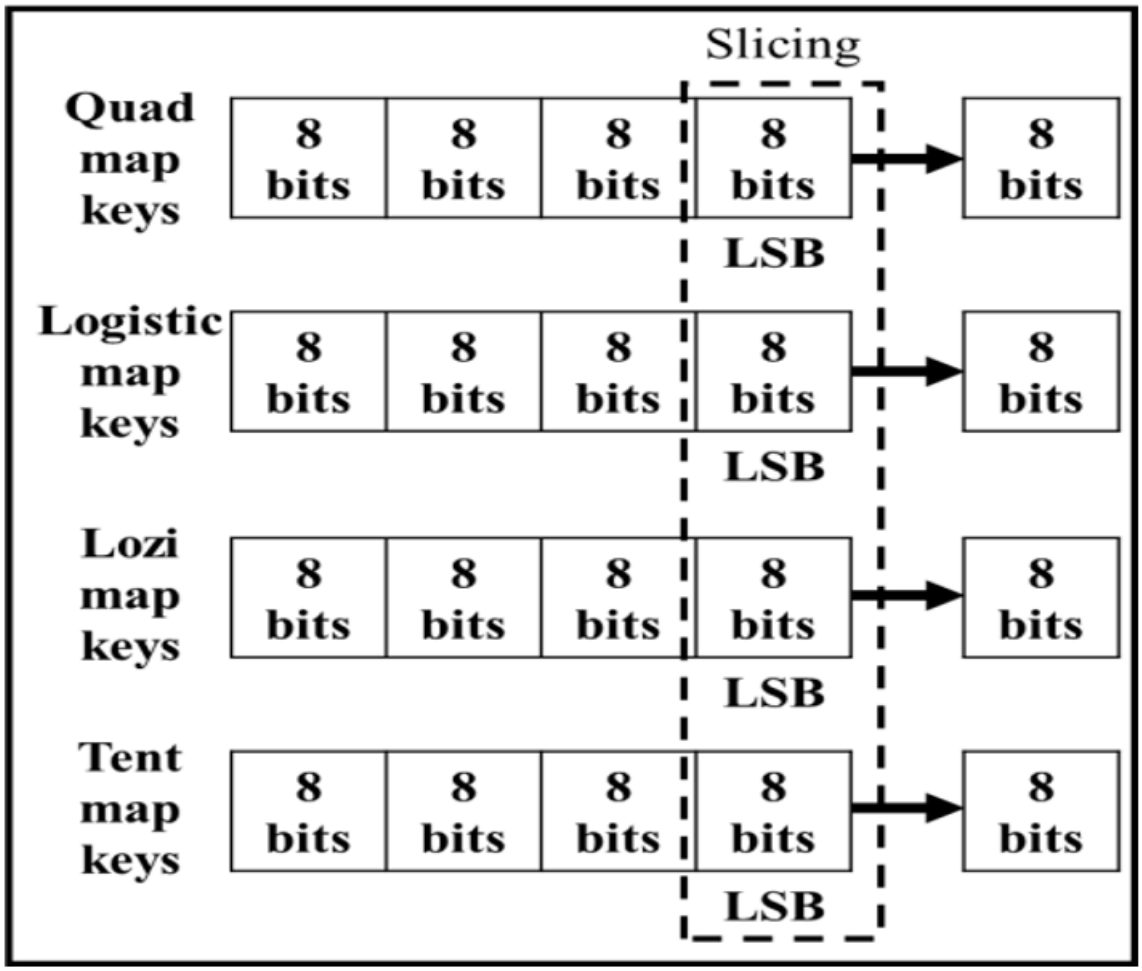

Figure 4.
Four keys slicing.

3. Hardware Implementation

The proposed algorithm hardware implementation is performed using the field-programmable gate array (FPGA) ZYNQ-7020 evaluation board kit [14], with a hardware-software co-approach. Utilization of resources in the FPGA Logic slice with a precision of 32-bit fixed-point numbers configuration. Fixed-point with 2 bits for the integer part and 28 bits for the fractional part. As mentioned before, four of the 32-bit encryption keys are used for computations, but only the least significant 8 bits are utilized for image encryption because they are the most changed parts. The least significant 8 bits are preferred due to their instant randomness within the generated key. These 8-bit key parts are used for encrypting corresponding portions of the image via XOR operations. Quadratic, Logistic, Lozi, and Tent chaotic maps generate the 8-bit keys for encrypting the first, second, third, and fourth image partitions, respectively.

The quadratic (Quad) map key is derived from a quadratic chaotic map [15]. This map, characterized by its complex and unpredictable behavior, uses tunable parameters and initial conditions to generate a unique key. The key is then employed in the encryption algorithm, transforming the original data into a secure ciphered format. The large key space provided by the tunable parameters of the quadratic chaotic map enhances the security of the encryption, making it resistant to brute-force attacks. Equation 3 describes the generation of the Quad key x (n+1), where a is a constant value $0 < a \leq 2$.

$$x(n+1) = a\text{-}x\,(n)^2 \qquad (3)$$

The hardware implementation, shown in Figure 5, starts from the multiplexer that is selected using the pulse signal unit. The output of this unit at the initial state is one to select x as the initial value (0.9316699891400337), and then becomes zero to select from the 32-bit register (x(n)). The result is that 32 bits will be truncated to choose the first 8 bits (LSB) as the key.

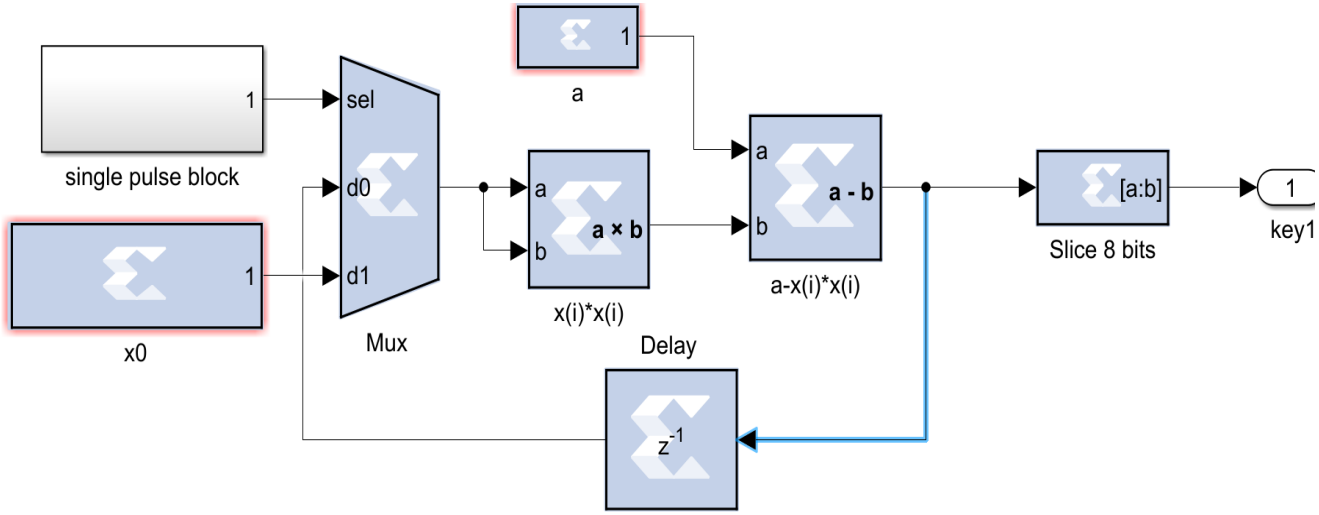

Figure 5.
FPGA implementation of the Quad map.

The logistic map key is derived from a logistic map, a one-dimensional nonlinear transformation that exhibits chaotic behavior. The key generation uses the logistic map's control parameter and initial value, both of which are adjustable.

To create a vast key space, this key is used in the encryption algorithm to convert the original data into a secure, ciphered format. The unpredictability and sensitivity to initial conditions of the logistic map enhance the security of the encryption, making it resistant to various cryptographic attacks. This key is generated by using Equation 4 where x(n) represents the current key value, r is a control parameter, and x(n+1) is the next key value. The control parameter r is typically a number between 0 and 4, typically between 3.5 and 4. The hardware implementation for this method is shown in Equation 4 and Figure 6.

$$x(n+1) = r^{*}x\,(n)(1 - x(n)) \qquad (4)$$

The Lozi map key is generated from a Lozi map, a two-dimensional piecewise linear chaotic map. The key is produced using the Lozi map's parameters and initial conditions, which can be tuned to create a large key space. This key is then employed in the encryption algorithm to transform the original data into a ciphered format, ensuring data security. The complex and chaotic behavior of the Lozi map contributes to the robustness of the encryption, providing resistance against common cryptographic attacks.

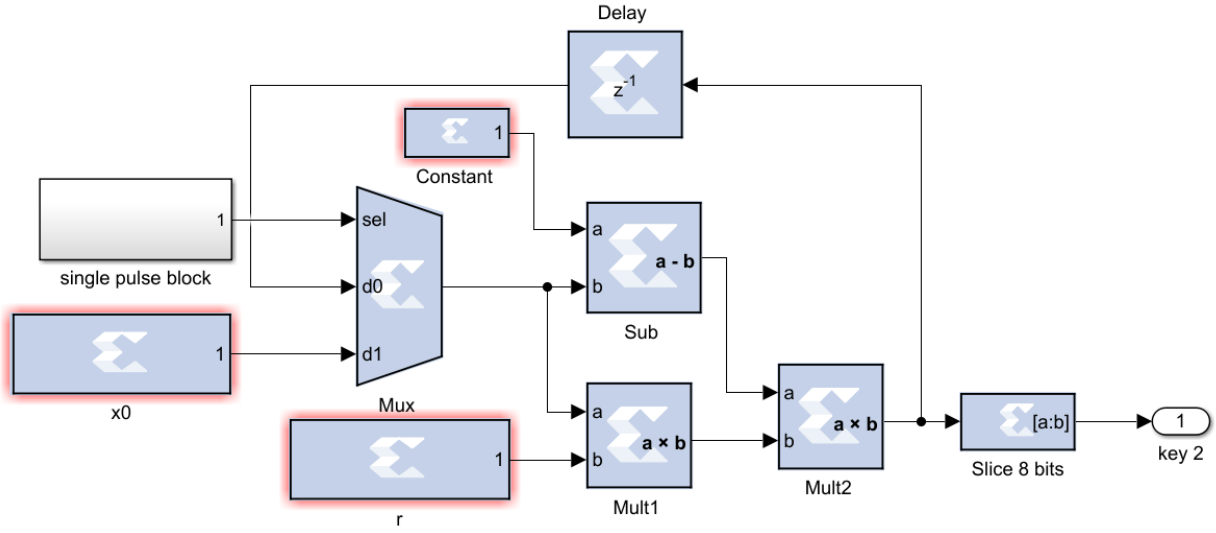

Figure 6.
FPGA implementation of Logistic map.

The key here is generated using two Equations 5 and 6; x(n) and y(n) represent the current state values, a and b are control parameters, and x(n+1) and y(n+1) are the next state values. The control parameters α and β can be adjusted to create a vast key space. Figure 7 shows the hardware implementation for this method.

$$x(n+1) = 1 - \alpha|x(n)| + y(n) \qquad (5)$$

$$x(n+1) = \beta\, x(n) \qquad (6)$$

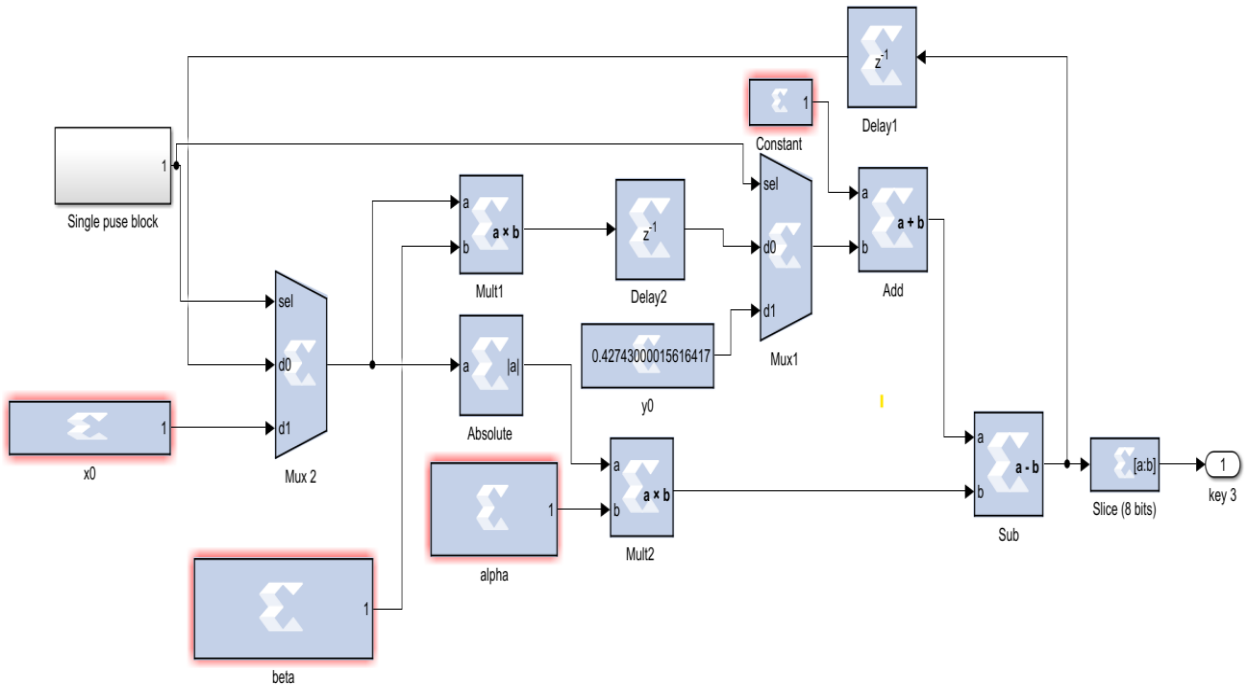

Figure 7.
FPGA implementation of the Lozi map.

The tent map key in encryption is based on a tent map, a piecewise linear, one-dimensional map exhibiting chaotic behavior. The key is generated using the tent map's control parameter and initial value, both of which can be adjusted to create a vast key space. Then, this key is used in the encryption algorithm to convert the original data into a secure, ciphered format. The encryption becomes more secure and impervious to various cryptographic attacks due to its unpredictable behavior and sensitivity to the tent map's initial settings. Equation 7 describes the generation of the tent map key, x(n) represents the current key value, μ is a control parameter, and x(n+1) is the next key value. The control parameter μ is typically a number between 0 and 2. The tent map's erratic behavior, which is desirable for encryption, is observed when the value of μ is within certain ranges, typically between 1 and 2. Figure 8 shows the hardware implementation for this method.

$$X(n+1) = f(x) = \begin{cases} \mu\,(n)\ for\ \ (n) < \frac{1}{2} \\ \mu\,(1-n)\ for\ (n) \geq \frac{1}{2} \end{cases} \qquad (7)$$

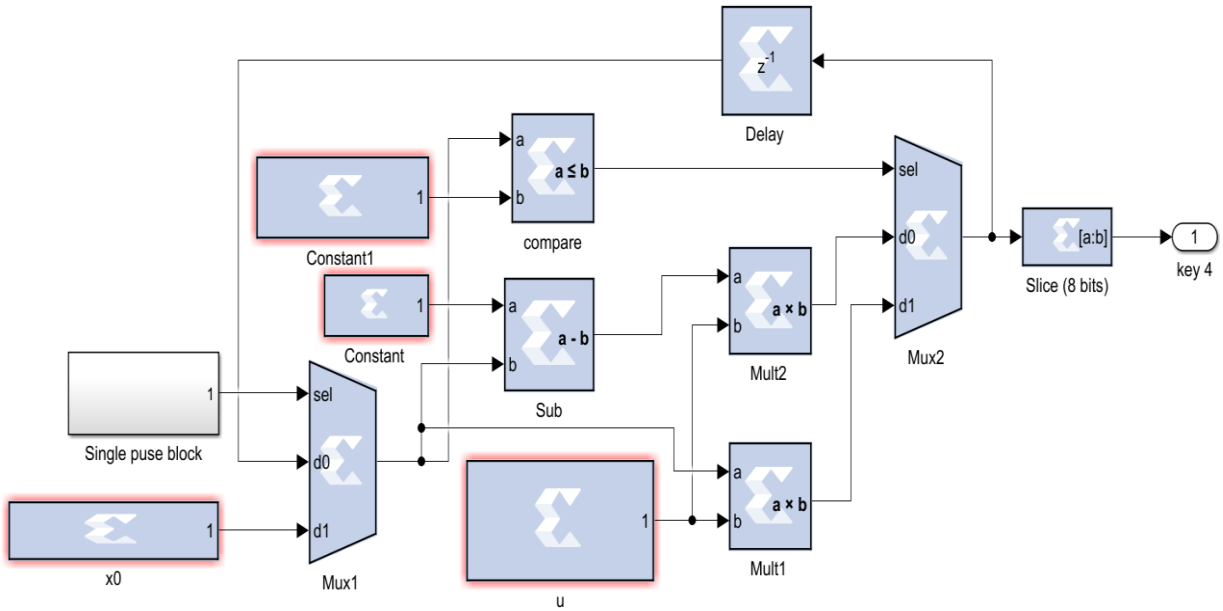

Figure 8.
FPGA implementation of Tent map.

4. Result and Discussion

4.1. NIST Test

15 tests were provided by the National Institute of Standards and Technology (NIST) for finding the state of the unpredictability of the cryptographic keys involved Frequency (Monobit) Test, Frequency Test within a Block, runs-test, Test for the Longest Run of Ones in a Block, Binary Matrix Rank Test, Maurer's "Universal Statistical" Test, and Cumulative Sums (forward, reversed) tests. The generated random keys undergo these tests then then finding the P-value when the P-value is larger than 0.01, as shown in Table 1. P-value represents the threshold value that is considered the degree of randomness. If the P-value is equal to zero, it is an indication. The bits are entirely non-random. While the maximum level of unpredictability is indicated by a P-value of one. Table 1 compares the results with Hasan and Saffo [16] and El Hadj Youssef et al. [18], showing that the proposed method achieves the best randomness values for runs, the longest run, Maurer, and Reversed Cumulative Sums tests.

4.2. Security Test Results

The first test that shows the distribution of each pixel in the image is the histogram. The original image's pixel distribution is haphazard, while the ciphered image's is methodical, making it impenetrable. To prevent the picture from being attacked, the ciphering should create a distribution with equalized features. Figures 9, 10 show the histograms of the ciphered images of the two proposed encryption systems, respectively.

Table 1.
NIST Keys Randomness Tests.

P-value of randomness tests/test name	**Results**		
	Hasan and Saffo [16]	El Hadj Youssef, et al. [17]	***proposed***
Frequency (Monobit) Test	0.9203	0.8343	0.1622
Frequency Test within a Block	0.9782	0.275	0.1315
Runs test	0.8666	0.7981	0.74
Test for the Longest Run of Ones in a Block	-	0.4559	0.772
Binary Matrix Rank Test	0.2918	0.2492	0.3637
Maurer's "Universal Statistical" Test	-	0.6579	0.7376
Forward Cumulative Sums Test	0.1256	0.4694	0.2365
Reversed Cumulative Sums Test	0.8945	0.4694	1

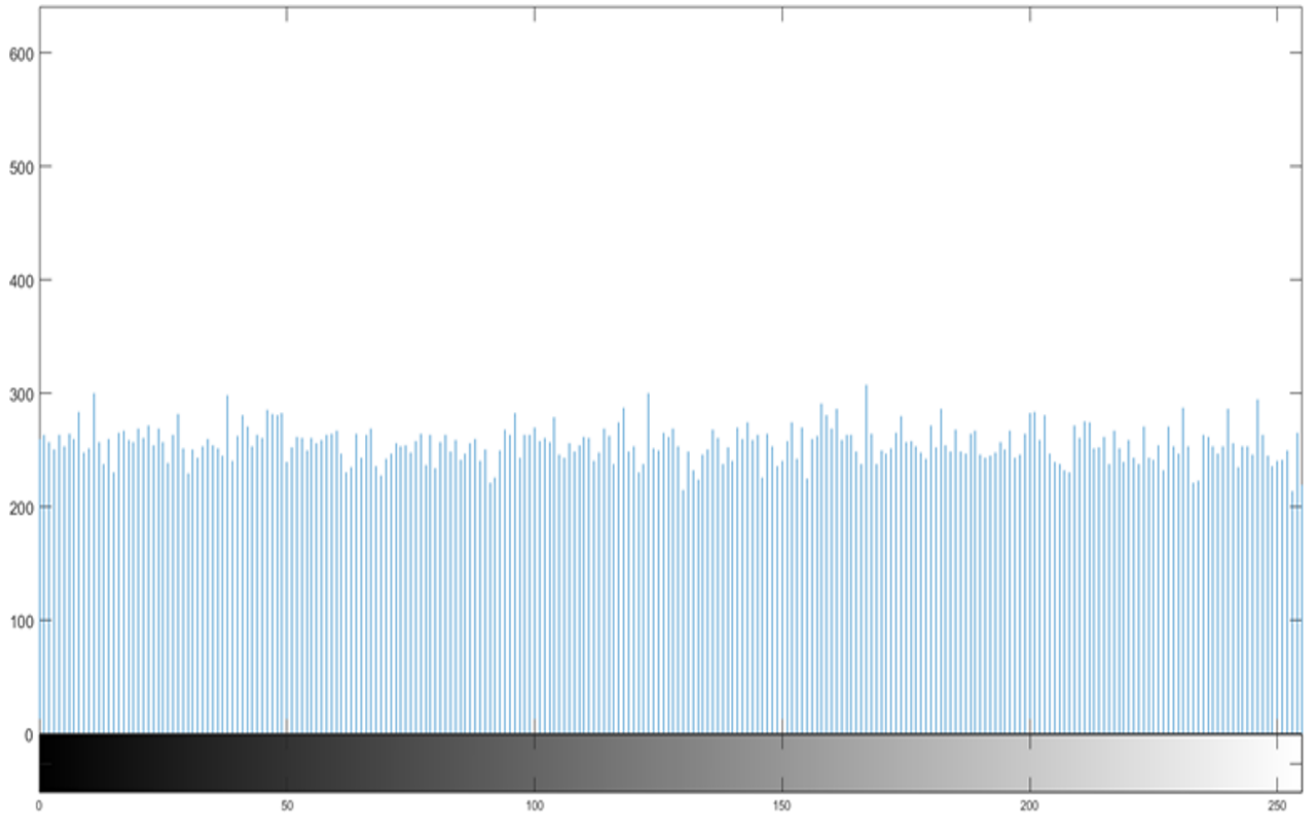

Figure 9.
Histogram of the ciphered image (proposed method 1).

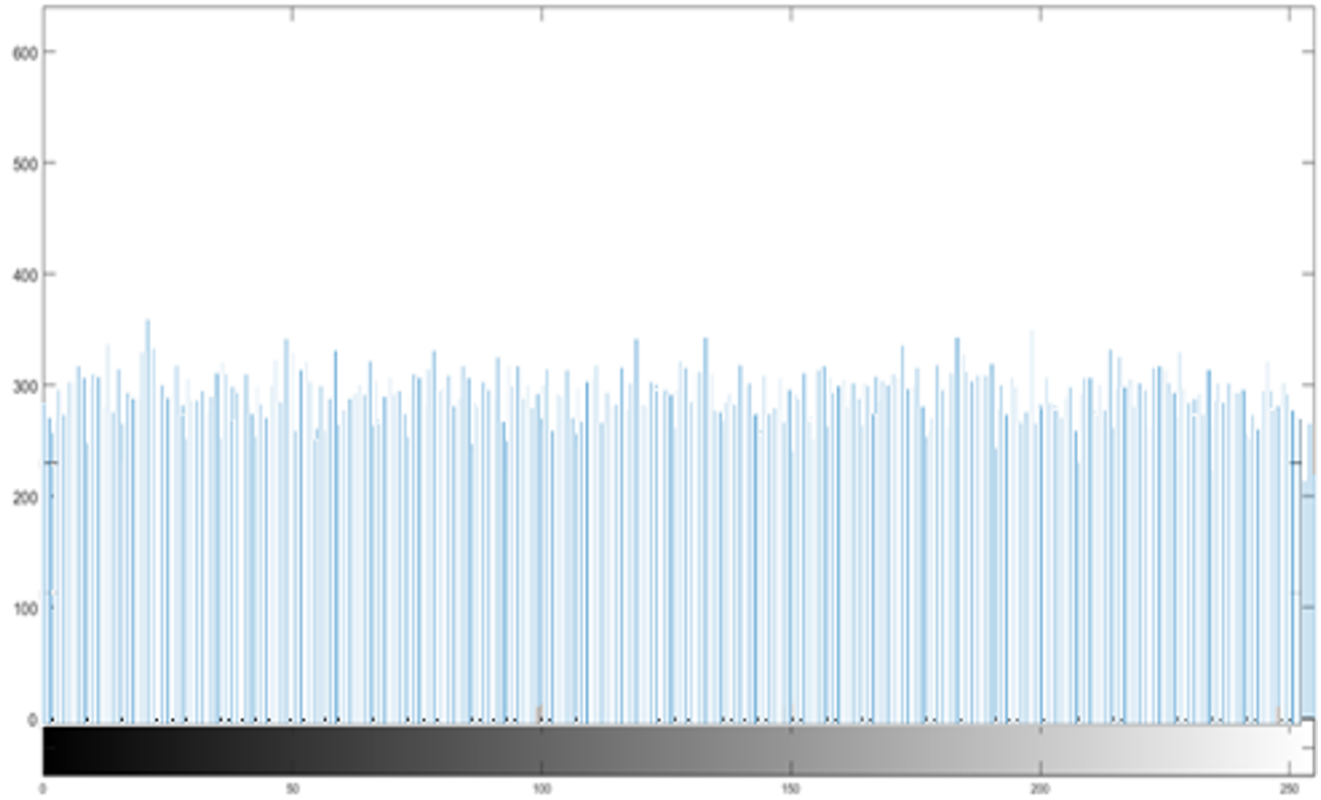

Figure 10.
Histogram of the ciphered image (proposed method 2).

The most popular tests are the Frequency (Monobit) Test, Frequency Test within a Block, Runs Test, Test for the Longest Run of Ones in a Block, Binary Matrix Rank Test, Maurer's Universal Statistical Test, and Cumulative Sums (forward, reversed) tests, as shown in Table 1. These tests are performed on the generated random bits, and if the P-value is higher than 0.01, then the tests are deemed successful. To decide whether to accept or reject the generated random bits, the P-value serves as a threshold for determining whether to accept or reject the bits. The highest degree of randomness is indicated by a P-value of one, but a P-value of zero indicates that the bits are not random at all. The comparisons with ref Hasan and Saffo [16] and ref El Hadj Youssef, et al. [17] in Table 1 show that our proposed has the best randomness value for runs, longest run, Maurer, Reversed Cumulative Sums tests.

This section looks at the systems' security performance using a variety of security metrics

such as histogram analysis, correlation, entropy Rodríguez-Orozco, et al. [18], key space Zhu, et al. [19], pixel changing rate [20] and structural similarity as shown in Table2. The analysis counted to the (256*256) image size and executed embedded Xilinx System Generator inside MATLAB/SIMULINK environment. Table 2 lists a comparison between this work and other relevant work when applied to the 256*256-cameraman image. All of the original image's information should be successfully hidden by a good image encryption, producing a ciphered image that seems random and unrelated. Two identical images are indicated by a correlation coefficient of 1. However, the ciphered image is the exact opposite of the original image, according to a correlation coefficient of -1 [21]. Information Entropy is another security measurement used to test the uncertainty of an image, which is computed in Merah, et al. [12]. A cryptographic technique that has a large key space is more resistant to brute-force attacks. Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are metrics that gauge the average intensity of variances between the original and encrypted images. SSIM (Structural Similarity Index) is another statistic used to determine how similar the original and encrypted images are. The plaintext and cipher images are less comparable when the SSIM value is smaller.

The correlation coefficient test shows that the proposed system number 2 has the better results since it is the nearest to -1. The entropy test showed that proposed system number 1 has the nearest value to 8, so it is the best. The key space offered by the proposed system is better, as seen in 2288. The NPCR change test showed that proposed system number 2 has the largest change. The UACI change presents the biggest change for proposed 1 and proposed 2. The structural similarity between the plain image and the ciphered image showed that proposed 1 has the best performance since it has the lowest number among others.

Table 2.
Security Tests Performance Comparison

Test name	Results		
	Shengtao, et al. [22]	**Proposed 1**	**Proposed 2**
Correlation coefficients	-0.0065	-0.0065	-0.0063
Information Entropy	7.9035	7.9972	7.9970
Key Space Analysis	10^{40}	2^{288}	2^{288}
Number of Pixels Change Rate (NPCR)	99.60	99.60	99.601
Unified Average Changing Intensity (UACI)	33.4477	33.4635	33.4635
SSIM (Structural Similarity Index)	-	0.0424	0.0425

4.3. FPGA Hardware Co-Simulation of the Proposed Systems

The splitting of the image rows and columns differs for each system, but the FPGA units are the same, so there is no difference in FPGA utilization between the two systems since the same components are used for both, as shown in Figure 11. The Xilinx System Generator (XSG) tool was used to generate the VHDL code for each model. The work was conducted using the ZYNQ 7000 SoC ZC702 Evaluation Kit. Table 3 provides details about the device applications. Throughput is calculated as (f × 8), where f is the maximum frequency, representing the number of bits processed per second. [23]. The primary metric for evaluating the effectiveness of cryptographic security measures is throughput.

5. Conclusion

This paper presents two FPGA-based image encryption systems utilizing four types of chaotic maps (Proposed 1) and (Proposed 2).

Table 3.
Xilinx Zc702 Evaluation Kit Device Area.

Resource Type	Results	
	El Hadj Youssef, et al. [17]	**Proposed**
LUT	1079	894
FF	67	1192
Slices	99	-
BRAM	-	2
DSP	-	28
Minimum period (ns)	19.5	1.49
Maximum frequency (MHZ)	51.25	667
Throughput (Gbit/sec)	3.28	5.336

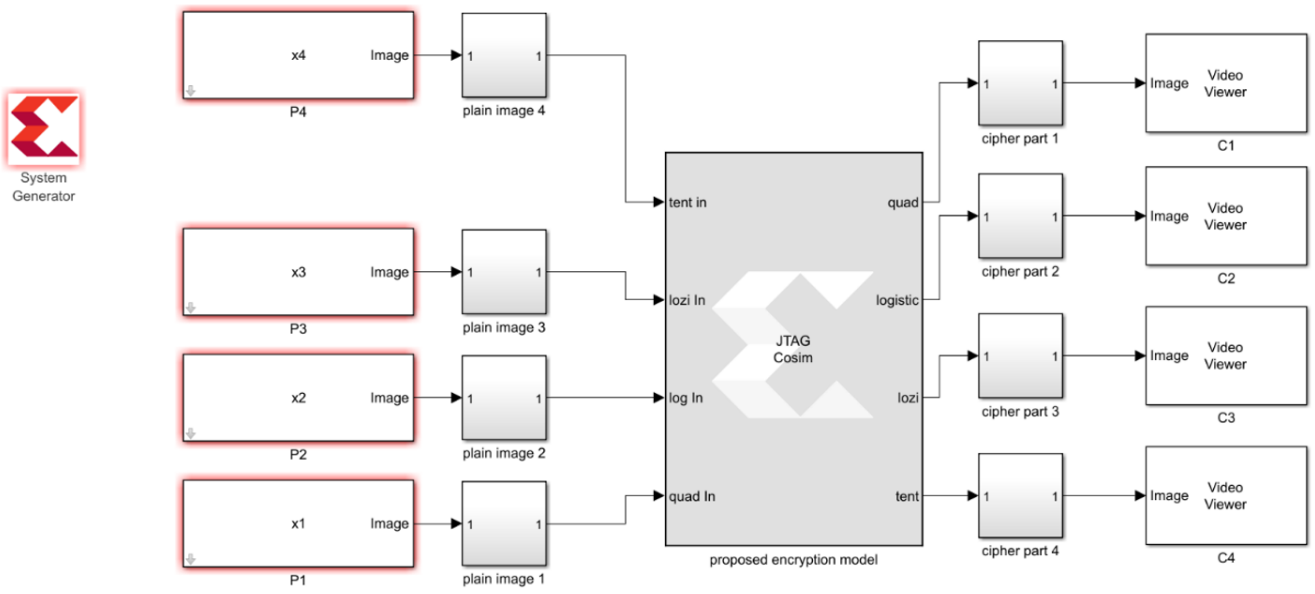

Figure 11.
Histogram of the ciphered image (Proposed2).

The first proposed system depends on splitting the message image vertically, then performing encryption, while the second utilizes the splitting of the original image horizontally before the encryption process.

The tests of these encryption systems are divided into two categories: firstly, NIST tests for the randomness of keys generated by the system; secondly, security performance tests such as correlation, entropy, key space, NPCR, UACI, and

SSIM. Table 2 shows that NIST test for our proposed encryption models is the best in most of tests as a comparison with other recent researchers. Table 3 showed correlation coefficients for the proposed system 2 have the best security efficiency since they are the nearest to -1. Information entropy has the best security efficiency since the nearest value to 8. The key space offered by the two proposed systems was 2288. The NPCR for proposed system 2 is the best, while the UACI for both proposed systems is the best. The structural similarity test (SSIM) between the two images (original and ciphered) showed that the proposed system 1 is the best since it has a less similar structure value.

References

[1] A. T. Suhail, H. G. Ayoub, and A. A. Gharbe, "A strong algorithm for randomly hiding a secret files inside true color image using large primary secret key," *Egyptian Informatics Journal*, vol. 30, p. 100692, 2025.

[2] H. G. AYOUB, "Dynamic iris-based key generation scheme during iris authentication process," presented at the International Conference on Contemporary Information Technology and Mathematics (ICCITM), 2022.

[3] Z. A. Abdulrazzaq, O. Tareq, O. H. Mohammed, and M. R. Ahmed, "New novel FPGA based image encryption methods using multiple chaotic maps," presented at the International Conference on Computer and Applications (ICCA), 2024.

[4] H. G. Ayoub, Z. A. Abdulrazzaq, A. F. Fathil, S. A. Hasso, and A. T. Suhail, "Unveiling robust security: Chaotic maps for frequency hopping implementation in FPGA," *Ain Shams Engineering Journal*, vol. 15, no. 11, p. 103016, 2024. https://doi.org/10.1016/j.asej.2024.103016

[5] K. H. Thanoon, A. F. Shareef, and O. A. Alsaif, "Digital processing and deep learning techniques: A review of the literature," *NTU Journal of Engineering and Technology*, vol. 1, no. 3, 2022. https://doi.org/10.56286/ntujet.v1i3.223

[6] J. Shi, T. Zhao, Y. Wang, Y. Feng, and J. Wu, "Chaotic image encryption based on boson sampling," *Advanced Quantum Technologies*, vol. 6, no. 2, p. 2200104, 2023.

[7] X. Chai, Z. Gan, K. Yuan, Y. Chen, and X. Liu, "A novel image encryption scheme based on DNA sequence operations and chaotic systems," *Neural Computing and Applications*, vol. 31, no. 1, pp. 219-237, 2019.

[8] J. Zhang, D. Fang, and H. Ren, "Image encryption algorithm based on DNA encoding and chaotic maps," *Mathematical Problems in Engineering*, vol. 1, p. 917147, 2014.

[9] M. Ahmad, M. N. Doja, and M. M. S. Beg, "Security analysis and enhancements of an image cryptosystem based on hyperchaotic system," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 1, pp. 77–85, 2021.

[10] G. Sathishkumar and D. N. Sriraam, "Image encryption based on diffusion and multiple chaotic maps," *arXiv preprint arXiv:1103.3792*, 2011.

[11] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications (NIST special publication 800-22 Revision 1a). U.S. department of commerce, technology administration," National Institute of Standards and Technology, 2001. https://doi.org/10.6028/NIST.SP.800-22r1a

[12] L. Merah, A. Ali-Pacha, N. Hadj-Said, B. Mecheri, and M. Dellassi, "FPGA hardware co-simulation of new chaos-based stream cipher based on Lozi map," *International Journal of Engineering and Technology*, vol. 9, no. 5, pp. 420-425, 2017.

[13] V. Patidar, K. K. Sud, and N. K. Pareek, "A pseudo random bit generator based on chaotic logistic map and its statistical testing," *Informatica*, vol. 33, no. 4, pp. 441–452, 2009.

[14] Xilinx, "ZC702 evaluation board for the Zynq-7000 XC7Z020 All Programmable SoC user guide UG850," vol. 5 2015. https://www.xilinx.com.

[15] M. K. Ibrahem and H. A. Qasim, "Implementation of VoIP speech encryption system using stream cipher with Lorenz map key generator," *International Journal of Scientific and Engineering Research*, vol. 8, no. 7, pp. 533-541, 2017.

[16] F. S. Hasan and M. A. Saffo, "FPGA hardware co-simulation of image encryption using stream cipher based on chaotic maps," *Sensing and Imaging*, vol. 21, no. 1, p. 35, 2020.

[17] W. El Hadj Youssef, M. Elboukhari, M. Rziza, M. Talea, and A. S. Sadiq, "A secure chaos-based lightweight cryptosystem for the internet of things," *IEEE Access*, 2023.

[18] E. Rodríguez-Orozco *et al.*, "FPGA-based chaotic cryptosystem by using voice recognition as access key," *Electronics*, vol. 7, no. 12, p. 414, 2018. https://doi.org/10.3390/electronics7120414

[19] S. Zhu, C. Zhu, and W. Wang, "A new image encryption algorithm based on chaos and secure hash SHA-256," *Entropy*, vol. 20, no. 9, p. 716, 2018. https://doi.org/10.3390/e20090716

[20] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749-761, 2004.

[21] H. S. Alhadawi, M. F. Zolkipli, S. M. Ismail, and D. Lambić, "Designing a pseudorandom bit generator based on LFSRs and a discrete chaotic map," *Cryptologia*, vol. 43, no. 3, pp. 190-211, 2019.

[22] G. Shengtao, W. Tao, W. Shida, Z. Xuncai, and N. Ying, "A novel image encryption algorithm based on chaotic sequences and cross-diffusion of bits," *IEEE Photonics Journal*, vol. 13, no. 1, pp. 1-15, 2020.

[23] S. Ismae, O. Tareq, and Y. T. Qassim, "Hardware/software co-design for a parallel three-dimensional bresenham's algorithm," *International Journal of Electrical and Computer Engineering* vol. 9, no. 1, pp. 148-156, 2019.