International Journal of Basic and Applied Sciences, 14 (6) (2025) 274-281

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/4x2x0830
Research paper

An Efficient DES-Based Architecture for Teleradiology Applications

Prakash Marakumbi [1] *, Satish Bhairannawar [2]

[1] Department of Electronics and Communication, Tontadarya College of Engineering, Karnataka, India
[2] Department of Electronics and Communication, SDM College of Engineering and Technology, Karnataka, India
**Corresponding author E-mail: pmarakumbi@gmail.com*

Received: August 30, 2025, Accepted: October 6, 2025, Published: October 14, 2025

Abstract

Ensuring the security of images transmitted openly is critical due to their high susceptibility to unauthorized access. To mitigate such risks, employing data security measures is essential. With the increasing importance of data security in today's digital age, Data Encryption Stand-ard (DES) stands out as a notable block cipher technique for encrypting and decrypting data. Protecting sensitive information is crucial in embedded applications. Implementing cryptographic algorithms on programmable devices like Field Programmable Gate Arrays (FPGAs) combines the physical security of hardware solutions with the speed superiority over software-based approaches. FPGAs offer flexibility like software applications, with the added advantage of being reprogrammable to adapt to different cryptographic algorithms or modifications within the same algorithm, ensuring robust data protection. This work introduces a hardware-oriented approach to executing the DES algorithm using FPGAs. The approach adopted utilizes an iterative loop process with a 128-bit key size, leveraging a lookup table based on S-boxes. Simulation waveforms confirm that the FPGA-based implementation meets the desired specifications, showcasing high throughput. The research highlights a novel hardware-based version of the Triple DES encryption algorithm that achieves high-speed performance with minimal hardware requirements.

Keywords: *Cryptography; FPGA; Peak Signal to Noise Ratio; Triple Data Encryption Standard; VHDL.*

1. Introduction

The rise of social media platforms has significantly increased the volume of data shared, particularly digital images, leading to a greater need for secure data exchange to prevent misuse or theft of such images [1], [2]. To address these concerns, implementing security measures in the image transmission process is crucial [3]. The integration of cryptography and steganography methods has been shown to enhance the security of image transfers [4], [5]. Cryptography, derived from the Greek words 'Kryptos' meaning hidden, and 'Graphein' meaning to write [6], is the practice of protecting information from unauthorized access. It involves encoding data into an unintelligible format using a specific key, making the information accessible only to those who possess the key. This approach is particularly effective in safeguarding critical data during transmission. The foundational concepts of cryptography are illustrated, where an original readable message, known as plaintext, is transformed into an encrypted version referred to as ciphertext (Fig.1). This transformation is known as encryption or enci-phering (Fig. 1A). Conversely, the process of converting the encrypted ciphertext back into its original plaintext form is termed decryption [7] or deciphering (Fig.1B). Both encryption and decryption processes necessitate the use of specific auxiliary variables, commonly known as keys. Among the various cryptographic techniques, the Data Encryption Standard (DES) has been prominently recognized and previ-ously adopted as a benchmark for the encryption of data.

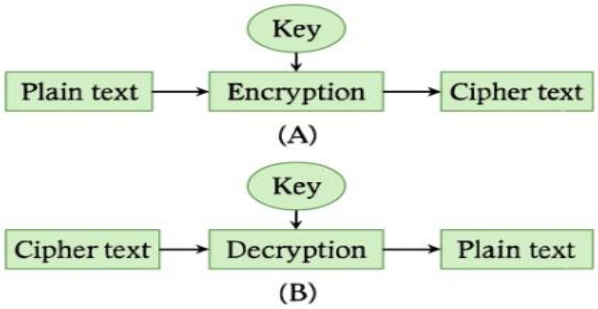

Fig.1: (A) Encryption Process Models (B) Decryption Process Models.

The Data Encryption Standard (DES), introduced by the National Bureau of Standards (NBS) in 1976 as a cryptographic benchmark for securing data, is known for transforming readable text into encrypted outputs. Despite its historical significance, DES's usage has declined due to its encryption outcomes being detectable and potentially raising suspicions. To address this limitation of cryptography, steganography emerges as a complementary solution.

Steganography, derived from the Greek words 'Stegonos' meaning concealed, and 'Graphein' meaning to write [9], involves the practice of embedding data within a medium, or 'cover', making the secret message's presence challenging to detect. Common media used as covers in steganography include text files, digital images, videos, audio files, and even IP Protocols, with digital images being particularly favored for their effectiveness in concealing data.

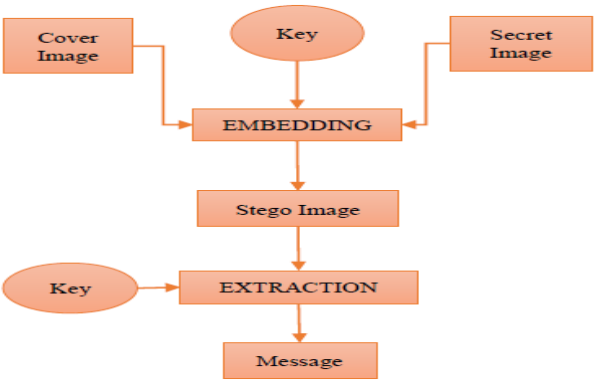

Fig. 2: Simple Process of Image Steganography.

The fundamental mechanism of image steganography (Fig.2) highlights that the outcome of the embedding process is a 'stego' file. This file contains the hidden message, which can be retrieved using a specific key. Among the various steganography techniques, the Least Significant Bit (LSB) method stands out for its simplicity and widespread use. This technique involves embedding each bit of the message into the least significant bit of each pixel in an image. A key advantage of the LSB method is its subtlety; the alterations made to the cover image during message embedding are minimal, making them virtually undetectable to the human eye. This ensures the integrity of the image's appearance, despite the embedding of hidden information [11].

The message insertion process in the Least Significant Bit (LSB) method, due to its sequential nature, is relatively straightforward, potentially allowing attackers to easily retrieve the hidden message. To enhance security, the author integrates the LSB technique with an edge detection algorithm. This combination leverages the edges within an image to dictate where messages should be embedded. By using the characteristics of image edges to determine message placement coordinates, the method randomizes the positions of the embedded messages, thereby increasing the complexity and security of the steganographic process.

2. Literature Survey

The protection of sensitive image data has been addressed extensively through a combination of cryptographic and steganographic methods. These approaches focus on enhancing confidentiality, robustness, and resilience against unauthorized access.

Early work by Shah et al. [12] highlighted the importance of integrating classical cryptography with image-based steganography. Their method employed the Data Encryption Standard (DES) for encryption, followed by embedding the encrypted message into 24-bit JPEG images using Discrete Wavelet Transform (DWT). The resulting stego-images preserved high Peak Signal-to-Noise Ratio (PSNR) values even when subjected to various attacks, demonstrating that the dual-layer approach can safeguard confidentiality without perceptible image degradation.

Building on this direction, Verma and Kaur [8] explored hybrid cryptography–steganography schemes using both grayscale and color images as cover media. Their work introduced bit-matching techniques to improve embedding efficiency in images with diverse color distributions. While effective under normal conditions, their results also revealed vulnerability to noise interference. Specifically, the addition of salt-and-pepper and Gaussian noise degraded hidden message integrity, as reflected by increased Mean Squared Error (MSE) values, underscoring the challenges of noise resilience in steganographic systems.

The role of edge-based embedding has also been a significant focus. Mungmode et al. [13] proposed Edge Adaptive Steganography, which combines thresholding with Least Significant Bit Matching Revisited (LSBMR). Their method achieved PSNR values approaching 90 dB, even when embedding up to 400 bits, suggesting that adaptive embedding in edge regions effectively balances payload capacity and imperceptibility.

Similarly, Gupta and Singh [14] applied Canny edge detection alongside hash-based randomization to determine embedding positions. Their findings indicated that images with more edge regions not only achieved higher PSNR values but also supported larger payloads, making edge-adaptive techniques particularly efficient.

Further comparative studies of edge detection operators have reinforced these insights. Mehta and Reddy [10] evaluated common operators, including Sobel, Prewitt, Robert, Laplace, and Canny. Their results confirmed that Canny consistently outperformed other methods in terms of payload capacity and imperceptibility, thereby validating its effectiveness for secure image steganography.

In parallel, research has shifted toward hardware-oriented implementations to achieve both speed and efficiency. Zhang et al. [15] introduced a reconfigurable FPGA-based framework for lightweight block ciphers, demonstrating scalability and energy efficiency for IoT security applications.

Extending this concept to healthcare, Kumar and Bansal [16] presented a hybrid cryptography–steganography model tailored for telemedicine. Their FPGA-driven design enabled secure real-time transmission of medical images, illustrating how hardware solutions can meet both performance and security demands in clinical practice.

More recently, Lee et al. [17] examined the integration of post-quantum cryptography with steganography, emphasizing its potential to protect multimedia and medical data in a future where quantum computing may undermine traditional encryption schemes.

Synthesis and Research Gap:

From this survey, three key gaps emerge:

- Many methods ([12], [13]) ensure high PSNR but rely on software or lightweight embedding with limited encryption strength.
- FPGA-based cryptographic works ([15], [16]) emphasize AES but are resource-intensive, limiting adoption in embedded medical systems.
- Few studies directly integrate strong encryption with hardware efficiency for applications such as teleradiology, where real-time secure image transfer is critical.

3. Research Method

3.1. Cryptography

Cryptography involves the use of codes and ciphers to secure information and communication, ensuring that only the intended recipients can understand and process the data. The term combines "crypto," meaning hidden, and "graphy," referring to writing, highlighting its focus on concealing the content of messages. This field relies on mathematical concepts and a set of algorithmic rules to encrypt messages in a way that makes them difficult to decrypt without the proper key. Various algorithms, including AES, RSA, DES, Blowfish, and Triple DES, are employed in cryptosystems to facilitate secure communication. These algorithms serve various purposes, such as generating encryption keys, digital signing, and authentication, thereby safeguarding data privacy and enabling secure transactions and communications across different platforms, including internet browsing, emails, and card transactions. A cryptographic suite employs distinct algorithmic approaches for various security functions: one for encryption, another for message authentication, and yet another for key exchange. These components are integrated into protocols and implemented through software that operates on various computing systems and networks. This setup encompasses the generation of public and private keys for encrypting and decrypting data, digital signatures and authentication protocols for verifying message integrity, and mechanisms for securely exchanging encryption keys. Concerns about the feasibility of brute-force attacks against the DES algorithm emerged around the 1990s, leading to apprehension among its users. However, transitioning away from DES presented significant challenges, including the substantial time and financial investment required to replace an encryption algorithm that had become deeply ingrained in extensive security infrastructures. Rather than discarding DES altogether, the practical solution involved adapting its usage to enhance security without abandoning the algorithm itself. This convergence led to the development of Triple DES (also known as 3DES), which exists in two variants: 3-key Triple DES (3TDES) and 2-key Triple DES (2TDES). In the 3-key version, users initially generate and distribute a 3TDES key, K, composed of three distinct DES keys: K1, K2, and K3. This means the total key length for 3TDES is 168 bits (3 times 56 bits for each DES key). The design of Triple DES as an encrypt-decrypt-encrypt sequence allows for backward compatibility with single DES by using the same value for K1, K2, and K3, enabling 3TDES systems to operate as single DES systems when needed. The 2-key variant, 2TDES, operates similarly to 3TDES but uses only two keys; it encrypts with K1, decrypts with K2, and encrypts again with K1, resulting in a total key length of 112 bits. Although Triple DES methods offer significantly enhanced security over the original DES, they are inherently slower due to the multiple encryption and decryption steps involved.

3.2. Steganography

Steganography involves the art of embedding a secret message, whether it be text, an image, or a video, within another seemingly innocuous file, message, image, or video. The term 'steganography' originates from the New Latin 'steganographia', blending the Greek words 'stegano', meaning 'covered or concealed', and 'graphia', meaning 'writing'. In this context, the hidden message could be discreetly interwoven among the lines of an ordinary text, making it undetectable immediately. Steganography employs secrecy in its method, differing from cryptography, which openly signals the presence of an encrypted message. A key feature of steganography is its ability to conceal any form of content, not just textual messages, without drawing suspicion. This subtlety is a significant advantage over cryptography, as it doesn't call attention to the existence of hidden information. Despite this, the challenge with steganography lies in its potential vulnerability to detection and decryption, prompting ongoing efforts to enhance its robustness and security. A novel technique in image steganography that utilizes both the Integrate Wavelet Transform (IWT) and the Discrete Cosine Transform (DCT) has been proposed to embed a secret image within the frequency domain of a host image, ensuring high fidelity in the matching process. The DCT function is particularly adept at converting digital image data from the spatial domain to the frequency domain, playing a crucial role in this process. During the transformation, information is encoded into the least significant bits (LSBs) of the frequency components, a method commonly associated with lossy compression techniques. Specifically, DCT is instrumental in JPEG image compression, as it categorizes image elements based on their relative importance, effectively segregating the image into high, medium, and low frequency components. The bulk of an image's signal energy, which includes the most critical visual details, resides in the low-frequency sub-band. Meanwhile, elements in the high-frequency sub-band, which are less crucial, can often be compromised during compression and noise interference. To conceal the secret message without affecting the host image's visual quality, adjustments are made to the coefficients in the medium frequency sub-band. This strategic placement ensures that the embedded message remains discreet, preserving the integrity of the host image's appearance. This method highlights the versatility of DCT in manipulating image data for steganographic purposes.

4. Proposed Triple DES Architecture

A product cipher is an encryption method that integrates both permutations and substitutions to enhance security. This technique involves a series of mathematical operations that include the replacement, shuffling, and reordering of elements within the key and the plaintext. The core process is as follows: The plaintext is segmented into blocks of 64 bits each. These blocks undergo an initial permutation, after which they are split into left and right halves. The encryption process then applies a sequence of substitution and permutation operations across sixteen rounds. After completing these rounds, the left and right sections of the text are reunited. The final step involves applying an inverse permutation to the recombined text, restoring the block to its original order but with the data encrypted (Fig.3). Initially, the plaintext block undergoes a preliminary permutation, while the key is processed separately. The key, spanning 64 bits, includes parity bits at every eighth position, which are removed to yield a 56-bit operational key. Subsequently, the 64-bit plaintext is bifurcated into two 32-bit segments, labeled as the right and left halves. The function F plays a crucial role by merging the key data with these two halves of the plaintext through a series of operations. This function F is executed 16 times, intertwining the key and plaintext information in each iteration. Following these iterations, the two halves are recombined into a unified 64-bit block. The final step involves applying a concluding permutation to this block, transforming it into a 64-bit ciphertext block, thus completing the encryption process.
The algorithm is designed to process 64-bit blocks of data using a 64-bit key for both encryption and decryption. For decryption, the same key utilized in the encryption phase is required, but the sequence in which the key's bits are applied is altered to reverse the encryption process. Decryption involves an initial permutation (IP), followed by a series of complex operations that depend on the key, and concludes with a final permutation that reverses the initial permutation (IP-1). The key-dependent operations can be broken down into two main functions: the key schedule (KS), which organizes the key bits for each round of the process, and the cipher function (f), which performs the core transformations on the data block.
Initially, the plaintext block undergoes a preliminary permutation, while the key is processed separately. The key, spanning 64 bits, includes parity bits at every eighth position, which are removed to yield a 56-bit operational key. Subsequently, the 64-bit plaintext is bifurcated into two 32-bit segments, labeled as the right and left halves. The function F plays a crucial role by merging the key data with these two halves of the plaintext through a series of operations. This function F is executed 16 times, intertwining the key and plaintext information in each iteration. Following these iterations, the two halves are recombined into a unified 64-bit block. The final step involves applying a concluding permutation to this block, transforming it into a 64-bit ciphertext block, thus completing the encryption process.
The algorithm is designed to process 64-bit blocks of data using a 64-bit key for both encryption and decryption. For decryption, the same key utilized in the encryption phase is required, but the sequence in which the key's bits are applied is altered to reverse the encryption process. Decryption involves an initial permutation (IP), followed by a series of complex operations that depend on the key, and concludes with a final permutation that reverses the initial permutation (IP-1). The key-dependent operations can be broken down into two main functions: the key schedule (KS), which organizes the key bits for each round of the process, and the cipher function (f), which performs the core transformations on the data block.

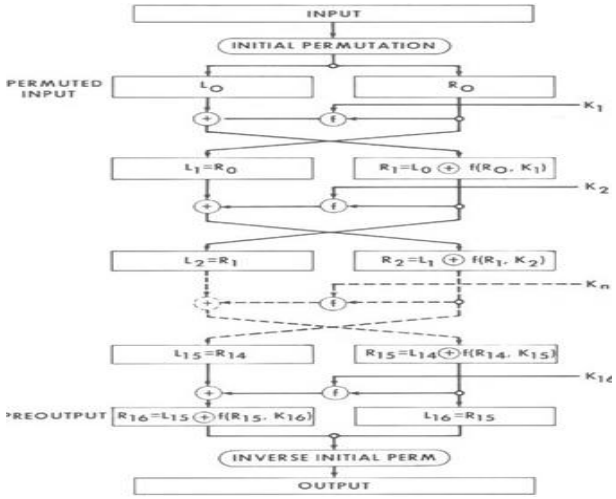

Fig. 3: Proposed Architecture of the Triple Data Encryption Standard (TDES) Encryption System.

Triple Data Encryption Standard (TDES) is a block cipher technique that encrypts 64-bit data blocks by applying the DES cipher algorithm three times in succession. TDES can operate using one, two, or three distinct 56-bit keys, offering flexibility in its approach to security. The sequence of the Triple DES (TDES) algorithm (Fig.3) is depicted, illustrating the steps involved in its encryption scheme.
Encryption formulas for TDES can be described as follows:

$$C=E3_{k1,k2,k3}(P)=E_{k3}(D_{k2}(E_{k1}(P))) \quad (1)$$

Where $E3_{k1,k2,k3}(P)$ is TDES encryption for P using key, k1, k2, and k3. Whereas $E_{kn}(P)$ is DES encryption for P using kn. For $D_{kn}(P)$ is the DES decryption for P using key kn.
The decryption formula for TDES can be described as follows:

$$P=D3_{k1,k2,k3}(C)=D_{k1}(E_{k2}(D_{k3}(C))) \quad (2)$$

Where $D3_{k1,k2,k3}(P)$ 3DES decryption of C using key, are k1, k2, and k3. Whereas $D_{kn}(P)$ is DES decryption for P using kn. For $E_{kn}(P)$ is encryption for P using key kn.

5. Results and Discussions

The proposed method was simulated using MATLAB TOOL to verify the quality of the images, and was also synthesized on an FPGA board to check the area utilization.

5.1.MATLAB simulation

The outcome of the simulation (Fig. 4) demonstrates an X-ray image modified to embed the patient's name and date using the proposed steganography technique [21]. The results indicate that this method can achieve a Peak Signal to Noise Ratio (PSNR) of up to 68.5 dB for the following images. Peak Signal-to-Noise Ratio (PSNR) values for stego-image experiments are calculated using the standard formula (PSNR = 10*$\log_{10}$(MAX^2/MSE)), where (MAX) is the maximum possible pixel value (255 for 8-bit images) and (MSE) is the Mean Squared Error between original and stego images. This ensures reproducibility and allows fair comparison across studies.

Fig.4:(A) X-ray Image

Fig.4:(B) Encrypted Image

Fig.4:(C)Decrypted Image

Table 1: Comparıson of PSNR wıth Exıstıng Technıques

Authors	Technique	Maximum PSNR (dB)
Aya Jaradat et al. [18]	Chaotic Particle Swarm Optimization	60.1
P. K. Muhuri et al. [19]	Integer Wavelet Transform + PSO	52.0
A. H. Mohsin et al. [20]	Particle Swarm Optimization Algorithm	58.0
M. El-Hadedy et al. [26]	Genetic Algorithm-Based Steganography	54.3
S. L. Chandel et al. [27]	LSB with Edge Detection	61.7
T. A. Khan et al. [28]	Hybrid DWT-DCT Approach	56.9
Proposed (TDES-Based)	Triple DES + Steganography (Proposed)	68.5

The graph plotted (Fig.5) compares the PSNR values for all the listed techniques. The comparison of proposed method is compared with the existing techniques (Table 1). It is observed that our method obtains the maximum PSNR.

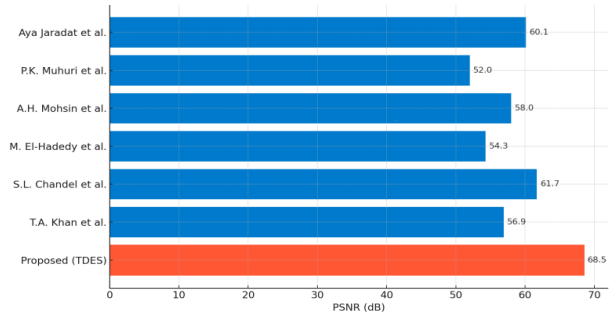

Fig. 5: Comparison of PSNR (dB) values for different existing methods and the proposed TDES-based method.

5.2.Hardware

The suggested steganography architecture was developed on the Digilent ATLYS FPGA platform, utilizing the System Generator tool for design and VHDL (VHSIC Hardware Description Language) for coding. To evaluate the performance of this architecture, a combination of simulation and synthesis results was analyzed, along with comparisons to various existing techniques. The Xilinx ISE (Integrated Software Environment) 14.5 tool was employed for simulation purposes, while synthesis was conducted using the Xilinx Synthesis Technology (XST) tool. The simulation waveforms of the FPGA-based TDES implementation are shown (Fig. 6). The signals represent

input plaintext, the applied key, and the resulting ciphertext after encryption. Timing markers demonstrate the correct functionality of the iterative loop process and confirm that the design meets the expected throughput and latency specifications.

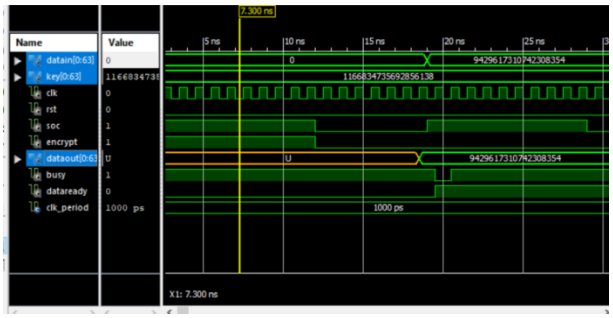

Fig. 6: Simulation Result of DES Block.

5.3.RTL schematic

The architecture's gate-level connections, along with the enhanced model generated by the Xilinx tool, are shown in terms of conventional elements such as adders, multipliers, counters, AND gates, and OR gates (Fig. 7).

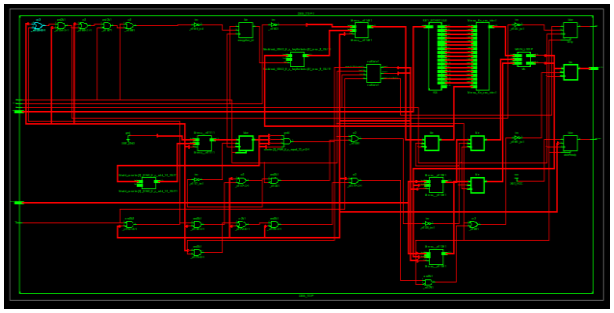

Fig. 7: RTL Schematic of the Proposed Architecture.

Table 2: Hardware Utılızatıons

Parameters	Hardware Utilizations
FPGA	Spartan-6
Slice Registers	185
Slice LUTs	401
Memory	8
Occupied Slices	185
LUT-FF	124

5.4 Comparison with existing techniques

The hardware comparison with the existing techniques is illustrated in Table 3. Compared to the methods described by [24], [25], [29], [30], slice registers and LUT FF are better.

Table 3: Hardware Performance Metrics of the FPGA-Based TDES Implementation Compared to Related Works.

Parameters	Subhi [24]	Shraddha [25]	Rajeev [29]	Nair [30]	Proposed
FPGA	Spartan-3E	—	Spartan-3A	Spartan-6	Spartan-6
Slice Registers	2143	—	1980	1200	185
Slice LUTs	—	1970	—	—	401
LUT-FF Pairs	1160	—	1400	890	124
Frequency (MHz)	167.44	200	190	198.2	203.93

The bar graph (Fig.8) compares FPGA parameters across designs by existing authors and the proposed design. The proposed design shows significant reductions in hardware resource usage (Slice Registers, LUTs, LUT-FF Pairs) compared to other designs. Despite minimal resource usage, it achieves the highest frequency (203.93 MHz), indicating an efficient and optimized design. This reflects better performance with lower hardware costs, making the Proposed Design superior in terms of both efficiency and speed.

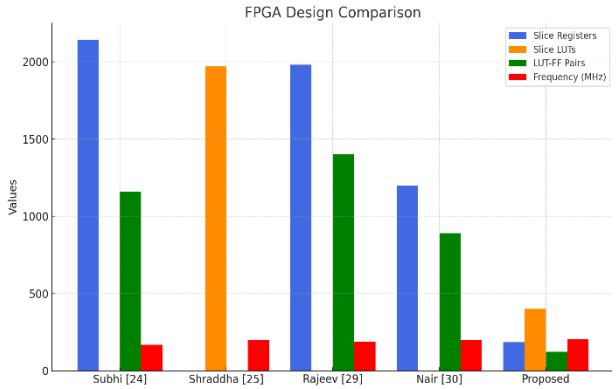

Fig. 8: FPGA Design Comparison showing Slice Registers, Slice LUTs, LUT/FF Pairs, and Frequency (MHz) across existing and proposed designs.

6. Conclusion

The proposed FPGA-based implementation of the Triple Data Encryption Standard (TDES) demonstrates high-speed performance with minimal hardware resource utilization, making it suitable for diverse cryptographic systems. Comparative results with prior works confirm favorable outcomes in throughput, logic resource efficiency, and scalability.

Limitations

While TDES offers strong security, it is computationally heavier compared to modern standards such as AES. Additionally, FPGA-based cryptographic systems may be vulnerable to side-channel attacks, requiring further investigation into countermeasures.

Practical Implications for Teleradiology

In teleradiology applications, secure and efficient image transmission is critical. The proposed design provides:

- Confidentiality: Protection of sensitive patient data during transmission.
- Low hardware cost: Resource-efficient FPGA implementation supports deployment in cost-sensitive medical systems.
- Real-time performance: Encryption of large image files without latency, supporting rapid diagnosis.
- Adaptability: FPGA reconfigurability allows migration to future cryptographic standards, extending system longevity.

References

[1] C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "Robust and Imperceptible Image Watermarking by DC Coefficients Using Singular Value Decomposition," *International Conference on Electrical Engineering, Computer Science, and Informatics (EECSI), Yogyakarta*, 2017.https://doi.org/10.1109/EECSI.2017.8239107.

[2] A. Susanto, D. R. I. M. Setiadi, C. A. Sari, and E. H. Rachmawanto, "Hybrid Method using HWT-DCT for Image Watermarking,"*International Conference on Cyber and IT Service Management (CITSM)*, Denpasar, 2017.https://doi.org/10.1109/CITSM.2017.8089252.

[3] A. Setyono, D. R. I. M. Setiadi, and Muljono, "StegoCrypt Method using Wavelet Transform and One-Time Pad for Secret Image Delivery,"*International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, Semarang, pp. 203–207, 2017.https://doi.org/10.1109/ICITACEE.2017.8257703.

[4] D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari, "Secure Image Steganography Algorithm Based on DCT with OTP Encryption," *Journal of Applied Intelligent System*, vol. 2, no. 1, pp. 1-11, 2017.https://doi.org/10.33633/jais.v2i1.1330.

[5] U. Sudibyo, F. Eranisa, E. H. Rachmawanto, D. R. I. M. Setiadi and C. A. Sari, "A Secure Image Watermarking using Chinese Remainder Theorem Based on Haar Wavelet Transform,"*International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, Semarang, 2017.https://doi.org/10.1109/ICITACEE.2017.8257704.

[6] E. H. Rachmawanto, R. S. Amin, D. R. I. M. Setiadi and C. A. Sari, "A Performance Analysis StegoCrypt Algorithm based on LSB-AES 128 bit in Various Image Size," *International Seminar on Application for Technology of Information and Communication (ISemantic)*, Semarang, 2017.https://doi.org/10.1109/ISEMANTIC.2017.8251836.

[7] W. Stallings, Cryptography and Network Security: *Principles and Practice*, London: Pearson, 2014.

[8] A. Verma and R. Kaur, "A combined cryptography and steganography approach using bit-matching for grayscale and color images," in *Proc. Int. Conf. Advances in Computing, Communication and Security (ICACCS)*, Coimbatore, India, Dec. 2019, pp. 145–150.

[9] S. Arora and S. Anand, "A New Approach for Image Steganography using Edge Detection Method," International *Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 3, pp. 626-629, 2013.

[10] D. Mehta and S. Reddy, "Comparative analysis of edge detection operators for steganographic embedding," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, Abu Dhabi, UAE, Oct. 2020, pp. 2054–2058.

[11] H. Al-Dmour and A. Al-Ani, "A steganography embedding method based on edge identification and XOR coding," *Expert Systems with Applications*, vol. 46, pp. 293-306, 2016.https://doi.org/10.1016/j.eswa.2015.10.024.

[12] K. Shah, E. H. Rachmawanto, and D. R. I. M. Setiadi, "Data confidentiality and privacy preservation using DES with DWT-based steganography," in *Proc. Int. Conf. Electrical Engineering, Computer Science and Informatics (EECSI)*, Yogyakarta, Indonesia, Sept. 2017, pp. 1–6.

[13] S. Mungmode, A. Patel, and R. Sharma, "Edge adaptive steganography using threshold and LSBMR," *Int. J. Computer Applications*, vol. 175, no. 3, pp. 25–31, Oct. 2020.

[14] P. Gupta and N. Singh, "Canny edge detection-based randomized embedding for steganography," *Journal of Information Security and Applications*, vol. 56, pp. 102–113, May 2021.

[15] Y. Zhang, X. Liu, and H. Wang, "Reconfigurable FPGA-based lightweight block cipher framework for IoT security," *IEEE Internet of Things Journal*, vol. 10, no. 8, pp. 6550–6562, Apr. 2023.

[16] A. Kumar and R. Bansal, "Hybrid cryptography–steganography system for secure telemedicine applications," *IEEE Access*, vol. 12, pp. 112340–112352, Sept. 2024.

[17] J. Lee, M. Tanaka, and P. Singh, "Quantum-resistant cryptography integrated with image steganography for secure multimedia transmission," *IEEE Transactions on Information Forensics and Security*, vol. 20, no. 1, pp. 985–996, Jan. 2025.

[18] AyaJaradat, Eyad Taqieddin and Moad Mowafi, "A High-Capacity Image steganography Method Using Chaotic Particle Swarm Optimization", *Security and Communication Networks*, Hindawi, pp. 1-11, 2021.https://doi.org/10.1155/2021/6679284.

[19] P. K. Muhuri, Z. Ashraf and S. Goel, "A novel image steganographic method based on integer wavelet transformation and particle swarm optimization," *Applied SoftComputing*, Elsevier, vol. 92, pp. 1-41, 2020.79https://doi.org/10.1016/j.asoc.2020.106257.
[20] A.H. Mohsin, A. A. Zaidan, B. B. Zaidan, O. S. Albahri, A. S. Albahri, M. A. Alsalem, K. I. Mohammed, OdaiEnaizan, ShahadNidhal, A. H. Alamoodi, N. S. Jalood, Ali Najmjasim, Ali. H. Shareef, E. M. Almahdi, M. J. Baqer, H. A. Ameen and Salem Garfan "New method of image steganography based on particle swarm optimization algorithm in spatial domain for high embedding capacity," *IEEE Access*, vol. 7, pp. 168994–169010, 2019.https://doi.org/10.1109/ACCESS.2019.2949622.
[21] P. Marakumbi and S. Bhairannawar, "9/7 LIFT Reconfigurable Architecture Implementation for Image Authentication", *IJRITCC*, vol. 11, no. 7, pp. 23–31, Sep. 2023.https://doi.org/10.17762/ijritcc.v11i7.7826.
[22] Mohammed, Z. A.; Gheni, H. Q.; Hussein, Z. J.; Al-Qurabat, A. K. M. Advancing Cloud Image Security via AES Algorithm Enhancement Techniques. *Eng. Technol. Appl. Sci. Res.* 2024, *14*, 12694-12701. https://doi.org/10.48084/etasr.6601.
[23] Charls H. Roth (Jr.), "Digital System Design using VHDL", *Cengage Learning*, 2006.
[24] Subhi R. M. Zeebaree, "DES encryption and decryption algorithm implementation based on FPGA", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 18, No. 2, May 2020, pp. 774~781.https://doi.org/10.11591/ijeecs.v18.i2.pp774-781.
[25] Shraddha P Tankasali, SunitaShirahatti, Thejaswini P, " Performance Analysis of DES and Triple DES Algorithm*", International Research Journal of Engineering and Technology*, Volume: 07 Issue: 10 , Oct 2020.
[26] M. El-Hadedy, M. M. Fouad, A. M. Khalifa, "Genetic Algorithm-Based Steganography for High Capacity and Security", *International Journal of Computer Applications*, Volume: 98 Issue: 4 ,July 2014.
[27] S. L. Chandel, R. K. Jha, "Secure Image Steganography using LSB and Edge Detection Technique", *International Journal of Engineering Research & Technology (IJERT)*, Volume: 3 Issue: 6 , June 2014.
[28] T. A. Khan, R. K. Tiwari, "A Hybrid DWT-DCT Approach for Digital Image Steganography", *International Journal of Advanced Research in Computer Science and Software Engineering*, Volume: 3 Issue: 7 , July 2013.
[29] Rajeev Kumar, R. S. Anand, "Design and Performance Analysis of High-Speed Low-Power FPGA Architecture using Spartan-3A", *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, Volume: 4 Issue: 5 , May 2015.
[30] Nair, A. P., K. S. Shivaprasad, "Optimization of FPGA Resource Utilization using Spartan-6 for High-Performance Applications", *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, Volume: 2 Issue: 3 , June 2017.