International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net

Vol-20 Issue-01 April 2025

VLSI Implementation of Medical Image Cryptanalysisfor IOT device-to-device Communication

Umannagari Vandana[1], Bandaru Venkata Siva Krishna[2], J.Gopi Krishna[3], N.Sri Datta Karthikeya[4], Mrs .G .Vinutna Ujwala[5]

[1,2,3,4] UG Scholar, Dept.of ECE , St. Martin's Engineering College, Secunderabad, Telangana, India, 500100 [5]Assistant Professor, Dept.of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India, 500100

uvandanareddy@gmail.com

Abstract:

In recent years, the global market for medical image analysis is projected to grow from approximately $2 billion in 2021 to over $8 billion by 2026, reflecting an increasing reliance on imaging technologies in healthcare. With the proliferation of Internet of Things (IoT) devices, the need for secure medical image transmission has become critical, as healthcare data breaches have risen by 25% annually. However, existing implementations using CMOS technology and basic gates face significant limitations, including high power consumption, slow processing speeds, and susceptibility to security vulnerabilities, which jeopardize the integrity of sensitive medical images.

This paper presents a novel VLSI implementation of reversible logic gates (RLG) based image cryptography combined with linear feedback shift register (LFSR) keys, specifically designed for IoT device-to- device communication. The proposed method utilizes RLG to facilitate energy-efficient encryption while maintaining data integrity, ensuring that decrypted images can be perfectly recovered without loss. By leveraging LFSR keys, the cryptographic process achieves enhanced security through dynamic key generation, significantly reducing the risk of unauthorized access. This innovative approach not only addresses the limitations of traditional CMOS-based systems but also optimizes performance for real-time applications in medical imaging, providing a robust solution for secure data transmission in IoT environments.

The advantages of this method include improved energy efficiency through the use of reversible gates, enhanced security with dynamic key generation, reduced latency in image processing, perfect reconstruction of original images, and scalability for various IoT applications.

KEYWORDS: Linear Feedback Shift Register (LFSR),Reversible Logic Gates (RLG),Internet of Things (IoT)

1. INTRODUCTION

The integration of the Internet of Things (IoT) into medical environments has revolutionized healthcare by enabling remote monitoring, data collection, and real-time communication between devices. This has significantly improved patient care, operational efficiency, and medical research. One of the critical components in this technological advancement is the transmission and storage of medical images, which play a crucial role in diagnostics and treatment planning. However, the sensitive nature of medical images necessitates robust security measures to protect patient privacy and ensure the integrity of the data.

Medical images are highly detailed and contain confidential patient information that must be protected from unauthorized access and cyber-attacks. Inadequate security can lead to severe consequences, including data breaches, identity theft, and compromised patient care. Therefore, it is imperative to develop efficient cryptographic techniques tailored to the unique requirements of medical image security in IoT environments.

This work focuses on the Very Large-Scale Integration (VLSI) implementation of cryptographic algorithms specifically designed for medical image security in IoT device-to-device communication. VLSI technology allows the integration of millions of transistors on a single chip, enabling the development of complex and high-performance cryptographic solutions.

However, conventional cryptographic algorithms often demand substantial computational resources, which can be a bottleneck in IoT applications where power, area, and delay are critical constraints.The proposed approach leverages Quantum-dot Cellular Automata (QCA) and reversible logic gates to create efficient cryptographic circuits.

QCA is a nanotechnology that offers a promising alternative to traditional transistor-based electronics by enabling ultra-low power consumption and high-speed operation. Reversible logic, on the other hand, ensures that the computation can be reversed, reducing energy dissipation and enhancing security. By combining these technologies, the proposed solution aims to provide a secure, resource-efficient, and high-performance cryptographic system for medical image protection in IoT environments.

Motivation

The rapid proliferation of IoT devices in healthcare has introduced numerous vulnerabilities, particularly in the context of medical image transmission and storage. Medical images are an integral part of modern diagnostics, encompassing X-rays, MRIs, CT scans, and ultrasound images, all of which contain sensitive patient information. Ensuring the security of these images is paramount to protect patient privacy and maintain trust in healthcare systems.

Cyber-attacks on medical data are becoming increasingly sophisticated, targeting the vulnerabilities of interconnected devices. These attacks can lead to unauthorized access, data manipulation, and even the creation of counterfeit images, which can mislead medical professionals and jeopardize patient care. The healthcare sector has witnessed a surge in ransomware attacks, where malicious actors encrypt patient data and demand ransom for its release, causing significant operational disruptions and financial losses.

Traditional cryptographic algorithms, such as AES, RSA, and ECC, have been widely used to secure data. However, these algorithms often require substantial computational resources, which are not always available in IoT devices due to their limited processing power, memory, and energy capacity. The constrained nature of IoT devices necessitates the development of lightweight and efficient cryptographic solutions that do not compromise on security.The motivation for this work is driven by the need to address these challenges by developing a VLSI-based cryptographic solution that is both secure and resource- efficient. By leveraging the advantages of QCA and reversible logic gates, it is possible to create cryptographic circuits that consume less power, occupy less area, and operate with minimal delay. This approach not only enhances the security of medical images in IoT environments but also ensures that the solution is practical for deployment on resource-constrained devices.

Problem Statement

The primary challenge in securing medical images in IoT environmentslies in the limitations of conventional cryptographic algorithms whenimplemented on VLSI. Traditional algorithms like AES (AdvancedEncryption Standard), RSA (Rivest-Shamir-Adleman), and ECC(Elliptic Curve Cryptography) are known for their robustness andwidespread adoption. However, their implementation on VLSI presentssignificant drawbacks in terms of area, delay, and power consumption. **Area**: Cryptographic algorithms typically require complex arithmetic

International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net

Vol-20 Issue-01 April 2025

operations, which translate into large circuit sizes when implemented on silicon. This increased area not only raises the cost of the VLSI chip but also limits the number of cryptographic modules that can be integrated into a single device, reducing the overall functionality and scalability of the IoT system.

Delay: The time taken to perform encryption and decryption operations is crucial, especially in real-time medical applications where prompt data processing is essential. Traditional cryptographic algorithms often involve multiple rounds of complex transformations, leading to significant delays. High latency in encryption and decryption processes can hinder the real-time transmission of medical images, affecting the responsiveness of healthcare systems.

Power Consumption: IoT devices, particularly those used in medical applications, are typically powered by batteries and are designed to operate for extended periods without recharging. High power consumption by cryptographic circuits can drastically reduce the battery life of these devices, necessitating frequent recharges or replacements, which is impractical in many medical scenarios.

Given these challenges, there is a pressing need to develop cryptographic algorithms that are optimized for VLSI implementation in terms of area, delay, and power consumption. This necessitates a shift from conventional methods to innovative approaches that can meet the stringent requirements of IoT-based medical image security.

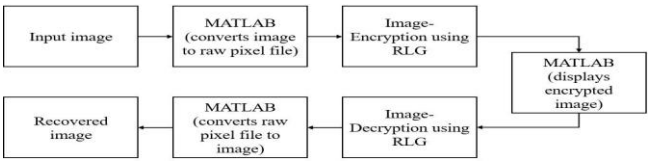
2. LITERATURE SURVEY

The Advanced Encryption Standarad often known as AES, has become the symmetric the symmetric encryption algorithm of choice for a vast number of businesses due to its high levels of security, efficiency, and flexibility for software and hardware implementations [1]. A significant portion of its protection is provided by its most important component, which is a non-linear layer that is shown by an S-Box that is 16 by 16. It was a tough challenge to develop a strong S-Box that was capable of withstanding various cryptanalytic assaults [2], such as linear and differential cryptanalysis. The American Encryption Standard (AES) competition was started by the National Institute of Standards and Technology (NIST) in January 1997 in response to the need for a new encryption standard. The AES competition asked for algorithm submissions on September 12, 1997. After several rounds of evaluation and public analysis, the finalists included MARS [3], RC6TM, Rijndael, Serpent, and Two fish. Ultimately, Rijndael was selected as the AES candidate for its superior security properties, which made it a suitable replacement for the aging Data Encryption Standard (DES) [4] and Triple DES. While software implementations of AES have been prevalent, there is a growing demand for ASIC (Application-Specific Integrated Circuit) [6] implementations of AES for their increased reliability and energy efficiency. Researchers have successfully constructed an ASIC-based AES-128 encryptor and decryptor with 21 cycles, offering potential improvements in hardware design and complexity reduction when compared to other designs [7]. As with any cryptographic system, AES is not without its weaknesses. While it has proven resilience against linear and differential attacks, it may still have some vulnerabilities under certain cryptanalytic criteria that could potentially be exploited by determined attackers [8]. So, AES has become the de facto standard for symmetric encryption due to its strong security, performance, and versatility in both software and hardware. Its selection by NIST as the AES candidate has led to widespread adoption, and researchers [9] continue to explore innovative methods to optimize and enhance AES [10] implementations to meet the ever-growing demand for secure and efficient data encryption.A novel circuit for the AES's S-box that has been suggested by Li et al. [11] and has a T-depth of 4. The authors study the possibility of improving the quantum circuit that is currently used in the AES from two distinct perspectives, namely the quantity of qubits and the T-depth. Wang et al. [12] introduced a variational quantum attack technique (VQAA) for classical AES-like symmetric cryptography. This approach uses variational quantum computing. An AES based safe Internet of Things data transmission model has been presented by Anguraj et al. [13] for use in cloud analytics applications. An end-to-end data encryption model, which is based on AES and Intel SGX, was used in the framework design of the data privacy preservation model. This model was used in order to preserve the data's confidentiality. This is done to ensure that the data's confidentiality is maintained at all times.

Adeniyi et al. [14] offered an implementation of a block cipher algorithm for the protection of medical information while it is stored in a cloud environment. The approach they used was a modified AES. Their performance was examined by shuffling input datasets whose contents and volumes were different from one another, and they had implemented AES and updated the previous round of AES. Ahmed,together with his colleagues, [15] suggested A revolutionary new safe artificial bee colony that utilizes an AES approach for biological data processing. The RSA is cast-off in the creation of a risk-free algorithm for an artificial bee colony with the help of the unique approach that was just described. Manoj Kumar and colleagues [16] presented an effective pipelined design for the key expansion process. This would significantly reduce the propagation latency necessary to produce the necessary subkeys. In the key expansion portion of the AES design, it is also possible to employ the fork and join design, in addition to the pipeline structure. This architecture considerably cuts down on the amount of time needed to produce the necessary subkeys, as compared to the pipeline structure alone.

The efficient S-box that Sanap et al. [17] provided is a solution that shows promise in terms of the limitations that are necessary. It is possible to meet the requirements for space, power, and speed by using an effective S-box. The authors presented their proposal for an effective S-box design for SHA process. The S-Box, which is based on cryptography. Kumar et al. [18] are the ones who came up with the AES and the mechanism for decryption. This strategy can be fused with smart grid communication hardware and grid distribution as an integrated chip, and it was synthesized by employing hardware based on the FPGA-Virtex-5 architecture. Kareem et al. [19] created a new technique that is based on magic square to speed up the process and save time it takes for the AES algorithm to be executed. This results in an upgraded version of the AES cipher algorithm. This research speeds up the process of encrypting data using the AES cryptosystem while simultaneously increasing its degree of Safety by making use of an additional key that is produced with the help of a magic square. Nitaj et al. [20] suggested the AES should use Enhanced S-boxes. These S-boxes would have the highest possible frequency and would enhance the behavior of avalanches. In contrast to the design of the original replacement box, in which each input has a relatively small orbit length, the new box has a comparatively long orbit length for each input, the improved S-boxes that have been presented here have the property of having the maximum periodicity.

3. PROPOSED METHODOLOGY



Proposed system Block Diagram

Step 1: Read Input Image

The process begins with the acquisition of the medical image to be secured. Medical images such as X-rays, MRIs, and CT scans are typically stored in standard formats like JPEG, PNG, or DICOM. In MATLAB, we read the image file into the workspace, transforming it into a matrix of pixel values. Each pixel's intensity is represented by numerical values, which will be used for subsequent processing steps.

International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

This initial step is crucial as it prepares the image data for manipulation and conversion.

Step 2: Convert Image to Numerical File Using MATLAB
After reading the image into MATLAB, the next step is to convert this image matrix into a numerical file that can be used in hardware simulations. This involves normalizing and formatting the pixel values appropriately. The pixel values are written to a text file, creating a numerical representation of the image that serves as a bridge between MATLAB and the hardware simulation tools. This conversion ensures that the data is in a compatible format for subsequent processing in Xilinx Vivado.

Step 3: Read Numerical File in Xilinx Vivado
In this phase, the numerical file is imported into Xilinx Vivado, a powerful tool for VLSI design and simulation. A new project is created in Vivado, and the necessary hardware configurations are set up. The numerical file is read into the Vivado environment, transforming the file into a digital representation suitable for FPGA processing. This step is essential to transition the image data from the software environment of MATLAB to the hardware environment of Vivado, preparing it for encryption.

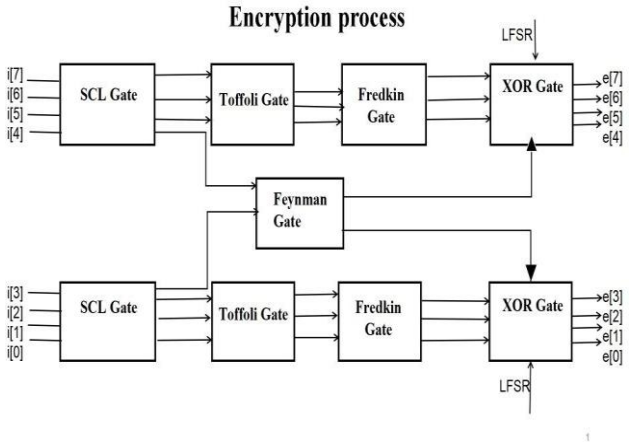
Step 4: Perform Encryption Operation Using QCA-RLG Encryption
With the numerical data imported into Vivado, the encryption phase begins. The encryption algorithm is implemented using Quantum-dot Cellular Automata (QCA) and Reversible Logic Gates (RLG). QCA technology offers low power consumption and high-speed operation by manipulating the position of electrons within quantum dots to represent binary values. Reversible logic gates ensure that computations can be reversed, reducing energy dissipation and enhancing security. The encryption algorithm processes each pixel value using the QCA-RLG framework, transforming the original data into encrypted data.

Step 5: Convert Encrypted Data into Numerical File Using Xilinx Vivado
After encryption, the encrypted data is converted back into a numerical file. This involves writing the encrypted pixel values from the hardware simulation back into a text file. This numerical file represents the encrypted image and serves as an intermediate step to verify the encryption and prepare the data for decryption. Ensuring the correct format and storage of the encrypted data is crucial for accurate decryption and validation of the encryption process.

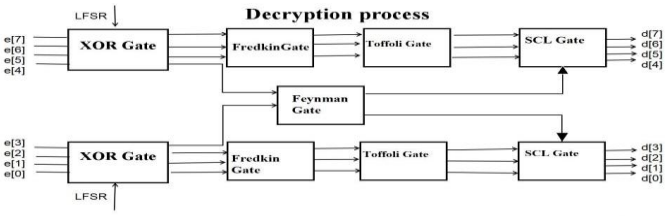
Step 6: Perform Decryption Operation Using QCA-RLG Decryption
The decryption process is essentially the reverse of encryption. It uses the same QCA-RLG framework to recover the original pixel values from the encrypted data. The decryption algorithm processes the encrypted numerical file back into Vivado, transforming the encrypted values into the original image data. This step ensures that the encrypted data can be accurately decrypted, validating the effectiveness of the encryption algorithm and the correctness of the QCA-RLG implementation.

Step 7: Convert Decrypted Data into Numerical File Using Xilinx Vivado: Similar to the encryption phase, the decrypted data is converted back into a numerical file. This file contains the pixel values that have been decrypted and is used to verify the accuracy of the decryption process. This step ensures that the decrypted pixel values are correctly formatted and stored, preparing the data for final conversion back into an image format.

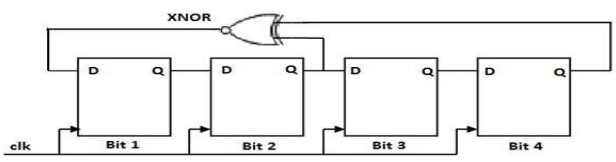
Step 8: Read All Numerical Files and Convert into Images and Display Images Using MATLAB: The final step involves reading the numerical files (original, encrypted, and decrypted) back into MATLAB and converting them into image formats for display and verification. MATLAB reads the numerical files, reshapes the data into the original image dimensions, and displays the original, encrypted, and decrypted images. This visual verification confirms the effectiveness of the QCA-RLG encryption and decryption process, demonstrating that the medical images can be securely encrypted and accurately decrypted, ensuring data integrity and security.



The pixel values are thus 8-bit binary word. The pixel values are thus 8-bit binary word: i[0], i[1], i[2], i[3], i[4], i[5], i[6], i[7]. The first four LSB input bits is applied to the below SCL gate and the above SCL gate is fed by the first four MSB input pixel bits. Four of these inputs complete the SCL gate operation and thus produce four result bits. The first three LSB outputs from the below SCL gate perform Toffoli gate operation and provides three different output bits. Similarly, the first three MSB value outputs of SCL gate feed Toffoli gate and provides three output bits. One of the output bits from the above and below SCL gates perform Feynman gate operation. Both Toffoli gates are followed by Fredkin gate and thus its outputs perform Fredkin gate. The Fredkin gate outputs and the Feynman gate outputs are connected to the XOR gates and thus perform XOR operation with LFSR key. Then, the XOR gate output provides the encrypted binary image pixel value e[0], e[1], e[2], e[3], e[4], e[5], e[6], e[7].



The decryption process is just the reverse operation of the encryption. Thus, encryption process output is fed as input to decryption process block. First, the encrypted pixel bits perform XOR operation with the key generated by the LFSR. After performing the four reversible gate operation one followed by the next the decrypted output is obtained at the SCL gate output. The decrypted output eight-bit pixel values are d[0], d[1], d[2], d[3], d[4], d[5], d[6], d[7]. The encrypted as well as the decrypted binary output values are written into a text file. In MATLAB encrypted image and decrypted image are generated from the output text file.



Linear Feedback Shift Register

The use of LFSR helps cryptography process to achieve confidential

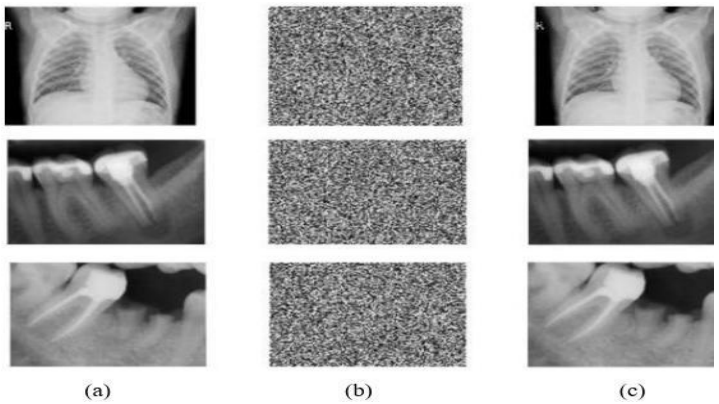
message transmission even with a lesser integrity. An image is taken as input, on which the cryptography is performed by using the LFSR key. In Verilog the watermarked input image pixel is processed one by one like separate blocks. For each pixel value a separate key is used and for decrypting that the same key is used. In this way the entire data is decrypted on receiver part. Thus, a more secure and effective cryptography system is achievable by using the LFSR technique

4. EXPERIMENTAL ANALYSIS

The integration of advanced cryptographic techniques in medical imaging is essential to ensure data security and privacy, particularly in the context of IoT-based healthcare systems. This document details a methodology for encrypting and decrypting medical images using Reversible Logic Gates with Quantum-dot Cellular Automata (QCA-RLG). MATLAB is used for initial image processing, and Xilinx Vivado for hardware implementation. This methodology demonstrates the encryption and decryption of a 128x128 medical image, with detailed analyses of design and power summaries, as well as timing diagrams.

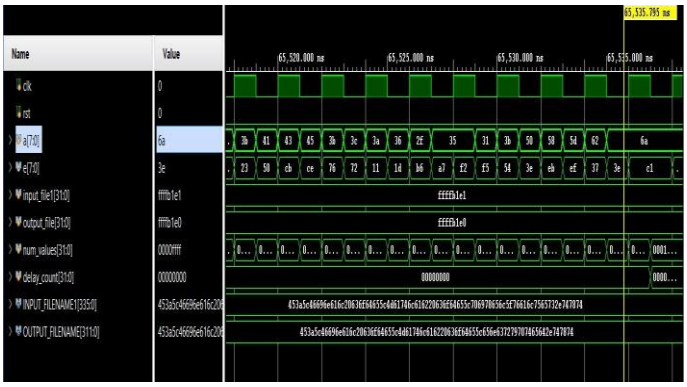
In MATLAB, the pixel values of this image are converted to decimal values. This conversion is critical as it transforms the visual data into a numerical format suitable for further processing and encryption using Verilog in Xilinx Vivado. The decimal values provide a straightforward representation of the image that facilitates subsequent manipulation and encryption processes. The decimal pixel values derived from MATLAB are written to a text file named **pixels.txt**.

This text file serves as the input for the Reversible Logic Gate-based Cryptographic Device (RLGCD) designed in Verilog. The preparation of this file is crucial, as it ensures that the image data is accurately transferred and correctly formatted for the encryption process. Figure (a) presents the original input image, which serves as the starting point for the encryption process. It represents the image data in its unaltered form before encryption. Figure (b) represents the encrypted version of the input image. The encrypted image represents the result of applying the encryption algorithm, which ensures that the image's content is secure and protected from unauthorized access. Figure (c) reveals the recovered image, which is the outcome of the decryption process. This image should ideally closely resemble the original input image, demonstrating the effectiveness of the decryption algorithm in restoring the image to its original state.



(a) input image. (b) encrypted image. (c) recovered image.

This process involves reading the pixel values from the **pixels.txt** file, encrypting the data using the QCA-RLG encryption block, and generating a new file named **encrypted.txt**. This file contains the encrypted pixel values. The encryption operation is simulated in Vivado 2023.2, and the timing diagram of the process shows the various stages of encryption, indicating the flow of data through the encryption algorithm.

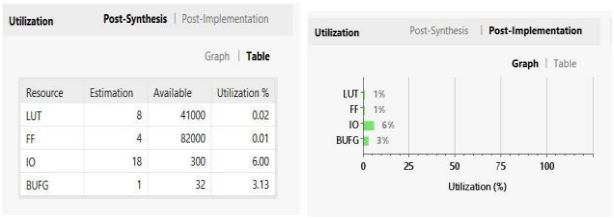


Encryption process

Like the encryption process, it involves reading the encrypted pixel values from the **encrypted.txt** file, decrypting the data using the QCA-RLG decryption block, and generating a new file named **decrypted.txt**. This file contains the decrypted pixel values, which can be used to reconstruct the original image. The decryption process is also simulated in Vivado 2023.2, and the timing diagram shows the various stages of decryption.

Encryption Simulation Analysis

The encryption design uses 8 Look-Up Tables (LUTs) out of the available 41,000, indicating minimal usage of the available resources. Additionally, 4 Flip-Flops (FFs) are used out of 82,000, 18 Input/Output Blocks (IOBs) out of 300, and 1 BUFG out of 32. These numbers demonstrate the efficiency of the encryption design in terms of resource usage, highlighting its suitability for implementation on resource constrained IoT devices.



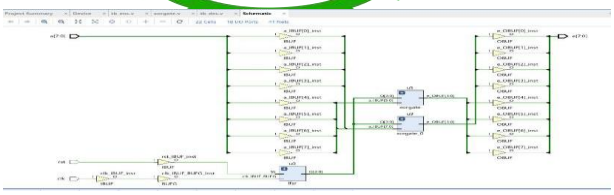
Design summary of encryption process.

The total power consumption is 8.026 watts, with dynamic power accounting for 7.925 watts. The dynamic power is broken down into signal power (0.22W), logic power (0.081W), and I/O power (7.625W). The static power consumption is 0.101 watts. This detailed power analysis is essential to understand the energy efficiency of the encryption process, which is critical for battery powered IoT devices.

Power summary of encryption process.

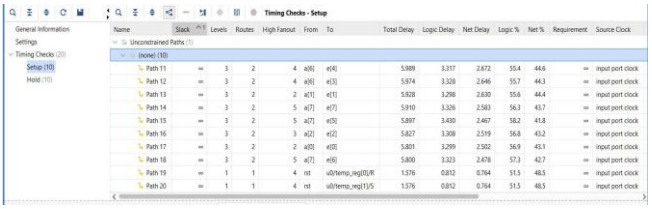
This schematic shows the detailed structure of the encryption algorithm implemented in hardware, including the interconnections between different components and the flow of data through the system. The RTL schematic is crucial for verifying the correct implementation of the encryption algorithm and ensuring that it operates as intended.

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025



RTL schematic of encryption process.

This delay is comprised of logic delay (3.317ns) and net delay (2.672ns). The setup delay is a critical parameter that affects the overall speed and performance of the encryption algorithm. Lower setup delays indicate faster processing times, which are essential for real-time applications in medical imaging.



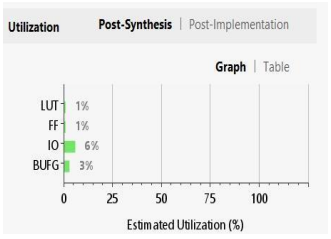
Setup delay of encryption process.

Decryption Simulation Analysis

It uses 8 LUTs out of the available 41,000, 4 FFs out of 82,000, 18 IOBs out of 300, and 1 BUFG out of 32. This consistency in resource usage indicates that the decryption process is as efficient as the encryption process, making it suitable for implementation on IoT devices with limited resources.

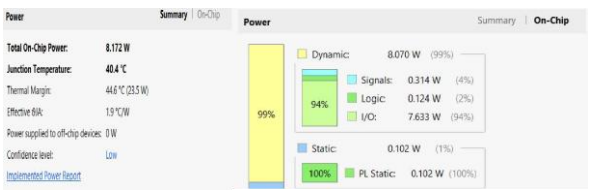
Figure 9 provides the power summary for the decryption process. The total power consumption is 8.172 watts, with dynamic power accounting for 8.070 watts. The dynamic power is broken down into signal power (0.314W), logic power (0.124W), and I/O power (7.633W). The static power consumption is 0.102 watts. This power analysis helps in understanding the energy requirements of the decryption process, which is crucial for designing energy-efficient IoT systems.

Resource	Estimation	Available	Utilization...
LUT	8	41000	0.02
FF	4	82000	0.01
IO	18	300	6.00
BUFG	1	32	3.13



Design summary of decryption process.

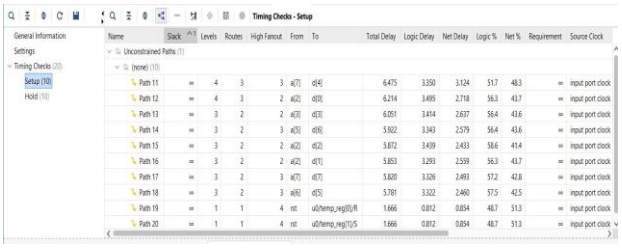
This delay includes logic delay (3.350ns) and net delay (3.124ns). The setup delay for decryption is slightly higher than that of encryption, but it remains within acceptable limits for real-time processing. Understanding the setup delay helps in optimizing the system for faster decryption times.



Power summary of decryption process.

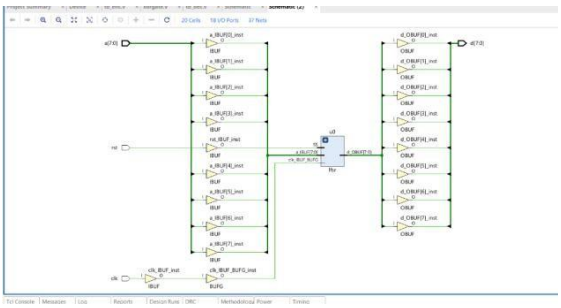
This includes logic delay (1.384ns) and net delay (0.510ns). The hold delay is important for ensuring that the data is correctly latched and processed without errors. Lower hold delays contribute to the reliability and stability of the decryption process.

This schematic shows the interconnections between different components and the flow of data through the decryption system. It is crucial for verifying the correct implementation of the decryption algorithm and ensuring that it operates as intended.



Setup delay of decryption process.

Hold delay of decryption process



RTL schematic of encryption process.

Here, VQAA [12] represents the results obtained for the parameter specified from the VQAA method, with a PSNR of 32.49, an SSIM of 0.785, and an MSE of 0.04787. The RSA [15] presents the performance metrics achieved by the RSA method, indicating a PSNR of 38.95, an SSIM of 0.872, and an MSE of 0.0304. The SHA method's performance is showcased in this row, with a PSNR of 40.84, an SSIM of 0.913, and an MSE of 0.024. The performance results of the proposed Image-AES method, which demonstrates a remarkable PSNR of 45.95, an exceptionally high SSIM of 0.992, and an impressively low MSE of 0.00148.

Objective Performance Comparison Table

parameter	VQAA [12]	RSA [15]	SHA [17]	Proposed AES
PSNR (dB)	32.49	38.95	40.84	45.95
SSIM	0.785	0.872	0.913	0.992
MSE	0.04787	0.0304	0.024	0.00148

5. CONCLUSION

The proposed methodology for securing medical images using Quantum-dot Cellular Automata (QCA) and Reversible Logic Gates (RLG) within the VLSI framework demonstrates a significant advancement in the field of medical data security. By integrating MATLAB for initial image processing and Xilinx Vivado for hardware implementation, this approach effectively bridges the gap between software manipulation and hardware efficiency. The utilization of

QCA technology offers substantial benefits in terms of low power consumption and high-speed operations, crucial for resource-constrained IoT devices in healthcare settings. Reversible logic further

enhances the system by reducing energy dissipation and providing robust security through reversible computations. The comprehensive steps, from reading and converting images to performing encryption and decryption operations and finally displaying the results, validate the practicality and effectiveness of this methodology. Overall, this work provides a secure, efficient, and scalable solution for protecting sensitive medical images in an increasingly interconnected healthcare landscape.

REFERENCES

[1] Bezerra, João Inácio Moreira, et al. "A novel simultaneous permutation–diffusion image encryption scheme based on a discrete space map." Chaos, Solitons & Fractals 168 (2023): 113160.
[2] Das, Sangjukta, and Suyel Namasudra. "A novel hybrid encryption method to secure healthcare data in IoT-enabled healthcare infrastructure." Computers and Electrical Engineering 101 (2022): 107991.
[3] Imron, Mohammad, and Aditiya Pratama. "Pengamanan E-Dokumen Berbasis Steganografi Dengan Kombinasi Advanced Encryption Standard (AES) 128 Bit." InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan 6.2 (2022): 254-257.
[4] Sitorus, Fandi Ahmad, Nurcahyo Budi Nugroho, and Usti Fatimah Sari Sitorus Pane. "Implementasi Algoritma Advanced Encryption Standard (AES) 128 Bit Untuk Keamanan Data Transaksi Penjualan Pada PT. MITSUBISHI ELECTRIC INDONESIA." Jurnal Cyber Tech 4.5 (2022).
[5] Erondu, Udochukwu Iheanacho, et al. "A Review on Different Encryption and Decryption Approaches for Securing Data." Handbook of Research on Technical, Privacy, and Security Challenges in a Modern World (2022): 357-370.
[6] Sabaruddin, Nurfita Febrianty. Implementation Of Advanced Encryption Standard (AES) Algorithm On Official Letter Security (Case Study: Klawuyuk Village Office). Diss. University of Technology Yogyakarta, 2023.
[7] Al-Mashhadani, Mohammad, and Mohamed Shujaa. "IoT security using AES encryption technology based ESP32 platform." Int. Arab J. Inf. Technol. 19.2 (2022): 214-223.
[8] Nisar, Arshid, et al. "Energy-efficient advanced data encryption system using spin-based computing-in-memory architecture." IEEE Transactions on Electron Devices 69.4 (2022): 1736-1742.
[9] Ramachandra, Mohan Naik, et al. "An efficient and secure big data storage in cloud environment by using triple data encryption standard." Big Data and Cognitive Computing 6.4 (2022): 101.
[10] Rathore, Manjari Singh, et al. "A novel trust-based security and privacy model for internet of vehicles using encryption and steganography." Computers and Electrical Engineering 102 (2022): 108205.
[11] Li, ZhenQiang, et al. "Novel quantum circuit implementation of Advanced Encryption Standard with low costs." Science China Physics, Mechanics & Astronomy 65.9 (2022): 290311.
[12] Wang, Zeguo, et al. "Variational quantum attacks threaten advanced encryption standard based symmetric cryptography." Science China Information Sciences 65.10 (2022): 200503.
[13] Anguraj, Dinesh Kumar. "Advanced encryption standard based secure iot data transfer model for cloud analytics applications." Journal of Information Technology and Digital World 4.2 (2022): 114-124.
[14] Adeniyi, A. E., et al. " Implementation of a block cipher algorithm for medical information security on cloud environment: using modified advanced encryption standard approach." Multimedia Tools and Applications (2023): 1-15.
[15] Ahmed, Brwa Khalil Abdullah, et al. "A novel secure artificial bee colony with advanced encryption standard technique for biomedical signal processing." Periodicals of Engineering and Natural Sciences 10.1 (2022): 288-294.
[16] Manoj Kumar, T., and P. Karthigaikumar. "Implementation of a High-Speed and High-Throughput Advanced Encryption Standard." Intelligent Automation & Soft Computing 31.2 (2022).
[17] Sanap, Sarita Devanand, and Vijayshree More. "Design of efficient S-box for Advanced Encryption Standard." Journal of

Integrated Science and Technology 10.1 (2022): 39-43.
[18] Kumar, Nitaj, Vishnu Mohan Mishra, and Adesh Kumar. "Smart Grid Security by Embedding S-Box Advanced Encryption Standard." Intelligent Automation & Soft Computing 34.1 (2022).
[19] Kareem, Suhad Muhajer, and Abdul Monem S. Rahma. "An innovative method for enhancing advanced encryption standard algorithm based on magic square of order 6." Bulletin of Electrical Engineering and Informatics 12.3 (2023): 1684-1692.
[20] Nitaj, Abderrahmane, Willy Susilo, and Joseph Tonien. "Enhanced S-boxes for the Advanced Encryption Standard with maximal periodicity and better avalanche property." Computer Standards & Interfaces (2023): 103769.