Journal of Electronics and Informatics (ISSN: 2582-3825)
www.irojournals.com/iroei/

Reversible Logic based Cryptography Design Algorithm using Random Keys

S. Rithiga[1], T. Venish Kumar[2], M. Idhayachandran[3], S. Prathap[4]

[1]Student, [2]Associate Professor, [3,4]Assistant Professor

[1,2,3,4] Dept. of Electronics and Communication Engineering, Nadar Saraswathi College of Engineering and Technology, Theni, India.

E-mail: rithigasivakumar@gmail.com[1], venishkumarnscet@gmail.com[2], mchandraan@gmail.com[3],prthpraja@gmail.com[4]

Abstract

Reversible computations, besides quantum computing, have various applications in digital signal processing, nanotechnology and bioinformatics. They are particularly useful in designing low-power devices and improving computational efficiency. Cryptography is vital for protecting sensitive information in fields such as bioinformatics and digital signal processing where private data is frequently exchanged. However, cryptographic algorithms can consume significant power and require large areas, particularly when implemented in hardware. Reversible logic gates offer a potential solution by being more power-efficient and potentially reducing implementation area. Using random numbers as keys for both encryption and decryption in a reversible logic gate-based cryptographic algorithm can enhance security. LSB watermarking is a technique to embed additional metadata into digital media, improving data security. To evaluate the performance of the Field Programmable Gate Array for the Reversible Logic Gate Cryptography Design architecture, comparing it to the other state-of-the-art approach is necessary.

Keywords: Reversible Logic Gate Cryptography Design (RLGCD), Random keys, Field Programmable Gate Array (FPGA), Watermarking

1. INTRODUCTION

Cryptography is the science of securing communications and shielding information from unauthorized access or disclosure [1]. It involves using mathematical algorithms to transform the original data (referred to as plaintext) into an unreadable format (referred to as ciphertext) to maintain confidentiality [2][3]. The process of encryption and decryption is fundamental to cryptographic systems. Encryption is the process of converting plaintext into ciphertext using

DOI: https://doi.org/10.36548/jei.2023.2.004
Received: 02.04.2023, received in revised form: 02.05.2023, accepted: 17.05.2023, published: 27.05.2023

an encryption algorithm and a secret key. Decryption, is the process of converting ciphertext back into plaintext using a decryption algorithm and the correct key.

Heat dissipation is a significant challenge in Very Large Scale Integration (VLSI) design, especially as the size of Integrated Circuits (ICs) continues to reduce and the number of transistors increases in accordance with Moore's law [4]. As ICs become smaller and more densely packed with transistors, the power density increases, resulting in higher heat dissipation. Landauer's work is based on the principles of information theory and thermodynamics [5][6]. In traditional irreversible computations, such as in conventional computing systems, information is lost during computation, leading to heat dissipation due to Landauer's principle, as mentioned previously. However, in reversible computations, there is no loss of information, and the process can be fully reversed, resulting in minimal heat dissipation [7].

2. PROPOSED ALGORITHMS

A) Reversible Logic Gates

Reversible Logic Gates' (RLGs) inputs and outputs constitute a one to one mapping, unlike traditional logic circuits that result in the loss of information and energy dissipation in the form of heat. As a result, RLGs offer the potential for zero power dissipation [8]. To optimize RLG designs, various constraints such as fanout limitation, quantum cost minimization, garbage production minimization, and gate-level optimization need to be balanced. Achieving efficient and effective reversible logic circuits with low power dissipation and high performance requires careful consideration of these constraints. CNOT gate, Fredkin gate, SCL gates and Toffoli gate are used to design these cryptography systems.

B) Steps to design

- Step 1: The input image is read and its size is determined using MATLAB.
- Step 2: The image pixel values are transformed to grayscale, which reduces the image to a single intensity channel. Then, the grayscale image is converted to binary, which transforms each pixel value into a 0 or 1.
- Step 3: Random bits are generated for use in watermarking.

- Step 4: The LSB watermarking is performed by modifying the 3rd and 4th LSB positions of the binary image using the generated random bits. This process embeds the watermark information in the image without significantly altering its appearance.
- Step 5: The binary values obtained after LSB watermarking are written to a text file.
- Step 6: The Verilog code takes this text file as input.
- Step 7: The encryption and decryption processes are done using Verilog.
- Step 8: Random keys are generated for encryption and decryption.
- Step 9: The decrypted output is presented in the form of text file.
- Step 10: MATLAB is used to plot both the input and decrypted images, and verify that they are the same.
- Step 11: Finally, the performance of the FPGA is evaluated.

C) LSB watermarking

LSB (Least Significant Bit) is a simple method used for embedding the watermark in the digital images [9] replacing the LSB pixel values to data bits of watermark. Since these bits carry less important information, the image's visual appearance is usually not significantly altered. Watermarking can be achieved by generating 32 random binary bits in MATLAB, which are used to replace the 3rd and 4th LSBs of pixels with an interval of 8 pixels [10]. The resulting bits of the image can be transformed to text file, as demonstrated in Fig.5.1.c.

D)Encryption process

1. The input data (i[0], i[1], ..., i[7]) is split into two groups of 4 bits each: the MSBs and the LSBs.
2. The MSBs are sent as input to an SCL gate, and the first three outputs of this gate are passed to a Toffoli gate.
3. The LSBs are sent as input to another SCL gate, and the first three outputs of this gate are also passed to a Toffoli gate.

4. The last outputs of both SCL gates are given as input to a CNOT gate.
5. The outputs of the two Toffoli gates are given as input to Fredkin gates.
6. The outputs of the Fredkin gates and the CNOT gate are passed as input to XOR gates.
7. The XOR operations are performed using the 4-bit key, which is randomly generated. The output of XOR gates (e[0], e[1], ..., e[7]) are the encrypted data.

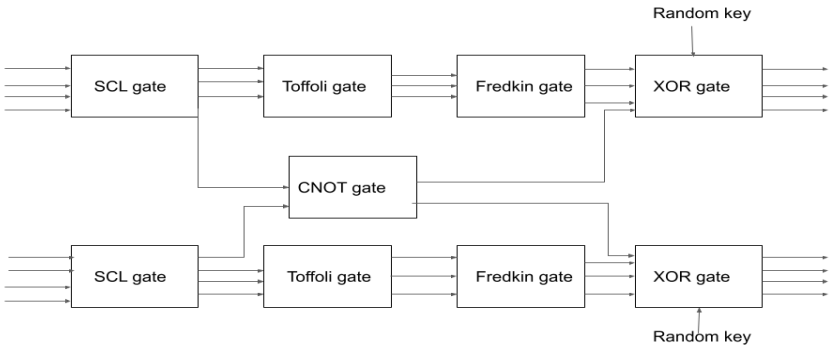

Figure 1. Encryption block diagram [16]

E) Decryption process

1. The encrypted output (e[0], e[1], ..., e[7]) is input into the XOR gate along with the same key used in encryption.
2. The outputs of the XOR gate are sent to Fredkin and CNOT gates.
3. The outputs of the Fredkin gates are used as inputs to Toffoli gates.
4. The Toffoli gates and CNOT gate outputs are fed into SCL gates.
5. The outputs of the SCL gates (d[0], d[1], ..., d[7]) are the decrypted outputs.

The inputs i[0], i[1], ..., i[7] and decrypted outputs d[0], d[1], ..., d[7] are verified that they are the same. If they are the same, then the decryption process was successful, and the original data has been recovered.

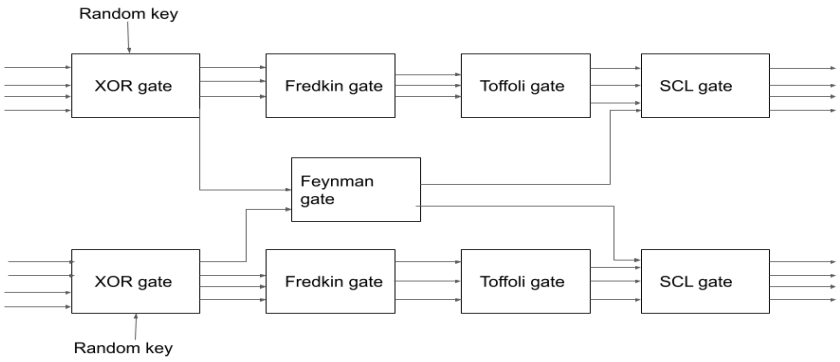

Figure 2. Decryption block diagram [16]

F) Random key generator

The key is a four bit binary value which is randomly generated. The same random key values are used for both encrypting and decrypting. This process is known as symmetric key cryptography [11]. Symmetric key cryptography can be computationally efficient, as it typically requires less processing power compared to asymmetric key cryptography, where the keys engaged in encrypting and decrypting are not the same.

3. EXPERIMENTAL RESULTS

Simulating an RLG-based cryptography system in Xilinx ISE involves designing and implementing the logic gates and circuits involved in encryption and decryption processes, using a Verilog (HDL).

Figure 3. Original input image

Figure 4. Binary image

The first step in this process involves converting the pixel values of the input image in Fig.3 into binary and the resultant binary image is shown in Fig.4 and all the binary values of image are written into text file as shown in Fig.5.1.a. After converting the pixel values to binary, the LSB watermarking can be performed using MATLAB. The watermarked image and its binary output are shown in Fig.5.1b. and Fig.5.1c.

Figure 5.1.a. Binary values of original image **Figure.5.1.b.** LSB Watermarked Image

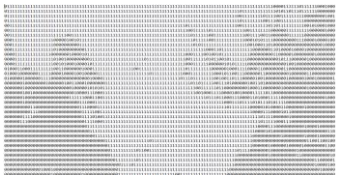

Figure 5.1.c. Watermarked output

To embed the watermark, 32 random bits are generated in MATLAB, which will be used as the watermark data. These bits are then inserted into the 3rd and 4th LSBs of the selected pixels in the image, one bit at a time. The order in which the pixels are modified is with an interval of 8 pixels. Once the watermark has been embedded in the image, the binary values of the watermark is inscribed as a text file as shown in Fig.5.1.c. This file is then used as input to the Reversible Logic Gate Cryptography Design (RLGCD), which is implemented using Verilog. Then the verilog code for encryption and decryption is used. After the decryption process, the output of the decryption algorithm is compared to the original input image to ensure that the decryption was successful.

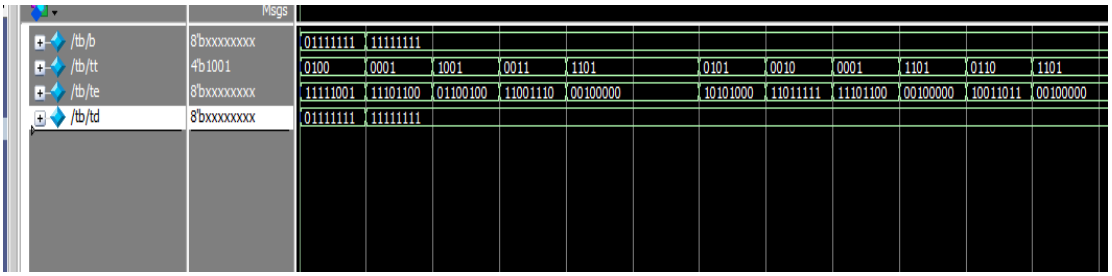

Figure 6. Timing diagram of cryptography process using RLGCD

In Fig.6, 'b' represents the binary values of the watermarked image, 'k' represents the key value used for both encryption and decryption, 'te' represents the encrypted value, and 'td' represents the decrypted value. The encrypted and decrypted binary values are shown in Fig.7 and Fig.8 respectively.

The decrypted values are written into a text file and given as input to MATLAB. By plotting the watermarked image and the decrypted image, the two images are visually compared, and ensured that they are identical.

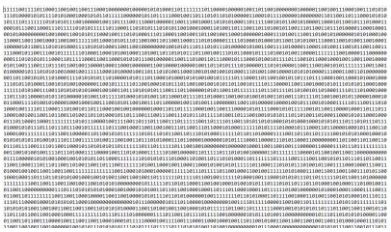

Figure 7. Encrypted output

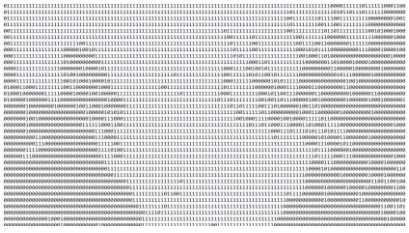

Figure 8. Decrypted output

Fig.9 represents the output of watermarked, encrypted, decrypted, output image of original image.

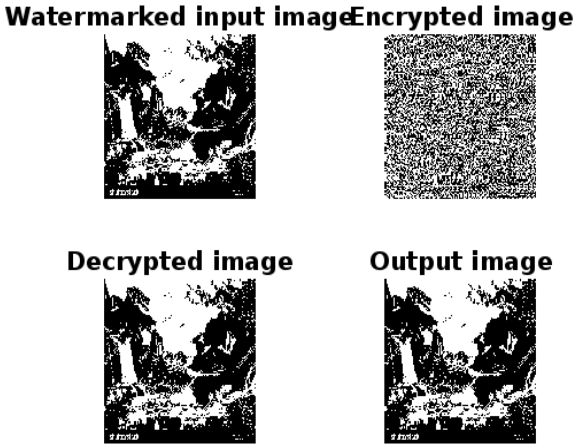

Figure 9. Outputs

Device utilization is an important metric to consider when designing hardware systems, as it directly impacts the cost and feasibility of implementation. The comparison of the prevailing system and RLGCD with random keys for various devices is presented in the table below.

Table 1. Comparison of the existing systems and RLGCD with random keys for various devices

Target FPGA	Circuit	LUT	Flipflop	Slice	Frequency
Virtex 7	Isogenies-MC[12][13]	185,871	218,012	77,425	158.5
Virtex 7	Scalable isogeny [14]	18,820	24,908	4791	202.1
Virtex 7	AES-NP[15]	19,547	53,478	4089	495.32
Spartan 3E	RLGCD [16]	37	40	42	175.047
Spartan 3E	RLGCD using random keys	4	-	2	48.828

Based on the table , it can be observed that RLGCD using random keys for the Spartan 3E device has a significantly better device utilization compared to other existing systems.

Table 2. FPGA result of RLGCD using random keys for Spartan 3E device

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of 4 input LUTs	4	1,920	1%	
Number of occupied Slices	2	960	1%	
Number of Slices containing only related logic	2	2	100%	
Number of Slices containing unrelated logic	0	2	0%	
Total Number of 4 input LUTs	4	1,920	1%	
Number of bonded IOBs	17	66	25%	
Average Fanout of Non-Clock Nets	2.00			

Table 2 represents the device utilization summary of RLGCD using random keys. RTL schematics of RLGCD using random keys are shown in Fig.11 and Fig.12.

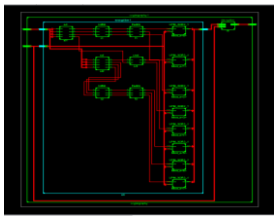

Figure 11. RTL Schematic of Encryption block

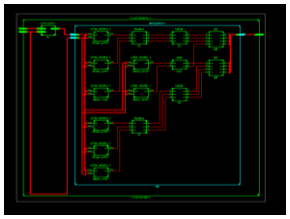

Figure 12. RTL Schematic of Decryption block

4. CONCLUSION

The devised cryptography using random keys aims to provide strong security and efficient power consumption by using random keys and watermarking. Reversible logic gates, such as CNOT, Toffoli, Fredkin, and SCL gates, are used for their inverse operation that can "undo" their effect, which is important for reversible computation. Using MATLAB, LSB watermarking is performed on an image, and the resulting text file is input to an RLGCD implemented in Verilog code using Xilinx software. The Spartan 3E with Reversible Logic Gate Cryptography Design (RLGCD) using random keys shows superior performance than the existing systems. RLG are essential for quantum computation, allowing computation to be run backwards in time, which is important for error correction and other quantum algorithms. Successful implementation and verification of output using Verilog is a positive step towards demonstrating the feasibility of RLGCD using random keys for cryptography.

REFERENCES

[1] Mehran Mozaffari Kermani, Kaj Reza Azarderakhsh, Siavash Bavat Sarmadi, “Fault resilient lightweight cryptography block cipher for secure embedded systems,” in IEEE Embedded System Letters, vol. 6, no. 4, pp.89–92, Dec. 2014.

[2] Shikha Kuchhal , Rakesh Verma, “Security design of DES using reversible logic,” Int. J. Comput. Sci. Netw. Security, vol. 15, no. 9, pp. 81–84, September 2015.

[3] Z. H. A. O. Guosheng,W. A. N. G. Jain, “Security analysis and enhanced design of a dynamic block cipher,” China Commun., vol. 13, pp. 15–160, January 2016.

[4] Gordon E. Moore, “Craming more components onto integrated circuits,”Electronics, pp.114-117, April 1965.

[5] Rolf Landauer, Irreversible and heat generation in the computing process, IBM Research and Development, vol.5, pp.183–191, July 1961.

[6] C.H. Bennett, "Logical reversibility of computation" IBM Research and Development, vol.17, pp.525–532, 1973.

[7] Nagamani A N#1, Jayashree H V#1,H R Bhagyalakshmi ―Novel Low Power Comparator Design Using Reversible Logic Gates‖, Indian Journal of Computer Science and Engineering (IJCSE).

[8] Raghava Garipelly, P.Madhu Kiran, A.Santhosh Kumar, "A Review on Reversible Logic Gates and their Implementations", International Journal of Emerging Technology and Advanced Engineering, vol-3,no-3, March 2013.

[9] Abduullah Bamatraf, Rosziati Ibrahim, Mohd. Najib. B, Mohd. Salleh,"Digital watermarking algorithm using LSB," in 2010 International Conference on Computer Applications and Industrial Electronics, Kuala Lumpur, pp. 155-159, 2010.

[10] Meenal Dadhe, Prof. Anup. R. Nage, "Design of high speed VLSI architecture for LFSR with maximum length feedback polynomial," in International Journal for Scientific Research & Development, vol. 3, no. 5, 2015.

[11] B. Koziel, R. Azarderakhsh, M. Mozaffari Kermani, D. Jao, "Postquatum cryptography on FPGA based on isogenies on elliptical curve," in IEEE Trans.Circuits Syst.I, vol. 64, no. 1, pp. 86–99, Jan. 2017.

[12] Saranya Karunamurthi, Vineyakumar Krishnasamy Natarajan, " VLSI implementation of reversible logic gates cryptography with LFSR key," Microprocessors and Microsystems, Elsevier, vol. 69, pp.68–78, September 2019.

[13] B. Koziel, R. Azarderakhsh, M. Mozaffari Kermani, "A high performance and scalable hardware architecture for isogeny based cryptography," in IEEE Trans.Comput., vol. 67, no. 11, pp. 1594–1609, Nov. 2018.

[14] D.P.Vasudevan, P.K.Lala, J.Di and J.P.Parkerson, ― "Reversible logic design with online testability", IEEE Trans. on Instrumentation and Measurement, vol.55., no.2, pp.406- 414, April 2006.

[15]H. Zodpe, A. Sapkal, “An efficient AES implementation using FPGA with enhanced security features,” in J.King Saud Univ.Eng.Sci., 2018, in press.

[16]Geethu Chandran, Dr. Helen Mary M C, Anjana G, “VLSI Implementation of Image Encryption and Decryption Using Reversible Logic Gates”, in 2020 International Conference on Power Electronics and Renewable Energy Applications,2020.

Author’s Biography

S.Rithiga is currently doing undergraduate in Electronics and Communication Engineering in Nadar Saraswathi College of Engineering and Technology.

Dr.T.Venish Kumar completed PhD in Anna University in the year 2019. He has 10 years of teaching experience and is currently working in the Department of ECE Nadar Saraswathi College of Engineering and Technology. He is doing his research in Analytical Modelling of HEMT devices.

Mr.M.Idhayachandran completed M.E. in the year 2011. He has 14 years of teaching experience and is currently working in the Department of ECE Nadar Saraswathi college of Engineering and Technology.

Mr.S.Prathap completed M.E. in the year 2013. He has 9 years of teaching experience and is currently working in the Department of ECE Nadar Saraswathi college of Engineering and Technology.