Implementation of Modified Dual-Coupled Linear Congruential Generator in Data Encryption Standard Algorithm

N. Akhila[1], Ch. Usha Kumari[2], K. Swathi[3], T. Padma[4], Padmavathi Kora[5]

[1] *M.Tech Scholar, Dept. of ECE, GRIET, Hyderabad, India*

[2,3,4,5]*Dept. of ECE, GRIET, Hyderabad, India*

akhilarao773@gmail.com, ushakumari.c@gmail.com

2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA) | 978-1-6654-3877-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICIRCA51532.2021.9544633

Abstract— **Data transmission in cryptography is held by encipher and decipher processes. Data Encryption Standard is one of the symmetric key encryption algorithms. Data Encryption Standard (DES) is one of the simplest cryptography algorithms. Modified Dual Coupled LCG (MD-CLCG) is an essential element of Pseudorandom Bit Generator (PRBG) because it requires less area and it is more secured compared to previously executed different algorithmic techniques of linear congruential generator (LCG) family and other pseudorandom bit generators. In cryptographic schemes key generation makes an important role. This paper has implemented a modified dual-CLCG for the key generation with the utilization of shift register in Data Encryption Standard cryptographic technique. Usage of Modified Dual-CLCG in Data Encryption Standard algorithm is designed and coded by the Verilog-HDL language and prototyped on FPGA device Spartan3E XC3S500E.**

Keywords— Pseudo Random Bit Generator (PRBG), modified Dual coupled LCG (MD-CLCG), Data Encryption Standard (DES), symmetric key encryption.

I. INTRODUCTION

Securing data in various IOT applications is becoming more difficult by the day, and internet privacy is getting more sensitive. Transferring large amounts of data across the internet to a million or more components could result in privacy concerns [1] [2]. Pseudorandom bit generator (PRBG) is a critical aspect in IOT dependent systems for security.

The Pseudorandom bit generator is a aspect that manages user privacy in IOT devices because internet security in IOT apps and data privacy in IOT premised devices is both difficult to achieve presently. With limitations such as area, starting clock lag, randomness, and power, the huge bit size of PRBG VLSI design is challenging. The PRBG is said to be random if it passes the National Institute of Standards and Technology (NIST) fifteen benchmark statistical tests.

Pseudorandom bits can be generated using a number of different methods. Linear feedback shift register, linear congruential generator (LCG), Blum blum shub generator (BBS), Coupled linear congruential generator (CLCG) and dual-coupled linear congruential generator are different types of linear congruential generators.

Lenore blum, Manuel blum and Michael shub were created Blum blum shub generator (BBS) in 1986 year as pseudorandom generator. The hardware implementation is used to conduct modulo of the largest prime numbers and to calculate the size of a specific prime number [3]-[6]. The input to first output clock intermission is 2n+5 clock cycles, while the output latency is 2n+5 clocks [7] [8].

The simplest technique is to use a linear feedback shift register (LFSR). The LFSR is made up of simple flip flops and an XOR gate. The LFSR's architecture is straightforward. The LFSR takes up less space and has less hardware complexity. Because of its linearity structure, it fails to pass randomization tests. LFSRs are n-bit counters that produce pseudo-random bits. For n-bit, LFSR can only produce 2n-1 sequences [9].

LCG has a smaller footprint and less hardware complexity. It fails to pass randomization tests due to its linearity structure, as anyone may predict the following sequence after some time.

A coupled linear congruential generator is made up of two LCGs connected in parallel and a comparator [10] [11]. CLCG is safer than a single LCG [11]. CLCG fails the NIST five tests as well as the Discrete Fourier Transform test. The CLCG DFT test, which comprises two inequality comparisons, shows that sequences have a periodic shape [12].

Two connected LCG layouts with two comparators merge with one controller unit and memory to create a dual-coupled LCG (flip flops). Dual-CLCG generates pseudo random bits with clock latency of 2n+5 clocks at first, then only 2 clocks later to generate the data. For n-bit pattern formation, Dual-CLCG requires (2n-1) flip flops. Dual-CLCG fails to reach the NIST statistical randomness requirement and to achieve maximum length of sequence.

Cryptography is an essential component in used in the security of computers. Encipherment algorithms include substitution and transposition the main requirement is no data loss. When the transmitter and recipient share the same private pass code (key), the design is considered as a secret key, single key, or symmetric key/encryption. The mechanism is considered to be two key encryption or asymmetric encipher if the sender and receiver have distinct private keys. A symmetric encipher process is made up of five aspects: plain text, encryption algorithm, secret key, cypher text, and decryption of the data.

An invention of the DES algorithm was a valid interpretation in the early stages of surveillance systems. This was the most intensively researched, validated, and utilised cryptoalgorithm, and it will remain that way for a long time.

II. RELATED WORK

In this section the design of modified dual coupled linear congruential generator and cryptography process of Data Encryption Standard techniques are going to be discussed.

A. Modified Dual-Coupled Linear Congruential generator

The PRBG approach is intended to produce more efficient pseudo random bit patterns than prior approaches. Dual-coupled LCG is used to develop the PRBG architecture. By swapping out one controller unit and one memory unit, the XOR gate is replaced. In comparison to previous methods, the PRBG approach using modified dual coupled LCG has a simple architecture and secured [13], represents in fig 1.

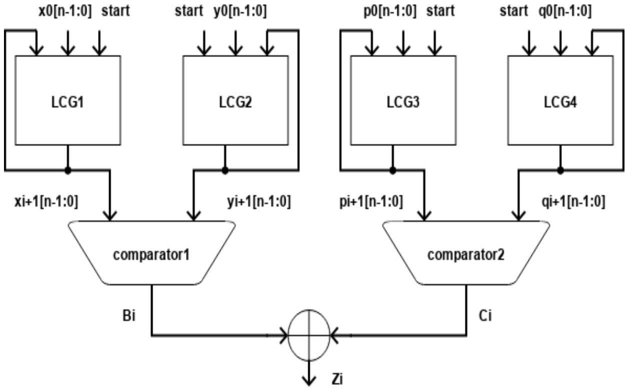


Fig 1: Modified Dual-CLCG architecture

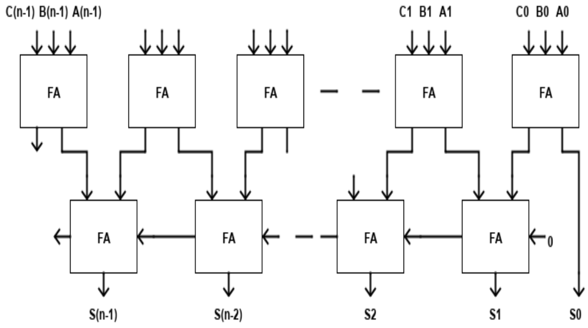
Pseudorandom numbers or bits are generated using the linear congruential generator. LCG is a key functional block in architecture of modified dual CLCG. Three operand modulo 2n Adder, n bit 2 by 1 multiplexer, and n-bit register are used to create a linear congruential generator. Architecture for Linear Congruential Generator is displayed in fig 2.

Fig 2: Architecture of the linear congruential generator

Full adders are used to create 3-operand modulo 2^n carry save adder. This carry save adder will carry out a standard addition operation. The addition process is performed in two phases by the Modulo Adder. In the first phase, a serial version of full adders is created, which performs addition operations on each bit that measures summation and carry out bits. The ripple carry adder is the next step in modulo adder and is used to calculate the end summation value. 3-operand modulo 2 power n carry save adder (CSA) block diagram displayed in fig 3.

Fig 3: Architecture of 3-Operand Modulo carry save Adder

To compare the two outputs of LCGs, magnitude comparators are utilised. The magnitude comparator is developed by utilising logic gates in this architecture. The magnitude comparator's output will satisfy two conditions: larger than and less than or equal to. A n-bit and two bit magnitude comparator architectures are shown in fig 4.



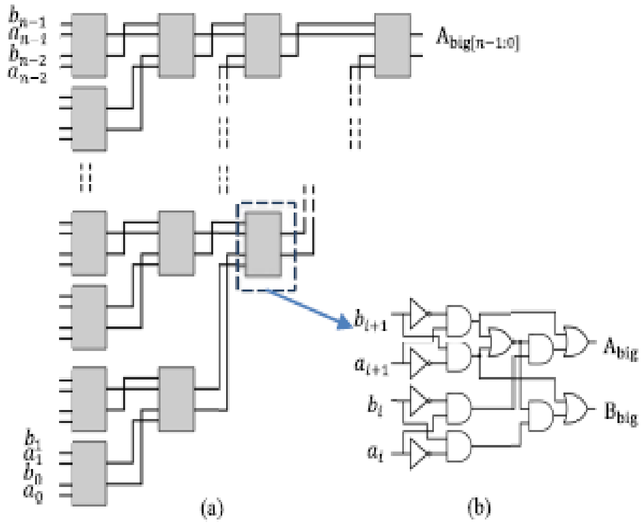


Fig 4: Architecture of (a) n-bit comparator (b) 2-bit Comparator

The comparison results for this PRBG with other approaches are mentioned in the Table I.

B. Data Encryption Standard

Cryptography is basically defined as the converting data or information into the indistinct form for the confidentiality.

TABLE I. AREA AND TIME COMPARISON OF DIFFERENT PRBGS

PRBG methods	Input to first output (initial clock latency)	Critical path	Output latency	Area
BBS [8]	2n+5 clock	$2(T_{2OA}+T_{MUX})$	2n+5 clock	$2A_{3OA}+A_{2OA}+A_{MUX4}+3A_{MUX2}+4A_{REG}+A_{FSM}$
CLCG	1 clock	$T_{3OA}+T_{MUX}$	1 clock	$2A_{LCG}+A_{COMP}=2(A_{3OA}+A_{MUX}+A_{REG})+A_{COMP}$
Dual-Coupled LCG	2^n clock	$T_{3OA}+T_{MUX}$	2 clock	$4(A_{3OA}+A_{MUX}+A_{REG})+2A_{COMP}+A_{TRI}+A_{MEM}+A_{FSM}$
MD-CLCG	1 clock	$T_{3OA}+T_{MUX}$	1 clock	$4(A_{3OA}+A_{MUX}+A_{REG})+2A_{COMP}+A_{X}$

Cryptography is a process that coverts an understandable information into cipher text by using a personal key. Cryptography contains two processes encipherment (encryption) and decipherment (decryption). Encipherment is used to modify the original theory into cipher text by using private key. Decipherment is used to modify the cipher text into original text by using same key which is used in encryption process. And private key is available at authorized people only. But a brute force exhaustion attack can be identified.

There are two ways of ciphers, block cipher and stream cipher. A block cipher allows a block of elements as one input at a time and for each input block it produces output in the block form. A stream cipher works with input elements simultaneously and gives output one element at a time.

A few federal companies provide reliability to electronic informative devices. Data Encryption Standard (DES) [14]-[16] is one of the cryptography algorithms which may be utilized by united companies to secure delicate information. Securing the information while in storage or in communication is mandatory to take care of confidentiality and coherence of theory mentioned by the data. The algorithm is basically contains mathematical procedures to modify the original information into cipher text and decipherment process. Block diagram for data encryption standard is shown in fig 5.

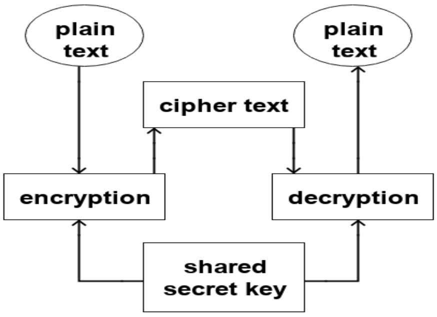


Fig 5: Data encryption standard process

In this project symmetric encryption is used. The flow chart figure for Data Encryption Standard is mentioned in the fig 6. Encryption process requires two inputs; one is plain (understandable) text and private key. In the DES encryption process, plain text is of 64 bit in length and private key is of 48 bit length.

As by observing the flow chart of DES encryption process it contains three stages. Initial permutation (IP) takes input as 64 bit plain text and gives output as permuted input. The output of initial permutation is enters into stage 2 which contains 16 rounds of the some function and it involves in both permutation and substitution functioning. Initial permutation output is divided into two halves and applied to functions along with the key. At the last (sixteenth) round of functioning the left and right halves are interchanged with key to provide pre output. In the stage 3, the pre output applied to inverse initial permutation (IP^{-1}) for 64 bit cipher text extraction by using inverse initial permutation function. The left and right portions of a 64-bit input node are processed separately as 32-bit quantities, marked L (left) and R (right).

The round key Ki has a 48-bit value. R is a 32-bit input. The steps involved in function 'f' are represented in fig 7. Initially R input is enlarged to 48-bits with the help of a predefined table that describes a permutation, followed by an expansion that duplicates 16 R bits. The substitution is made up of 8 S boxes; each box it takes 6-bits as outer code four bits. In the predefined table, these transformations are provided. The initial and final bits of the entry to Si box yield a 2-bit binary number that can be used to select one of four Si replacements given by the four rows of the table. One of the sixteen columns is chosen by the middle four bits. The four bits in the centre determine which of the 16 divisions will be chosen. Converting the specific point in the unit chosen by the column and row to its four-bit value yields the result.

DES decryption is also involves in three stages as in encryption process. The decryption of encrypted data follows the same mechanism as the encrypting data, namely, initial permutation, followed by 16 rounds of sophisticated key dependent computations, and finally, the final permutation, which yields the decrypted data, but with a little difference. That is for encryption, the keys provided to round 1 until round 16 are from key 1 to key 16, but for decryption, the keys are delivered in the reverse order, i.e., key16 is given to round 1, key15 is issued to round 2, and so on until round 16.

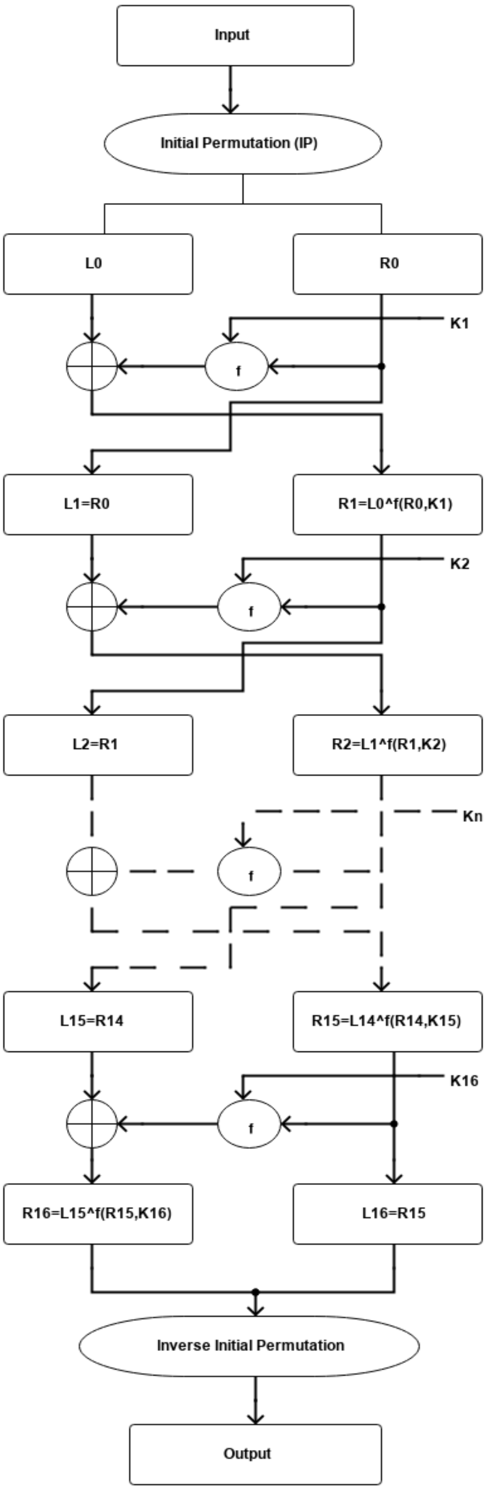


Fig 6: Flowchart for data encryption standard (DES) encryption process

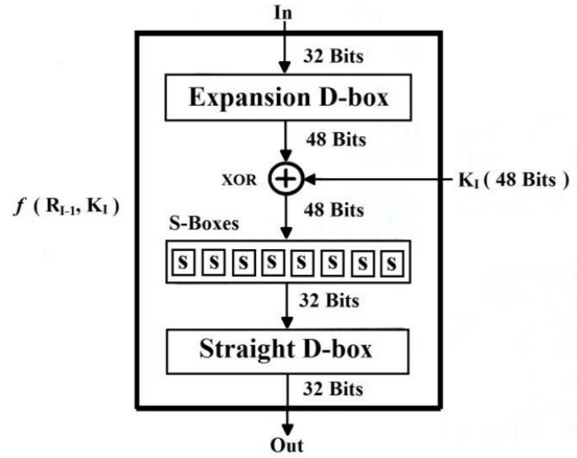


Fig 7: function (f) representation of DES

III. PROPOSED WORK

Pseudo random bit generator is also known as a counter. The output of modified dual-CLCG is applied to the serial in parallel out shift register a 1 bit data outcome of MD-CLCG is converted into 64-bit length because for DES algorithm the key must be of initially 64 bit length. The outcome (Z_i) of modified dual clcg is used for the key generation in data encryption standard process. The data output of serial in parallel out shift register is taken as key (K). Block diagram for the proposed work is represented in fig 8.

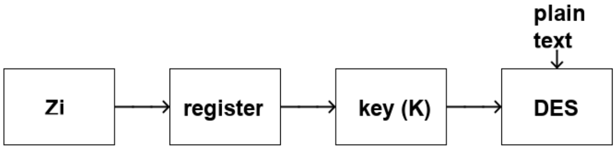


Fig 8: block diagram representation of Proposed work

The process of Data encryption standard are discussed in the previous section. A 64 bit length of plaintext is taken as input and applies a converted 16 subkeys of each having length 48 bits to input to generate the cipher text. The decryption procedure converts the cypher text back into ordinary text.

Flow chart diagram for key generation Data Encryption Standard is shown in fig 9. In a symmetric key encryption process the key generated by using the permuted choice tables 1 and 2 (PC-1 & PC-2). Initially, a 64-bit key is taken as input from the serial in parallel out shift register for key creation, with 56 bits produced being used by the algorithm and 8 bits not being used by the method, which is referred to as detection bits of error. The 8 fault identifying bits are given to an 8-byte odd integrity check.

In some cases, key is created in an encoded manner. The cipher created by encrypting a pass code with a password encrypting key is specified as a random 64 bit number. The frequent patterns of the encoded key cannot be altered until the key is decoded in this situation. The preservation of the data is determined by the level of protection offered for the key used to encode and decode information. In key generation process 16 sub keys are generated and each sub key contains 48 bit length. After that, the 56-bit number is broken up into 2-28 bit values, C_0 and D_0. C_{i-1} and D_{i-1} must be applied to the rotational, or circular left shift, of specified bits at each round, according to the iteration table [14]. The coming cycle utilizes these changed values as input. They even provide input to the table's permuted choice-2, it yields a 48 bit outcome that can be used as an intake to function f (R_{i-1}, K_i) in DES process.

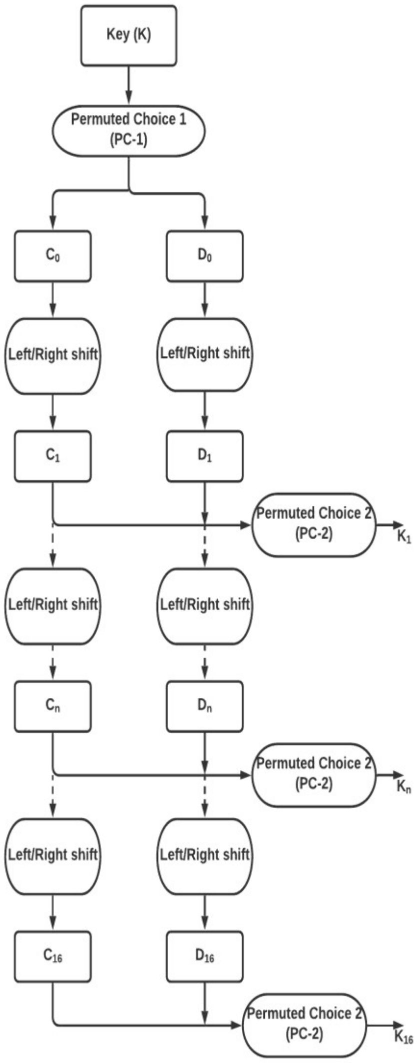


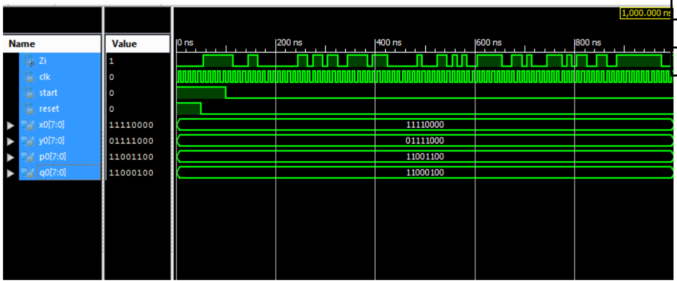
Fig 9: Flow chart for KEY generation

After key generation process, the derived subkeys (k1, k2, k3,....,k16) are applied to data encryption standard along with original (plain) text. The process of DES is mentioned in previous section.

IV. RESULT ANALYSIS

The implementation of modified dual-CLCG in Data encryption standard by using the Xilinx ISE 14.7 tool and coded by using Verilog-HDL language. Logic simulation is done by the ISIM simulator for this design.

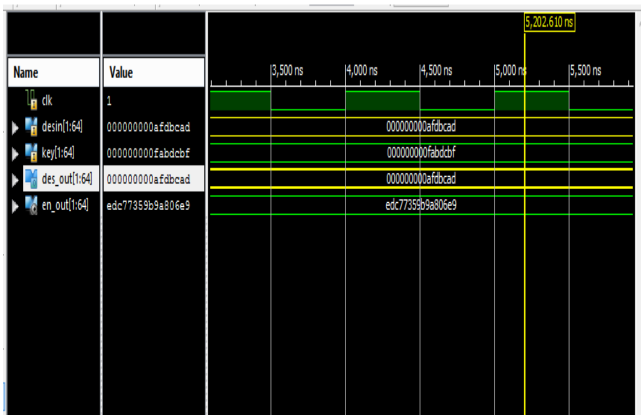
Modified dual-CLCG generator design uses a start button which acts like enable or selection line to the multiplexer. If start is '0' then design takes seed as input after one clock pulse start becomes '1' then design accepts feedback input. Zi is the final outcome of the modified dual-CLCG and simulation result window for this design is represented in fig 10.



Name	Value
Zi	1
clk	0
start	0
reset	0
x0[7:0]	11110000
y0[7:0]	01111000
p0[7:0]	11001100
q0[7:0]	11000100

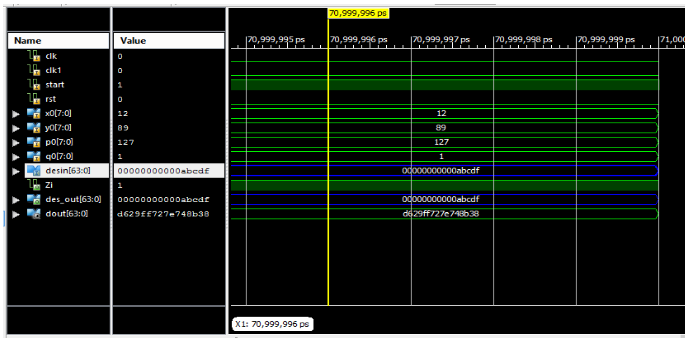
Fig 10: simulation result for modified dual-CLCG

Data encryption standard has two inputs those are plain text and key of having 64-bit length. Encryption process output is the unreadable text (cipher text). The simulation result of data encryption standard is represented in fig 11.The outcome of modified dual-CLCG is applied to the 64-bit serial in parallel out shift register and this result is used for key generation. The simulation result for data encryption standard using modified dual-CLCG is represented in fig 12.



Name	Value
clk	1
desin[1:64]	000000000afdbcad
key[1:64]	000000000fabdcbf
des_out[1:64]	000000000afdbcad
en_out[1:64]	edc77359b9a806e9

Fig 11: simulation result for data encryption standard



Name	Value
clk	0
clk1	0
start	1
rst	0
x0[7:0]	12
y0[7:0]	89
p0[7:0]	127
q0[7:0]	1
desin[63:0]	00000000000abcdf
Zi	1
des_out[63:0]	00000000000abcdf
dout[63:0]	d629ff727e748b38

Fig 12: simulation result for modified dual-CLCG in DES

After completion of simulating the design it is synthesized by using isim simulator tool. The synthesis results for implementation of modified dual-CLCG in DES are mentioned in Table II.

TABLE II. DESIGN UTILIZATION

Elements	Design utilization
LUTs	6453
Max. Frequency	17.432MHz
Memory	394712
Delay	9.494ns

V. CONCLUSION

The modified dual-CLCG is designed by using linear congruential generator with less area and more secured. Cryptography is a process of both encipherment and decipherment. Data Encyrption Standard is a cryptography algorithm and it is also a symmetric encryption algorithm. The sender and recipient had the same key in a symmetric key encryption technique. The key is generated by the modified dual-CLCG component. Key generation includes predefined tables such as permuted choice 1 and permuted choice 2. The specifications of modified dual-CLCG were calculated by the Xilinx ISE 14.7 tool using Verilog-HDL prototype and simulated through Spartan 3E XC3S500E. The key generated from modified dual-CLCG can also be utilized for other advanced cryptography schemes like advanced encryption standard, RSA surveillance, and so on. These cryptography techniques are used at usage of mobiles, pay TV, and storing files using secured freeware.

ACKNOWLEDGMENT

I'm grateful to my teachers and review committee for their ideas, which helped me better the manuscript. I'd like to express my gratitude to my family for their encouragement and support throughout the study process. The work of the first author is based on work from modern cryptographic technologies and data transmission issues.

REFERENCES

[1] Q. Zhang, L. T. Yang and Z. Chen, "Privacy Preserving Deep Computation Model on Cloud for Big Data Feature Learning," in IEEE Transactions on Computers, vol. 65, no. 5, pp. 1351-1362, 1 May 2016.

[2] J. Zhou, Z. Cao, X. Dong and A. V. Vasilakos, "Security and Privacy for Cloud-Based IoT: Challenges," in IEEE Communications Magazine, vol. 55, no. 1, pp. 26-33, January 2017.

[3] T. W. Cusick, "Properties of the x/sup 2/ mod N pseudorandom number generator," in IEEE Transactions on Information Theory, vol. 41, no. 4, pp. 1155-1159, July 1995.

[4] Sidorenko A., Schoenmakers B. (2005) Concrete Security of the Blum-Blum-Shub Pseudorandom Generator. In: Smart N.P. (eds)

Cryptography and Coding. Cryptography and Coding 2005. Lecture Notes in Computer Science, vol 3796. Springer, Berlin, Heidelberg.

[5] Blum, L. et al. "A Simple Unpredictable Pseudo-Random Number Generator." SIAM J. Comput. 15 (1986): 364-383.

[6] C. Ding, "Blum-Blum-Shub generator," IEEE Electron. Lett., vol.33, no.8, p.667, Apr. 1997.

[7] P. Peris-Lopez, E. San Millán, J. C. A. van der Lubbe and L. A. Entrena, "Cryptographically secure pseudo-random bit generator for RFID tags," 2010 International Conference for Internet Technology and Secured Transactions, 2010, pp. 1-6.

[8] A. K. Panda and K. C. Ray, "FPGA Prototype of Low Latency BBS PRNG," 2015 IEEE International Symposium on Nanoelectronic and Information Systems, 2015, pp. 118-123.

[9] D. Xiang, M. Chen and H. Fujiwara, "Using Weighted Scan Enable Signals to Improve Test Effectiveness of Scan-Based BIST," in IEEE Transactions on Computers, vol. 56, no. 12, pp. 1619-1628, Dec. 2007..

[10] R. S. Katti and S. K. Srinivasan, "Efficient hardware implementation of a new pseudo-random bit sequence generator," 2009 IEEE International Symposium on Circuits and Systems, 2009, pp. 1393-1396.

[11] R. S. Katti and R. G. Kavasseri, "Secure pseudo-random bit sequence generation using coupled linear congruential generators," 2008 IEEE International Symposium on Circuits and Systems, 2008, pp. 2929-2932.

[12] R. S. Katti, R. G. Kavasseri and V. Sai, "Pseudorandom Bit Generation Using Coupled Congruential Generators," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 57, no. 3, pp. 203-207, March 2010.

[13] A. K. Panda and K. C. Ray, "Modified Dual-CLCG Method and its VLSI Architecture for Pseudorandom Bit Generation," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 66, no. 3, pp. 989-1002, March 2019.

[14] Wayne G. Barker. 1991. "Introduction to the Analysis of the Data Encryption Standard (DES)," Aegean Park Press, USA.

[15] Wikipedia contributors. "Data Encryption Standard." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 13 Jul. 2021. Web. 14 Jul. 2021.

[16] Stallings, William. "*Cryptography and Network Security: Principles and Practice*". 4th ed. Upper Saddle River, N.J.: Pearson/Prentice Hall, 2006.