A Novel Data Encryption Standard (DES) by using Pseudo Random Binary Sequence Key in Matlab

Som Banerjee
Dept. of Computer Science and Engineering
Modern Institute of Engineering and Technology, Hoogly, India
sombanerjeecse@gmail.com

Chandrama Dey
Dept. of Electronics and Communication Engineering
Pailan College of Management and Technology, Kolkata, India
chandramadey1992@gmail.com

Abstract — Data Encryption Standard (DES) is one of the most important symmetric key cryptographic algorithm that have gained importance globally because of its use in different areas of cryptography. DES consists of initial permutation, 16 round (key generation, s-box and p-box), final permutation. Generating a highly random key is a very difficult task which actually determines the quality of security offered by the cipher text generated by the algorithm. Pseudo random binary sequence can be an alternative way to solve this problem. Pseudo random binary sequences create random bit generation, which helps in increasing the security of the key generated. In this manuscript, a 64 bit pseudo random key generated by using a pseudo random binary sequence generator is applied over a DES for performing encryption and decryption. The key generated by pseudo random binary sequence has proved much higher level of security (randomness p-value of 0.607 tested by NIST frequency mono bit test) and time complexity (of 0.054) than the existing key.

Keywords — Cryptography, Data Encryption Standard, Decryption, Encryption, NIST, P-value, Pseudo Random Binary Sequence, Random Number, Symmetric Key Cryptography.

I. INTRODUCTION

While sending information (message, voice, pictures) through insecure channel medium, there is a great possibility of data disclosure and data manipulation if that is not secured fully. Therefore, efficient mechanisms are needed to protect the data when it gets transmitted through any insecure channel or via any medium. One way that we can protect such information is to convert the readable message into unreadable form using a key before transmission through a medium and this process is called encryption. On the other side, the encrypted unreadable message is retrieved back to the original readable form by the inverse process of encryption called decryption. Cryptographic techniques are mainly divided in two main parts. When one single key is used in both side for ciphering and deciphering then it is known as symmetric cryptography. When two different keys are used in both side for ciphering and deciphering then it is known as asymmetric cryptography [4][15].

Pseudo random number generator (PRNG) is very much important in the implementation of different security techniques and is often considered to be fundamental procedure for cryptography. The potential of a cryptographic algorithm to justify its security is dependent on the key unpredictability which can be achieved by using random number generator [16]. It is known that true or ideal random number generator (TRNG) is just an approximation. In fact a real TRNG must be able to produce sequence of independent bits, which, if again restarted, does not produce same repeated sequence. Practically pseudo random number generator (PRNG) can be classified into two major categories [18], namely pseudo random number generator (Pseudo-RNG) and physical random number generator (physical-RNG). Pseudo-RNG is an algorithm that expands a small short seeds into a required long bit sequences. Physical-RNG is based on noise which converts a noise into required sequences.

This manuscript introduces a pseudo random binary number generated key based Data Encryption Standard (DES) implementation and is compared with the existing Data Encryption Standard (DES) implementation. It is noted from the simulation result that the PRBS key based DES proves to be more secure (randomness p-value of 0.607 tested by NIST frequency mono bit test) and has a better time complexity (of 0.054) than the existing DES.

II. DESCRIPTION OF DES

A. Architecture of DES algorithm

Data Encryption Standard (DES) algorithm works with a compact array of 64 bit messages. The first step is known as initial permutation. As the initial permutation finishes, the 64 bit messages is broken down into two parts each of which is of 32 bit long consisting of left part and right part as shown in Figure 1. Then, an operation starts which is also known as a function *f* of DES. Inside this function, the message gets mixed with the key and this function continues for 16 times which is known as rounds of DES [1]. All rounds consist of same operations. At the end of 16 rounds function, the right and left parts are mixed and then the final permutation completes the algorithm and it is known as reverse initial permutation [15].

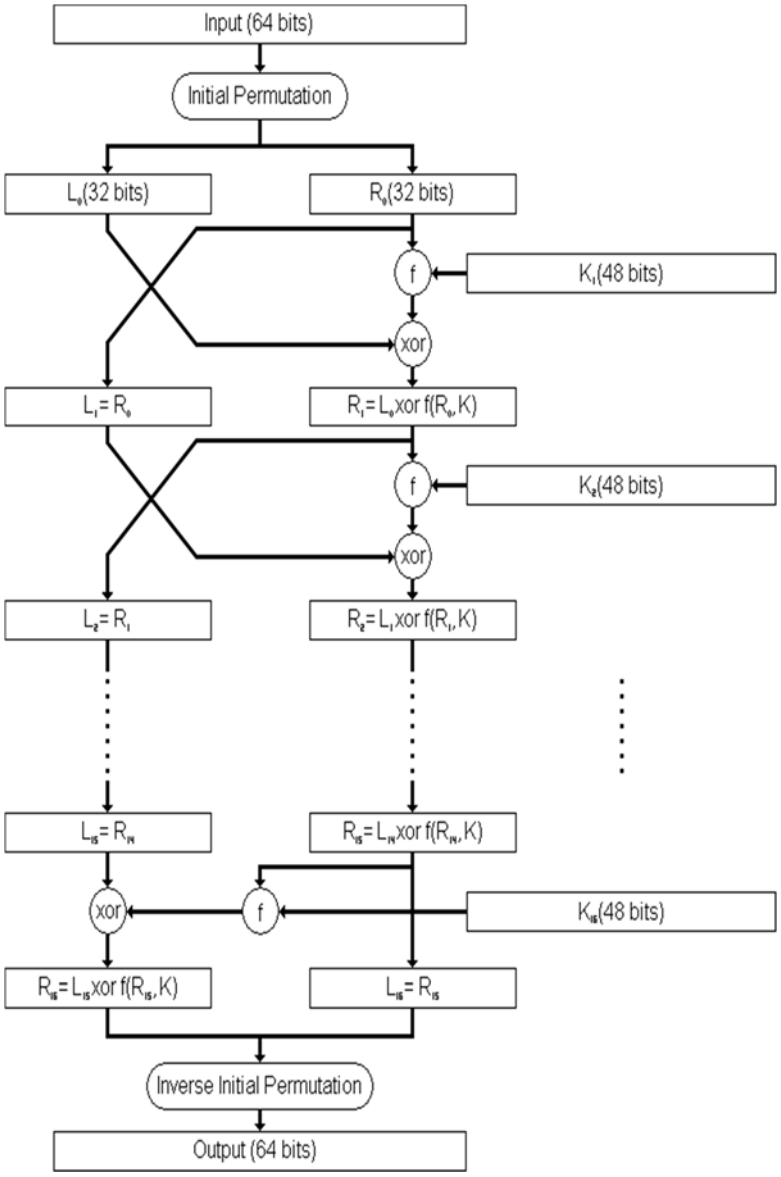


Figure 1. Complete architecture of DES algorithm

B. Round function of DES algorithm

In each of the separate round as shown in Figure 2, the key bits get shifted, and out of this, 48 bit gets selected from the 56 bit key. The right part consisting of 32 bit gets expanded to 48 bit and is known as expand permutation. Then the 48 bit right part gets mixed with a 48 bit shifted and permuted key by xor operation. Then the 48 bit xored block is sent to eight parallel substitution boxes also called s-boxes and 32 bit comes out of the substitution boxes and is then again permuted. There are 4 parts inside a function *f* namely expand permutation, key shift and permutation, substitution box, permutation box. The output produced from function *f* gets mixed with the left part by another xor operation. The result produced from these operations forms the new right half and the old right half becomes the new left half for the next round. These procedures are repeated for consecutive 16 times, to make 16 DES rounds [2] [3] [5].

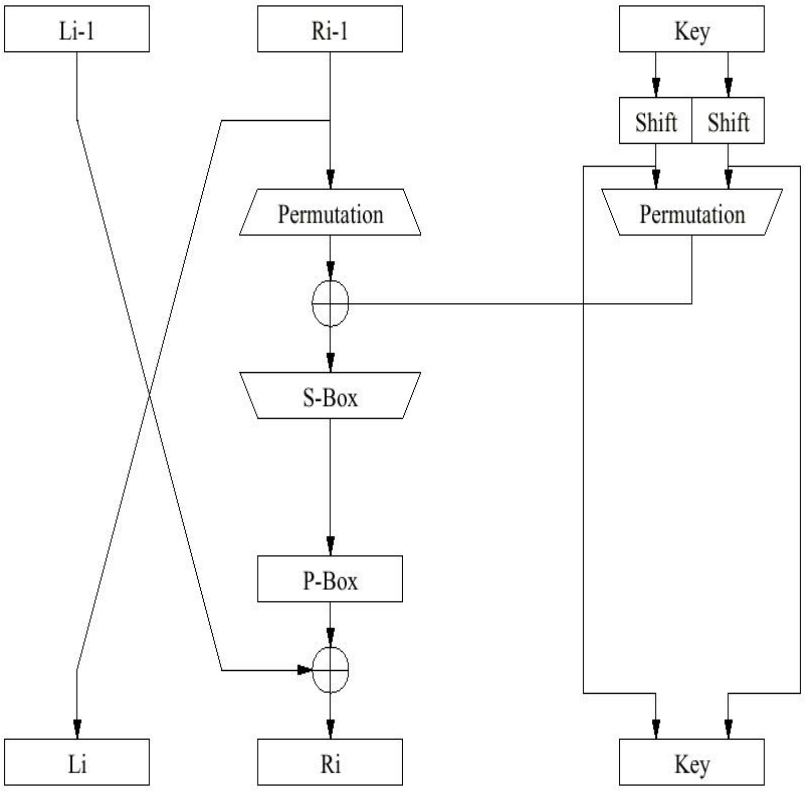


Figure 2. Round function of DES algorithm

III. PSEUDO RANDOM BINARY SEQUENCE (PRBS)

A pseudo random binary sequence is widely used in different areas such as computing simulation, data sampling, statistics, game theory, gambling and cryptographic techniques [12]. A pseudo random binary sequence must be cryptographically consistent and infeasible to any type of attacks. Random numbers which are used in cryptography process must be considered as a fundamental part of a cryptographic function [8] [9] [10] [11] [15]. A defect or fault in the creation of random number can bring about a total failure of the process. So, there is a need that the security of the generated random bits to be used for cryptography application must be checked by using statistical NIST test suite. A diagram to show pseudo random binary sequence generator is given in Figure 3.

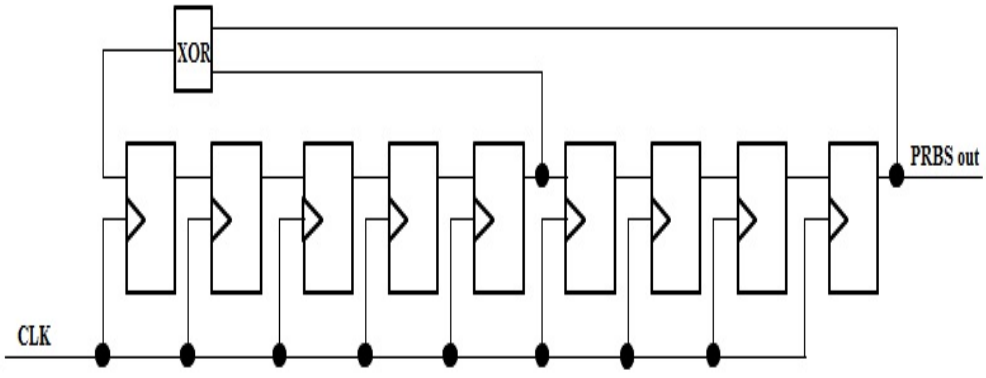


Figure 3. Pseudo random binary sequence generator circuit

IV. PROPOSED WORK

A. Proposed architecture of DES algorithm

The complete operation of DES algorithm and in which part the pseudo random binary sequence generated key is applied is shown in Figure 4. The pseudo random binary sequence which is utilized as a key is generated randomly. By using the existing DES techniques and pseudo random binary sequence generated key, which is more secured and better time complexity is estimated in this manuscript.

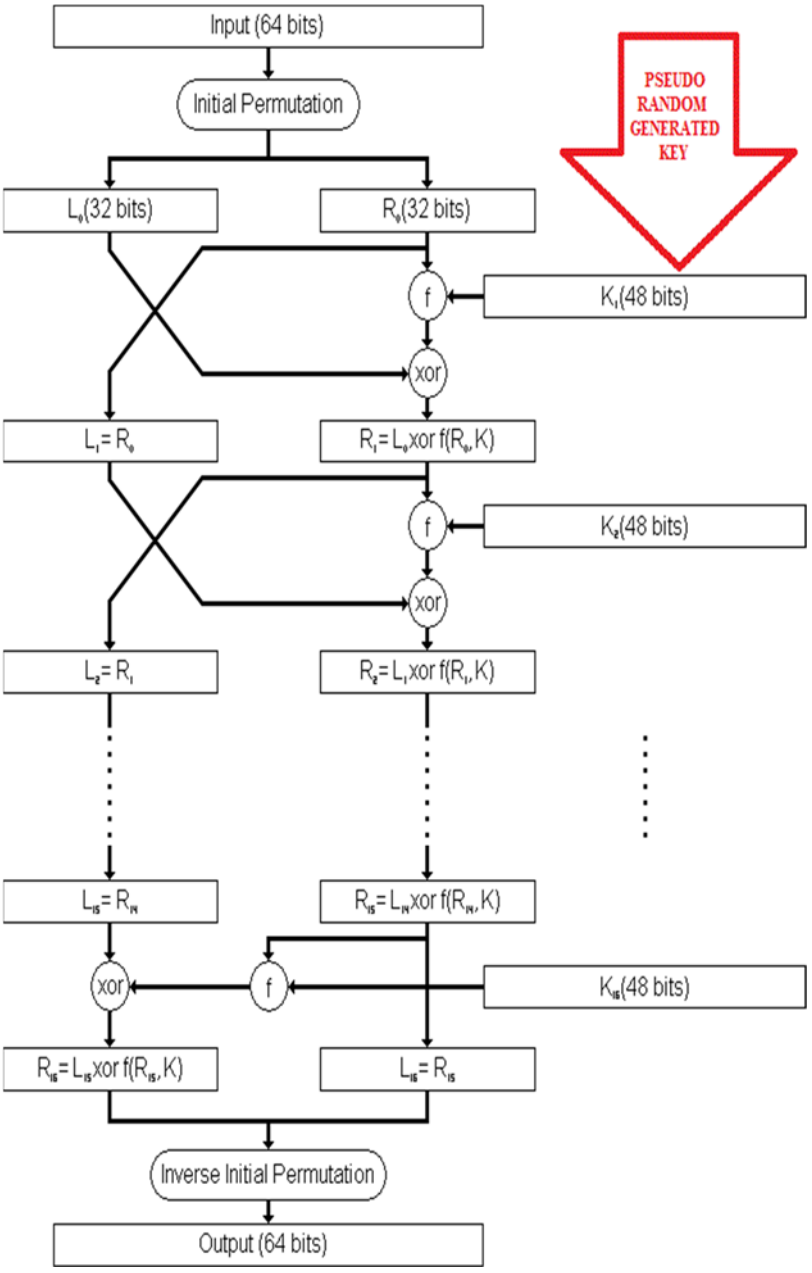


Figure 4. Proposed architecture of DES algorithm

B. Proposed block diagram of implementation

The Data Encryption Standard (DES) is implemented using pseudo random binary sequence generated key totally in Matlab interface. The pseudo random binary sequence is generated by pseudo random sequence generator. The proposed work architecture is to implement a data encryption standard (DES) algorithm by using key generated by pseudo random binary sequence [13]. The pseudo random binary sequence generated key is fed to the algorithm and encryption and decryption of data is done as shown in Figure 5. At every simulation, the key generated is completely a new random number thereby producing different cipher text at each simulation. Implementation of Data Encryption Standard encryption/decryption using the above sequence is done to improve the security and time complexity than the existing techniques.

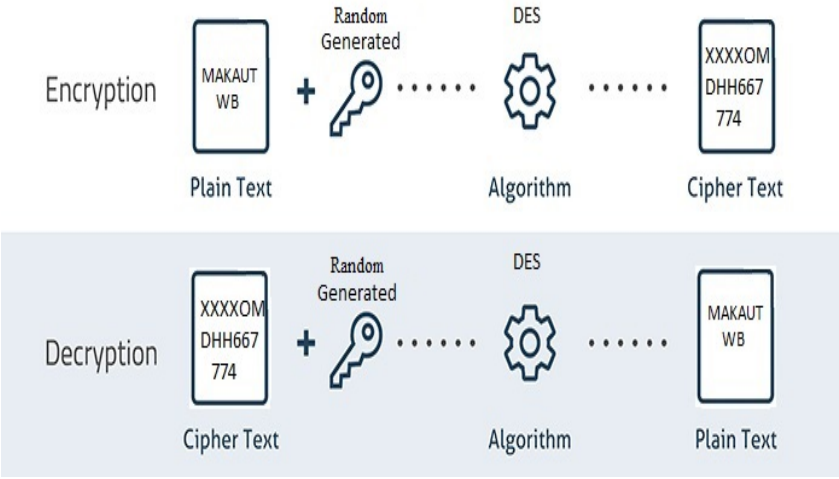


Figure 5. Proposed block diagram of implementation

V. SIMULATION RESULTS

A. Generating pseudo random binary sequence

The pseudo random binary sequence is generated by using a random sequence generator. This 64 bit pseudo random binary sequence is used as a key to implement DES algorithm and encryption/decryption is done. The Matlab graphical simulation of the generated pseudo random binary sequence is given in Figure 6.

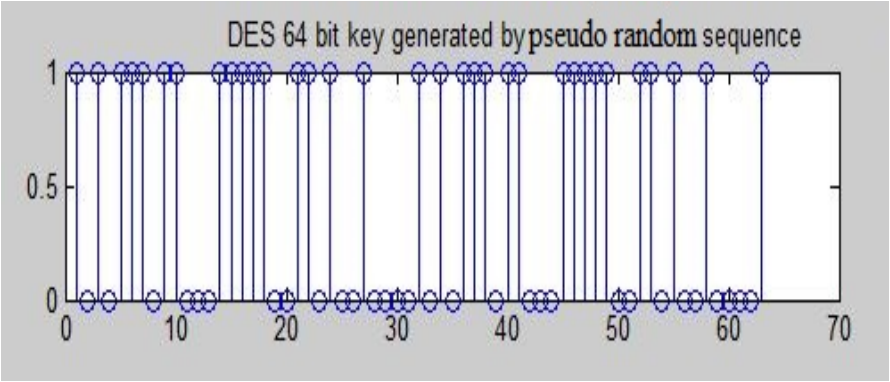
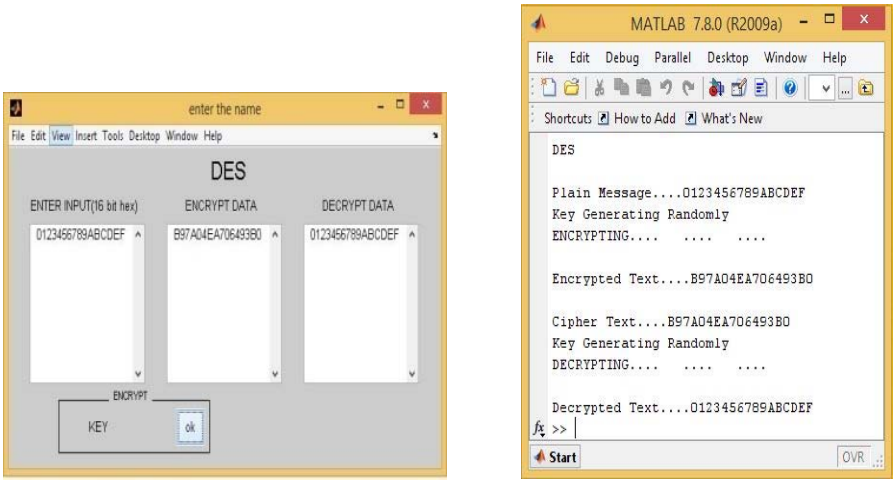


Figure 6. Pseudo random binary sequence generation

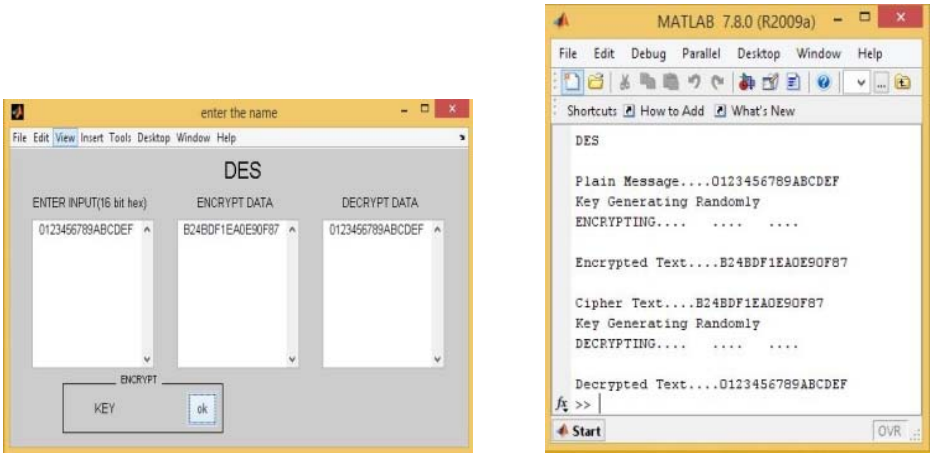
B. DES algorithm key using pseudo random binary sequence

The generated random 64 bit pseudo random binary sequence is applied as key to implement Data Encryption Standard encryption/decryption. The generated key is fed inside the DES cryptographic algorithm and DES step is performed over the messages. As DES is a symmetric key cryptographic algorithm the same key is used for both encryption and decryption. Figure 7 and Figure 8 shows implementation of message encryption over randomly generated key over the same messages at two different instances. At different instance of simulation and with different time interval, different binary bit sequence gets generated thereby changing the DES key at each time and also the cipher text changes periodically. As a result the randomness of the DES key is maintained, which affect the security of the cryptography to a high extent making this a better random cryptography. The same message is encrypted at different instance of time and two different cipher text gets generated showing the randomness of the key. Both the back end and the front end interface if given in the Figure 7 and Figure 8.



(a) (b)

Figure 7. DES encryption and decryption by pseudo random binary sequence generated key for first observation (a) GUI (b) command window



(a) (b)

Figure 8. DES encryption and decryption by pseudo random binary sequence generated key for second observation (a) GUI (b) command window

C. Pseudo random binary sequence randomness testing (Frequency mono bit test)

To understand the random property of a binary sequence there are NIST test suite. The frequency mono bit test is a random property tests of statistical test suite being developed by the National Institute of Standards and Technology (NIST) [17] are performed over the pseudo random binary sequences generated in this manuscript. The frequency mono bit test follows the decision rule at (1% level) which says that, if p-value ≥ 0.01 the sequence is purely random otherwise the sequence is not random [14].

The whole implementation is done by using Matlab tool. 200 observations are taken and in all observation, the 64 bit pseudo random binary sequences are checked by frequency mono bit test. The NIST randomness analysis for 200 observations values are furnished in TABLE I. It is observed that the average p-value of frequency mono bit test after 200 observation comes to be exactly 0.607, which is above the existing p-value 0.382 [7] as shown in TABLE I**.** It proves that the generated pseudo random binary sequence based key is more random in nature**.** According to the observation it can be said that the proposed p-value came is better than any other experimental work related to pseudo random binary sequence based cryptography.

TABLE I. ANALYSIS OF P-VALUES

RANDOMNESS TEST	P- VALUE [7]	P-VALUE [PROPOSED]
Frequency Mono Bit Test	0.382	0.607

Now all the 200 observed p-value found by NIST frequency mono bit test is plotted in a graph using Matlab. The variation of the random bit generation is observed and it is concluded from the graph that most of the p-values comes above the existing p-value though some point came under the existing p-value. The graph shown in Figure 9 denotes the average p-value after 200 observations and the existing standard p-value for NIST frequency mono bit test for randomness analysis. The violet line denotes the proposed average p-value (0.607) and the below yellow line denoted as existing p-value (0.382) for NIST STS randomness analysis [6].

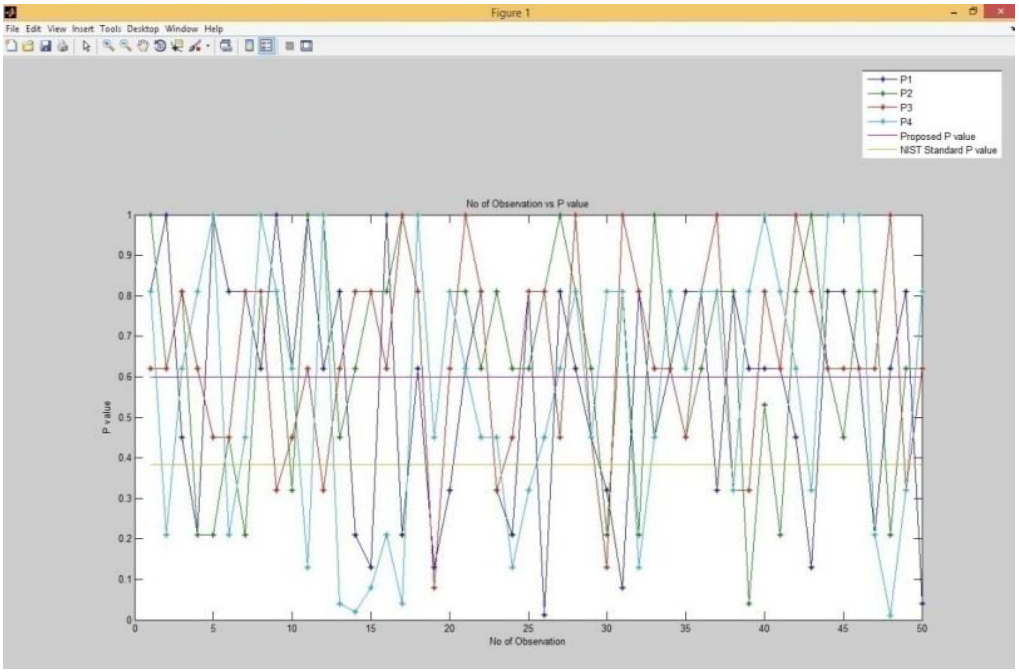


Figure 9. Graph representing Number of observation vs p-value

1) Analysis of time complexity: From the Matlab simulation, it is noticed that the execution time during DES encryption/decryption using proposed pseudo random binary sequence generated key is 0.054 which is less than existing one having 0.081 as shown in TABLE II. So it is concluded that the time complexity of DES (proposed pseudo random binary sequence based key) is less than DES (existing).

TABLE II. ANALYSIS OF TIME COMPLEXITY

ALGORITHMS	TIME COMPLEXITY	REMARKS
DES (EXISTING)	0.081	LESS
DES (PROPOSED)	0.054	BETTER

2) Analysis of randomness and security: From the Matlab simulation, it is noticed that in case of DES security for encryption/decryption, the proposed pseudo random binary sequence (having a frequency mono bit test p-value of 0.607) is better than existing (having a frequency mono bit test p-value of 0.382) pseudo random binary sequence as shown in TABLE III. At different instance of simulation and with different time interval, different binary bit sequence gets generated thereby changing the key at each time and also the cipher text changes periodically (shown in Figure 5 and Figure 6). As a result the pure randomness is maintained which affect the security of the cryptography to a high extent making a better random cryptography.

Table III. ANALYSIS OF RANDOMNESS AND SECURITY

PARAMETERS	EXISTING WORK	PROPOSED WORK
KEY SIZE	64 bits	64 bits
FREQUENCY MONO BIT TEST FOR RANDOMNESS	0.382 [7]	0.607
POSSIBLE COMBINATION TO CRACK	2^56	2^56 (random)
REMARKS	AVERAGE	BETTER

VI. CONCLUSION

A DES algorithm which is computationally fast and cryptographically secure because of pseudorandom binary sequence generated key has been proposed and presented in this manuscript. DES algorithm is implemented by mixing randomly generated key bits with message bits and from the implemented randomness tests, it has been proved that the generated sequences are unpredictable in nature and passed successfully NIST frequency monobit test suites. The algorithm is implemented in Matlab language on a 64 bits word length computer. Hence, excellent performance based on security (having a frequency mono bit test for randomness p-value of 0.607) and time complexity (0.054) is achieved.

REFERENCES

[1] Bruce Schneier, "Applied Cryptography", Second Edition, Wiley Computer Publishing, 1996.
[2] Coppersmith, "The Data Encryption Standard (DES) and Its Strength Against Attacks", IBM Journal of Research and Development, pp.243 – 250, May 1994.
[3] Mehlrotra and Mishra, "Comparative Analysis of Encryption Algorithms For Data Communication", IJCST Vol.2, Issue.2, June 2011.
[4] Schaumuller, "Cryptanalysis of the Data Encryption Standard by The Method of Formal Coding", In Advances in Cryptology Eurocrypt 82 Proceedings Springer Verlag, pp.235-255, 1983.
[5] Biham and Shamir, "Differential Cryptanalysis of the Full 16-round DES", Lecture Notes in Computer Science: Advances in Cryptology Proceedings of CRYPTO Springer Verlag, pp.487-496, 1993.
[6] Ahmad and Izharuddin, "Randomness Evaluation of Stream Cipher for Secure Mobile Communication", International Conference on Parallel, Distributed and Grid Computing, pp. 165–168, 2010.
[7] Ahmad and Farooq, "Chaos Based PN Sequence Generator for Cryptographic Applications", International Conference Signal Processing and Communication Technologies, pp.83-86, 2011.
[8] N. Ruggeri, "Principles of Pseudo-Random Number Generation in Cryptography", 2nd edition, 2013.
[9] K. Chandra Sekhar, K. Saritha Raj, "An Efficient Pseudo Random Number Generator for Cryptographic Applications", International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-4 Issue 1,October 2014.
[10] T. Siegenthaler, "Decrypting a Class of Stream Ciphers Using Cipher text Only", IEEE transactions on computers, vol. c-34, no. 1, January 2011.
[11] R. S. Katti and R.G. Kavasseri, "Secure Pseudo-random Bit Sequence Generation using Coupled Linear Congruential Generators", IEEE transactions on computers, PP. 4-8, 2014.
[12] Alka Sawlikar, Manisha Sharma, "Analysis of Different Pseudo Noise Sequences", International Journal of Computer Technology and Electronics Engineering (IJCTEE), Volume 1, pp. 67-74, 2012.
[13] A. Fuster Sabater , L.J. Garc A Villalba, "An efficient algorithm to generate binary sequences for cryptographic purposes", Theoretical Computer Science,pp.679–688, 2013.
[14] E. Cruselles, M. Soriano, and J. L. Melus, "Uncorrelated PN Sequence Generator for Spreading Codes in CDMA Systems", Sixth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 3, pp.1335-1340, 2012.
[15] M. Agrawal, "Cryptography: A Survey", IETE Technical Review, Vol 16, Nos. 3&4, pp 287-296, May-August 1999.
[16] H. Abunahla, D. Shehada, C. Y. Yeun, B. Mohammad and M. A. Jaoude, "Novel secret key generation techniques using memristor devices", AIP Advances 6, 025107, pp.1-10, 2016.
[17] A. Lee, "Guideline for Implementing Cryptography in PN Sequences in the Federal Government", National Institute of Standards and Technology, NIST Special Publication 800-21, November 2009.
[18] W. Schindler and W. Killmann, "Evaluation criteria for true (physical) random number generators used in cryptographic applications", Cryptographic Hardware and Embedded Systems – CHES2002, Springer-Verlag, vol. 2523, Lecture Notes in Computer Science, pp. 431–449, 2003.