Abstract. Cryptography is a type of art that translate strings of plain text into strings of cipher text and is very important in sharing of secret information in communication system. It is very essential to have cryptographic algorithms to provide high security by utilizing optimum computational resources during information sharing to achieve data confidentiality, privacy, integrity and other security services. The traditional conventional encryption algorithms (DES, AES) have high computational cost due to generating number of sub keys and bilinear concept. Hence this paper propose the sequential state machine (Mealy state Machine) based cryptosystem which is very light weight in nature and provide high security due to its weakly invertability concept with relatively smaller key size. The paper also presents the performance analysis on proposed algorithm and describes the security analysis on it.

Keywords: Data Security, Confidentiality, Privacy, Encryption, Decryption, Finite automata, Finite state Machine, Sequential state machine, Mealy Machine, Integrity, Cryptography, security.

1. Introduction
Cryptography is an art of language used for transmitting the secret writing[1]. The main idea of cryptography is to translate meaningful message or sentence to uncommon message which is not understood by unauthorized person. This process is useful only when hard to retrieve the original message. The cryptography acts as middle man between sender and receiver who take care of secrecy of information being transmitted. In the cryptography technology, the original or human understandable message is called "Plain text", the translated message is called "cipher text or crypto text" and the technique is used for the translation process is called "Encryption and decryption" technique. The research scientists have proposed different hybrid encryption techniques of symmetric or one key encryption[2, 3] (DES, AES, Blowfish, Twofish etc.) and Asymmetric or public key encryption (RSA, ECC, Elgamal, ABE, IDE, FKE, QKE, HE, etc.). The basic objective of these techniques to provide confidentiality, privacy, integrity, authentication, authorization, auditing, non-repudiation services[4, 5] over the data transmission in insecure channel.
All these techniques are based on number theory which might cost heavily due to utilization of large number of computational resources, bi-linear theory and logarithmic concept. Due to this reason, many researchers have shown their interest towards the state machine based cryptosystem. The mathematical model of finite state machine based cryptography is widely studied and presented in [6]. Authors have presented the one-key and public key based encryption and decryption technique. The idea of state machine based cryptosystem is very simple. The encryption process will be depends on the state machine that acts as key space for encryption process and the inverse of state machine acts as key space at receiver end for decryption process. the security of the system relies on the difficulties in constructing the inverse machine to decrypt the information.

The traditional cryptography techniques are based on number theory[7] gives more computational cost in terms of time complexity. The authors of [8, 9, 10, 11, 12, 13] have started proposing the cryptosystem based on cellular automaton and later finite state automaton based cryptosystem - a mathematical model has been presented by the author[6]. The author also described that, the applications of finite state machine can be used for one-key and two-key cryptosystem. In this paper we discusses the performance of one-key cryptosystem based on sequential state machine(Mealy state machine) for different size of plain text. This paper also gives an insight into security analysis on proposed methods.

1.1. Motivating Example
One key or shared key cryptosystem is a method of encryption and decryption process that is used for secrecy communication between two players[14]. This cryptosystem is faster than two-key cryptosystem - consumes the lot of network and CPU resources due to its bi-linear concept used in key generation algorithm and one-key cryptosystem is also efficient to provide confidentiality and privacy of information being transmitted in communication channel[14]. Due to the better efficiency and faster in time complexity than two-key cryptosystem, it is used in encrypting huge amount of data being exchanged between client and server.

The security of existing one-key cryptosystem[2, 3] (DES, AES, Blowfish, RC4) is depends on length of key, bi-linear theory and logarithmic concept of number theory. This will leads to significant delay in encryption and decryption process even though these cryptosystems are highly effective[15]. The researchers [6, 9, 10, 11, 12, 13] have concentrated only on two-key cryptosystem based on finite state machine. It is very essential to have one-key cryptosystem that consumes less resources and provide faster encryption and decryption mechanism during data in-transit. Consider the scenario of HTTP client server secure communication[19] as a motivating example where user as a client would like to make online money transaction through the online banking system. The customer enters url of netbanking in browser that will initiate HTTP connection between customer and bank server. The bank server takes the login credentials and verify. Once the customer credentials are accepted, the HTTP protocol(at Client side) shares the secret key (shared key) to bank server as session key to have secure communication between customer and bank server. The all communication between customer and bank server will be encrypted and decrypted by using session key at both the ends[20]. It is very important that, the encryption and decryption algorithm must be faster and efficient one for large data set. Hence we propose a very light weight one-key cryptosystem for encryption and decryption process based on sequential state machine. The system has relatively small key size and higher security due to its weakly invertability concept.

The rest of the paper is organized as follows: Section 1 briefs about the introduction on cryptography and its applications and about sequential state machine. The sequential state machine based one-key cryptosystem is explained in section 2 with proposed method and algorithm. The section 3 discusses the experimentation details and security analysis on proposed method. The section 4 gives you the related work on cryposystems based on state machines.
proposed by researchers and scientists. Finally we conclude the paper in section 5.

2. Cryptosystem based on Sequential state Machines
Sequential state machine changes their states and produces the output sequence for each input symbols[21]. The behavior of the sequential state machine are represented by set of components: Set of states, set of input symbols, set of output symbols, start state and final state. There are two types of sequential state machines - Moore Machine and Mealy machine and these machines are also called as finite state machine since the states in the machines are finite. These machine are used as cryptography techniques due to its generation of output sequence for each input sequence. Most of the proposed cryptography uses the applications of cellular automata, transducers (More and Mealy Machine) and state machines with accepting state and without output symbols. It means to say that, the system provides output sequence for each input sequence at different states are called sequential state machines. These Moore and Mealy machines are also called as finite state machine since the states in the machines are finite. The sequential state machines (Moore and Mealy) $M$ consists of six tuples i.e. $M = (Q, I, O, \delta, \lambda, q_0)$ where the tuples $Q, I, O, q_0$ represents collection of states, input symbols, output symbols and an initial state respectively. The tuple $\delta, \lambda$ are one way mapping function where $\delta$ defines $\delta : Q \times I \rightarrow Q$ and $\lambda$ defined as $\lambda : Q \rightarrow O$ for moore machine and $\lambda : Q \times I \rightarrow O$ for mealy machine. The mapping function are expanded as follows:

(i) The Mapping function $\delta$ for any given state $q_0 \in Q$ and the input sequence $a_0a_1....a_{l-1} \in I$ with length $l > 0$, then we get next states $q_i \in Q$ i.e $\delta(q_0, a_0a_1a_2....a_{l-1}) = q_{i+1} \in Q$ where $q_0$ is the current state and $q_{i+1}$ is the next states for each input sequence $a_0, a_1, a_2....a_{l-1} \in I$. Consider the transition table of $M$ shown in table 1 and string 10010, then

$$\delta(A, 1) = C \quad \delta(C, 0) = A \quad \delta(A, 0) = B \quad \delta(B, 1) = D \quad \delta(D, 0) = A$$

(ii) The mapping function $\lambda$ for moore and Mealy machines are defined as

(a) The mapping function for mealy machine $\lambda$ with any state $q_0 \in Q$ and the input sequence $a_0a_1a_2....a_{l-1} \in I$ with length $l > 0$, then we get sequence of output symbols $b_0, b_1, b_2....b_{l-1} \in O$ i.e $\lambda(q_0, a_0a_1a_2....a_{l-1}) = b_0b_1b_2......b_{l-1}$ where $b_0b_1b_2......b_{l-1} \in O$ are the set of output sequence for the input sequence $a_0a_1a_2....a_{l-1} \in I$. This is also equivalent to $\lambda(\delta(q_0, a_0), a_1a_2....a_{l-1}) = b_0b_1b_2......b_{l-1}$. In this length of output sequence must be same as length of input sequence. Consider the transition table of $M$ shown in table 1 and string 10010, then

$$\lambda(A, 1) = 1 \quad \lambda(C, 0) = 0 \quad \lambda(A, 0) = 0 \quad \lambda(B, 1) = 0 \quad \lambda(D, 0) = 1$$

i.e

$$\lambda(A, 10011) = 10001$$

(b) For moore machine, the mapping function $\lambda$ is with any states $q_0, q_1,....q_n$, then we get sequence of output symbols on each state $b_0, b_1, b_2....b_{l-1} \in O$ i.e $\lambda(q_i) = b_0b_1b_2......b_{l-1}$ where $b_0b_1b_2....b_{l-1} \in O$ are the set of output sequence for the input sequence $a_0a_1a_2....a_{l-1} \in I$ on each state $q_i$.

We are using mealy state machine in our proposed method. The applications of Mealy state machine (MSM) is used for encoding (Encryption) and decoding (Decryption) of crytostystem. It is very essential to obtain inverse of MSM $M = (Q, I, O, \delta, \lambda, q_0)$ can be used for decoding process. One of the most important issue in information coding and transmission is to reconstruct the input sequence (data transmitted) from output sequence at receiver end. We use
MSM based encoder and decoder to encrypt and decrypt the data at sender and receiver end respectively. That means, receiver must have inverse of MSM to reconstruct the input sequence from the received output sequence.

The inverse of MSM is a state machine that produces the input sequence with finite delay. If machine is information loss less then only inverse of MSM can be constructed. Therefore, first the MSM must be verified whether it is information lossless or not [23]. A MSM is said to be information loss less then only inverse of MSM can be constructed. Therefore, first from the received output sequence.

A MSM based encoder and decoder to encrypt and decrypt the data at sender and receiver end pair of states. Consider the states in the MSM are $Q$ and $b$ text (output sequence).

The plain text (input sequence) $a_0a_1a_2....a_n$ is reconstructed at the receiver end for the output sequence $b_0b_1b_2....b_n$ i.e $\lambda'(q, b_0b_1b_2....b_n) = a_0a_1a_2....a_n$, the output mapping function can be written as $\lambda'(q, b_i) = a_i$. Consider the MSM with four states ($A, B, C, D$) where $A$ is initial state and $B$ is final state and is also a loss less, the inverse of $M$ is shown in the table 1 for understanding purpose. The table 2 shows the encoding(Encryption) and decoding(Decryption) process by using MSM for the sample input sequence 11100101 and the same sequence is reconstructed at receiver end from the output sequence 11000011. The transition movement along with output bit generation for each input bit is also presented in table 2.

The plain text (input sequence) $a_0a_1a_2....a_i$ will be represented over the $GF(2)$ or sequence of binary bits. These binary bits will be input symbols to MSM $M = (Q, I, O, \delta, \lambda, q_0)$ that encrypts it and generate the cipher text (output sequence). Before generating output sequence, the permutation function is applied to plain text to construct the randomness of cipher text as shown in figure 1(a). The encoder in figure 1(a) takes plain text $a_i$ and apply the permutation function $g(a_i)$ on plain text $a_i$ and give it to MSM $M$. The MSM $M$ generates the cipher text (output sequence) $b_i$ with help of output mapping function $\lambda$ and the same is shown in figure 1(a). The plain text will be reconstructed at the receiver end by using inverse of MSM $M' = (Q', I', O', \delta', \lambda', q_0')$ as shown in figure 1(b). The decoder in figure 1(b) takes cipher text $b_i$ and apply the inverse permutation function $g^{-1}(b_i)$ on $b_i$ and give it to inverse of MSM $M'$. The $M'$ generate the corresponding output sequence as plain text for input sequence (cipher text) $b_i$ received from encoder and the same is shown in figure 1(b).

The algorithm 1 presents the step wise step for encryption and decryption process by using MSM and the size of key space is used in the algorithm is length of states in MSM. If size of each state is one byte then the size of key space will be $K = |Q| \times 8$. Consider the same scenario discussed in motivating example, the plain text over $GF(2)$ is communicated between bank customer and bank server for transmission of 11101101100101011...... The plain text 11101101100101011...... is permuted by using function[6] $g$, get the intermediate plain text and use the mealy state machine for encryption. The state machine shown in table 1 for example purpose will encrypt intermediate plain text 101101101010001...... with 32 bit key size and produce the cipher text as follows

$$C_i = E_M(101101101010001....)$$

$$\lambda(A, 101101101010001....) = 10111111100100010000100010001000......$$

The cipher text for the intermediate plain text is 1011111110010001000.... and client at the sender end will transmit it to the bank server. The bank server use the inverse of MSM $M'$ as shown in table 1 just for reader understanding and decrypt the cipher text to retrieve the plain text as follows and The pictorial representation of the same is shown in figure 2

$$MSM = E_{M^{'}}^{-1}(C_i) = D_{M'}(C_i)$$
\[ \lambda'(A, 101111111001000\ldots) = 101101101010001\ldots. \]
The inverse permutation function \(g^{-1}\) is applied on message 101101101010001\ldots to get the plain text i.e
\[ g^{-1}(101101101010001\ldots) = 111011011100101\ldots. \]

Figure 1. Encryption and Decryption process based on Mealy State Machine

**Algorithm 1:** Encipher and Decipher Algorithm
1. Let \(P\) be plain text over \(GF(2)\);
2. Select \(M = (Q, I, O, \delta, \lambda, q_0)\) is MSM and is weakly invertible;
3. Define the Key space \(K\) with length \(l = |Q \times 8|\) and \(K = A, |K| = l\);
4. Encryption:
   i. Apply the permutation function \(g[6]\) on \(P\), \(MSG = g(P)\)
   ii. Apply MSM \(M\) to generate Cipher text: \(C_i = E_M(\text{MSG})\)
      \[ E_M = \lambda(q, a_i) = \lambda_{a_i}(q) \quad a_i \in MSM \]
      \[ C_i = \lambda(q, a_i) = \lambda_{MSM}(q) \]
5. Decryption:
   i. Get \(C_i\) cipher text
   ii. Compute inverse of \(M\) i.e \(M' = (Q', I', O', \delta', \lambda', q'_0)\)
   iii. Apply \(M'\) to get Message
      \[ MSG = E_{M'}^{-1}(C) = \lambda'(q, C_i) = \lambda'_{C_i}(q) \]
   iv. Apply the inverse permutation function \(g^{-1}\) on \(MSG\) to get Plain Text \(P\)
      \[ P = g^{-1}(MSG) \]

3. Experimental setup and results discussion
The performance analysis of proposed method based on sequential state machine (mealy state Machine) have been implemented and tested in DELL Inspiron 14 3000 series having Intel(R) Core(TM) i3-4005U CPU@1.70GHz with installed primary memory(RAM) of 4GB(3.41GB usable) under 32 bit operating system running in Windows 7 Home basic of service pack1. The designed encryption algorithm is implemented in python 3.6 version. The mealy state machine used in proposed method that consist of set of states \(Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}\), input and
output symbols $I = O = \{0, 1\}$ and the start state $q_0$. That means to say that, we considered $|Q| \times 8 = 64$ bit key space for the encryption and decryption process and the number of rounds will be depending upon number of plain text block size.

The plain text (Input sequence) $a_0a_1a_2......a_i$ will be represented over $GF(2)[2, 3]$ or sequence of binary bits. These binary bits are inputs to the MSM that encrypt plain text (msg in figure 3) and produce cipher text (cipher in figure 3) and original plain text (in figure 3) can be retrieved back from cipher text, the same is as shown in figure 3. It also apply the permutation function $g(a_i)$ to binary bits of plain text. The plain text is divided into number of blocks of size 64 bits. The performance has been conducted for both DES and designed algorithm in python 3 interpreter with above mentioned system configuration and speed of proposed method has been observed for different size of plain text and same is presented in the table 1 and figure 4. The computer system with above configuration was running with 97 process with CPU usage 51% and physical memory 73%. The table 3 and figure 4 shows that, the proposed method has better time complexity compare to DES algorithm in encryption and decryption process.
3.1. Security Analysis
The proposed cryptography based on mealy state machine has variable length of key space is depends on the number of states in the machine. If number of states in the MSM is 8, then size of key will be 64 bit. The key space is of variable length depends on the number of states present in the state machine and same is used in the proposed method. The security of algorithm will be increased if MSM has more number of states. Since entire MSM is a key space in the encryption and decryption process, it is very difficult to find the decryption key for eavesdropper due to its weakly invertable concept. In our implementation, we define 8 states, that means to say that size of key is 64 bit in the designed algorithm. The eavesdropper need $2^{64} \times 8$ time to get encryption key. Even if eavesdropper is able to get the encryption key, but it is very hard to get inverse of MSM. Hence the security of proposed method is completely depends on the Mealy state machine and number of states used in it.

4. Related Work
The sequential state machines are the models that takes the sequence of inputs and produces the sequence of outputs. These models are used as cryptographic techniques due to its generation of outputs sequence for the input sequences. The primary concept of sequential state machine based cryptosystem is very simple: The encryption key will be a state machine itself and inverse of it is the key for decryption. The security of whole system is depends on the difficulties in identifying state machine and inverse of given state machin[24]. Most of the proposed cryptography uses the applications of cellular automata, transducers (More and Mealy Machine) and state machines with accepter and without output symbols. The author of [8] has proposed the cellular automata based cryptosystem in 1985 to generate pseudorandom stream of bits $C_i$, then later the sequence of bits will be added to plain text $a_i$ for cipher text generation i.e $C_i = P_i \oplus a_i$. The decryption process is found in [16] and in this method, if plain text sequence $a_i$ is known to adversary, then known-plain text attack can be possible. To overcome this problem, the author of [25] presented cryptosystem based on reversible cellular finite automata. The cipher text will be generated by using cellular finite automata, but the key size used in this method is big problem[24]. The other cryptography techniques based on cellular automata is found in [9, 10, 11, 12, 13].

The author of [26] has presented novel encryption technique for sharing of secrete information by using finite state machine. The author has used the recurrence relation matrix to get the cipher text, but there is no performance testing of proposed automata based cryptography. These systems depends on constructing the inverse of it. The author of [27] has developed and presented one-key cryptosystem by using extended mealy state machine. However, it is observed that, the brute force attack is possible on the proposed method and also it is slower than the DES encryption alogrithm. Then later the author [6] presented a mathematical model of cryptography (one-key and two key) based on extended finite state automata. The authors presented different mathematical models of public key cryptosystem like FAPKC, FAPKC2 and FAPKC3 based on linear and non-linear finite state automata.

The stream cipher technique based on Rabbin-scott model of finite state machine is presented in [17] i.e state machine without output symbols. The proposed method suffers from the practical difficulties that effects the performance of algorithm. The author of [7] has presented DES encryption technique based on finite automata for plain text as well as images. The author have used finite state machine for generating 16 sub-keys for encryption and decryption process and also presented the performance testing for different texts size and images. The author of [18] have proposed finite state machine based block cipher to exploit the idea of parallel computation and introduce the new block cipher that perform well with high security. This block cipher depending upon the moore machine that is used for encryption and decryption process. The Moore machine based cryptography is also presented in [28]. The paper presents the secret sharing technique by using fibonacci recurrence matrix to provide higher security so
that, no adversary can exploit the cipher text. This paper lacks of proof and experiment results on the proposed method.

Based on the observations made in related work, the most of the authors have concentrated only on public key based cryptosystem based on finite state machine which are relatively slow compare to DES encryption algorithm and exploited to cryptanalytic attacks. Hence we propose symmetric key (one key) encryption technique based on sequential state machine (Mealy Machine) which is faster than DES and provide higher security due to weakly invertible state machine concept.

5. Conclusion

The one-key cryptosystem is faster than two-key cryptosystem consumes the lot of network and CPU computational resources. One-key cryptosystem is also efficient to provide confidentiality and privacy of information being transmitted in communication channel. The traditional conventional one-key cryptosystems are in high computational cost due to security dependency on size of key, key generation algorithm and bilinear concept. It is very essential to reduce the time complexity of cryptosystem during data communication system. The proposed sequential state machine - Mealy State Machine(MSM) based one-key cryptosystem that reduces the time complexity 30% w.r.t DES algorithm that utilizes the optimal computational resources and provide the higher security due to difficulties in identifying MSM M as well as inverse of MSM M'. The proposed algorithm is explained with help of motivating example and then implementation results are compared with DES block cipher algorithm with 64 bit key size, the proposed method is faster than the DES algorithm.

This work can be further expanded by using extended mealy machine with additional customize excitation function to provide higher randomness and higher security.

[1] Valtteri Niemi. Cryptology: language-theoretic aspects. In Handbook of Formal Languages, pages 507–524. Springer, 1997.
[2] William Stallings. Cryptography and network security, 4/E. Pearson Education India, 2006.
[3] Behrouz A Forouzan. Cryptography & network security. McGraw-Hill, Inc., 2007.
[4] Dorothy Elizabeth Robling Denning. Cryptography and data security. Addison-Wesley Longman Publishing Co., Inc., 1982.
[5] Douglas R Stinson. Cryptography: theory and practice. Chapman and Hall/CRC, 2005.
[6] T Renji. Finite automata and application to cryptography, 2009.
[7] Sarshad Abubaker and Kui Wu. Dafa-a lightweight des augmented finite automaton cryptosystem. In International Conference on Security and Privacy in Communication Systems, pages 1–18. Springer, 2012.
[8] Stephen Wolfram. Cryptography with cellular automata. In Conference on the Theory and Application of Cryptographic Techniques, pages 429–432. Springer, 1985.
[9] Joan Daemen, René Govaerts, and Joos Vandewalle. A framework for the design of one-way hash functions including cryptanalysis of damgård’s one-way function based on a cellular automaton. In International Conference on the Theory and Application of Cryptology, pages 82–96. Springer, 1991.
[10] Ivan Bjerre Damgård. A design principle for hash functions. In Conference on the Theory and Application of Cryptology, pages 416–427. Springer, 1989.
[11] Howard Gutowitz. Cryptography with dynamical systems. In Cellular Automata and Cooperative Systems, pages 237–274. Springer, 1993.
[12] S Nandi, BK Kar, and P Pal Chaudhuri. Theory and applications of cellular automata in cryptography. IEEE Transactions on computers, 43(12):1346–1357, 1994.
[13] P Guan. Cellular automaton public-key cryptosystem. Complex Systems, 1(1):51–56, 1987.
[14] Dawn M. Turner Peter Smirnoff. Symmetric key encryption - why, where and how it’s used in banking, January 2019.
[15] Symmetric cryptography, 2019. [Online; accessed 12-June-2019].
[16] Willi Meier and Othmar Staffelbach. Analysis of pseudo random sequences generated by cellular automata. In Workshop on the Theory and Application of Cryptographic Techniques, pages 186–199. Springer, 1991.
[17] P Domnosi. A novel cryptosystem based on finite automata without outputs. In Automata, Formal Languages And Algebraic Systems, pages 23–32. World Scientific, 2010.
[18] Gh Khaleel, SHERZOD TURAEV, MI Mohd Tamrin, et al. A new block cipher based on finite automata systems. *International Journal on Perceptive and Cognitive Computing*, 2(1), 2016.

[19] National Academies of Sciences Engineering and Medicine. *Decrypting the Encryption Debate: A Framework for Decision Makers*. The National Academies Press, Washington DC, 2018.

[20] Basics of cryptography: the practical application and use of cryptography, April 7, 2018. [Online; accessed 12-June-2019].

[21] J.S. Dhillon D.P. Kothari. *Digital Circuits and Design*. Pearson, 2016.

[22] John M Yarbrough and John M Yarbrough. *Digital logic: Applications and design*. West Publishing Company St. Paul, MN, 1997.

[23] Zvi Kohavi and Niraj K Jha. *Switching and finite automata theory*. Cambridge University Press, 2009.

[24] G Khaleel, S Turaev, I Al-Shaikhli, and MI Mohd Tamrin. An overview of cryptosystems based on finite automata. *Journal of Advanced Review on Scientific Research*, 27(1):1–7, 2016.

[25] Jarkko Kari. Cryptosystems based on reversible cellular automata. *Manuscript*, August, 1992.

[26] PA Jyotirmie, S Srilakshmi, A Chandra Sekhar, and S Uma Devi. A novel encryption for secret sharing using finite state machine. *International Journal of Mathematical Archive EISSN 2229-5046*, 4(6), 2013.

[27] Marc Gysin. A one-key cryptosystem based on a finite nonlinear automaton. In *International Conference on Cryptography: Policy and Algorithms*, pages 165–173. Springer, 1995.

[28] B Krishna Gandhi, A Chandra Sekhar, and S Srilakshmi. Cryptographic scheme for digital signals using finite state machine. *International journal of computer applications (September 2011)*, 2011.
Table 1. Mealy state machine with its inverse

| $\delta$ | $M$ - Transition Table | $\delta$ | $M'$ - Transition Table |
|---|---|---|---|
| $a$ = 0 | $a$ = 1 | $b$ = 0 | $b$ = 1 |
| $A$ | $B$, 0 | $C$, 1 | $A$ | $B$, 0 | $C$, 1 |
| $B$ | $C$, 1 | $D$, 0 | $B$ | $D$, 1 | $C$, 0 |
| $C$ | $A$, 0 | $D$, 1 | $C$ | $A$, 0 | $D$, 1 |
| $D$ | $A$, 1 | $C$, 0 | $D$ | $C$, 1 | $A$, 0 |

Table 2. Process of Enciphering and Deciphering

| Simple Enciphering and Deciphering Process |
|---|---|---|---|---|---|---|---|---|
| Input Sequence $a_i$ to $M$ | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| Transitions of $M$ on $a_i$ | $A$ $>$ $C$ | $C$ $>$ $D$ | $D$ $>$ $C$ | $C$ $>$ $A$ | $A$ $>$ $B$ | $B$ $>$ $D$ | $D$ $>$ $A$ | $A$ $>$ $C$ |
| Output Sequence $b_i$ from $M$ to $M'$ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Transitions of $M'$ on $b_i$ | $A$ $>$ $C$ | $C$ $>$ $D$ | $D$ $>$ $C$ | $C$ $>$ $A$ | $A$ $>$ $B$ | $B$ $>$ $D$ | $D$ $>$ $A$ | $A$ $>$ $C$ |
| Input sequence $a_i$ from $M'$: | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |

Table 3. Speed of DES and MSM algorithms for different size of plain texts

| PT in Bytes | ENC SPEED IN SEC | DEC SPEED IN SEC |
|---|---|---|
| | DES | PROPOSED | DES | PROPOSED |
| 64 | 0.046 | 0.0024 | 0.0623 | 0.0019 |
| 128 | 0.093 | 0.0078 | 0.1 | 0.0071 |
| 256 | 0.18 | 0.0153 | 0.171 | 0.0156 |
| 384 | 0.249 | 0.0156 | 0.265 | 0.0156 |
| 512 | 0.343 | 0.0468 | 0.327 | 0.0348 |
| 1024 | 0.655 | 0.0557 | 0.684 | 0.0498 |
| 4024 | 0.243 | 2.683 | 0.187 | 2.605 |
| 12288 | 7.21 | 4.913 | 7.132 | 4.780 |

![Figure 4. Performance comparison of DES and MSM encryption and decryption](image-url)