

# A New Digital Encryption Scheme

## Matrix Rotations and Bytes Conversion Encryption Algorithm

Sanikommu Krishna Reddy<sup>1</sup>

M.Tech Student

Department of Computer Science & Engineering  
VVIT, Nambur (V), Guntur (Dist.), India

R. Sudha Kishore<sup>2</sup>

Assistant Professor

Department of Computer Science & Engineering  
VVIT, Nambur (V), Guntur (Dist.), India

**Abstract--** Information security could be a difficult issue in today's technological world. There's a requirement for a stronger encoding that is extremely exhausting to crack. Completely different encrypted algorithms are planned thus far to come up with encrypted information of original information. In this we have proposed a new replacement algorithmic rule for Digital encoding called as "Matrix Rotations and Bytes Conversion Technique" (MRBC) which reduces size of the data as well as form of the data. The experimental results show that the new theme has very fast encoding and safer.

**Keywords--** Digital Information, Information security, Matrix Rotations, Bytes, Encryption, Decryption, Plain Text, Cipher text.

### I. INTRODUCTION

In today's world most of the communication is done using electronic media. Data security plays an important role in such communication. Hence, there is a need to protect data from malicious attacks. This can be achieved by Cryptography. Security of the information over the insecure mode of communication, Internet, has been an area of research for several years. There are several techniques developed for encryption/decryption of the information over the years [10][11].

In this paper we discuss a new technique for encrypting data which enables good diffusion and is having a unique technique of decrypting it back to the plaintext and is easy to implement using matrix rotations technique. The encryption algorithm of magic cube projected by Yongwei et al, which is implemented in three dimensional space [1][2][3]. The algorithm is complicated and very difficult to understand. F.Y.LI Min. Proposed queue transformation based digital image encryption algorithm [4], which works efficiently with low time complexity compared to Yongwei et al. M. Kiran Kumar, S.Mukthyar Azam proposed efficient digital encryption algorithm based on matrix scrambling technique [5], which works efficiently with low time complexity but drawback of this algorithm is don't change the form and size of the data. To overcome this drawback we proposed a "Matrix Rotations and Bytes Conversion Encryption Algorithm" (MRBC) Technique which is based

on random functions, shifting and reverse techniques of circular queue, with efficient time complexity.

In section II, Encryption Process is explained. In section III, Decryption Process is explained. In section IV, results are explained and finally Conclusion is provided in section V.

### II. ENCRYPTION PROCESS

Encryption is the process of encoding messages or information in such a way that only authorized parties can read it. In an encryption scheme, the message or information, referred to as 'plaintext', is encrypted using an encryption algorithm, generating 'cipher text' that can only be read if decrypted[6].

MRBC technique is especially concerned in 2 levels. In the 1<sup>st</sup> level we perform "8-to-7 Bytes" Technique. In 8-to-7 Bytes Technique, each 8 bytes of Text becomes 7 bytes of text. So, the advantage of this technique is it reduces the size of data. In the 2<sup>nd</sup> level we perform "Matrix Rotations Technique". In "Matrix Rotations Technique", we have to perform shifts and reverse operations on given data. In MRBC encryption process first off all, plain text is chooses which is from input text file and given binary equivalent of plain text as input to the "8-to-7 Bytes technique", each 8 bytes of Text becomes 7 bytes of text. So, the advantage of this technique is it reduces the size of data. The output of the "8- to-7 Bytes technique" is given as input to the "Matrix rotations technique". In this technique, we would like to perform two operations like Row Transformation and Column Transformation. In Row Transformation we are able to perform three operations like row left, row right and row reverse operations and conjointly we are able to perform three operations in Column Transformation like column up, column down and column reverse operations. Finally cipher text is obtained from encryption algorithm which is in Binary form so we can convert into ASCII equivalent of cipher text. The Fig 1 shows total encryption process of MRBC technique.

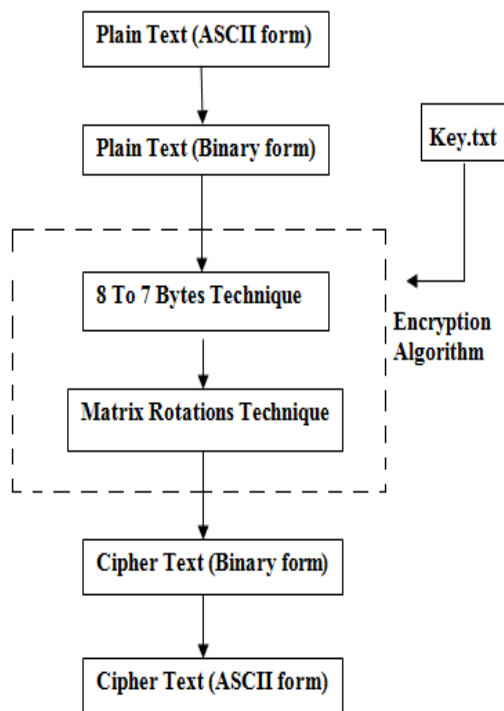


Fig 1: Block Diagram for Encryption Process

#### A. Algorithm for “8-To-7 Bytes technique”:

In “8-To-7 Bytes technique”, each 8 bytes of text becomes 7 bytes of text. So, the advantage of this technique is it reduces the size of data and change the form of data also.

The following are the steps for doing “8-To-7 Bytes technique”.

1. Read data into the matrix, ‘A’.
2. Read every 8 letters as one group and set count value with number of 8 letters groups.
3. Read binary equivalent 8 letters of the group into matrix  $B[ ][ ]$ .
4. Do,  $B[0][0] = B[7][7], B[1][0] = B[7][6], B[2][0] = B[7][5], B[3][0] = B[7][4], B[4][0] = B[7][3], B[5][0] = B[7][2], B[6][0] = B[7][1]$ .
5. Write ASCII equivalent values of Binary values on output file.
6. Check whether Count is  $> 0$  or not
  - i. If count  $> 0$  then move to ‘step 3’.
  - ii. Otherwise move to “step 7”.
7. End.

#### B. Algorithm for Matrix Rotations(MR) Technique:

In this technique, we would like to perform two operations like Row Transformation and Column

Transformation. In Row Transformation we are able to perform three operations like row left, row right and row reverse operations and conjointly we are able to perform three operations in Column Transformation like column up, column down and column reverse operations [7][8][9]. The Fig 2 shows total process of Matrix Rotations Technique.

The following are the steps for doing Matrix Rotation Encryption technique.

- a. Read data into the matrix, ‘A’.
- b. Set number of rounds as ‘w’.
- c. Generate random number as ‘i’ and takes binary sequence of that random number ( $b = \text{Binarysequence}(i)$ ).
- d. Let  $k = \text{Digit}(b_t)$ , where ‘t’ is bit position, k value is either 0 or 1, which deciding either to perform Row transformation or Column Transformation.
  - i. If  $k = 0$ , then Row transformation is performed.
  - ii. If  $k = 1$ , then Column transformation is performed.
- e. Check whether binary sequence in vector b is completed. If it is completed, again start from first digit in binary sequence of b.(LSB-->MSB).
- f. Repeat steps ‘d’ and ‘e’ for ‘w’ number of times.
- g. Record all sub keys sequentially in a key file.

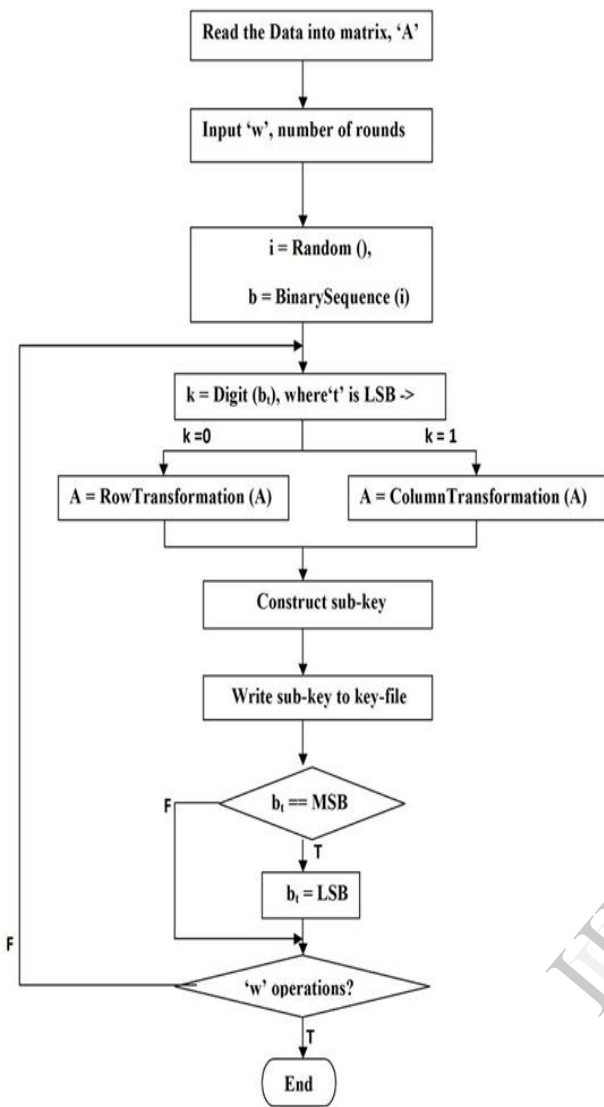


Fig 2: Block Diagram for Matrix Rotations Technique

1) **Row Transformation:** The Row transformation algorithm is shown in Fig 3, two rows are selected randomly,  $r1 = \text{Random}(m), r2 = \text{Random}(m)$  similarly two random values of columns  $c1, c2$  are selected i.e.  $c1 = \text{Random}(n), c2 = \text{Random}(n)$ , to determine the range of rows on which transformation has to be performed. The constraint here is  $r1 \neq r2$  and  $c1 \neq c2$ . Let  $x1 = \min(c1, c2), x2 = \max(c1, c2)$ , Here  $x1$  and  $x2$  becomes lowest index and highest index of the sub array selected in the rows  $r1$  and  $r2$ . Let  $op = \text{Random}() \bmod 3$ , hence  $op$  takes three possible values 0, 1 and 2. Thus three row operations are performed on the matrix. If  $op = 0$ , then circular left shift operation is performed on rows  $r1$  and  $r2$  on the sub portion of range  $x1$  to  $x2$ . If  $op = 1$ , then circular right shift operation is performed on rows  $r1$  and  $r2$  on the sub portion of range  $x1$  to  $x2$ . If  $op = 2$ , then perform reverse operation on the sub array of  $r1$  and  $r2$  in the range

from  $x1$  to  $x2$ . For each operation a sub-key is constructed and recorded in a key file.

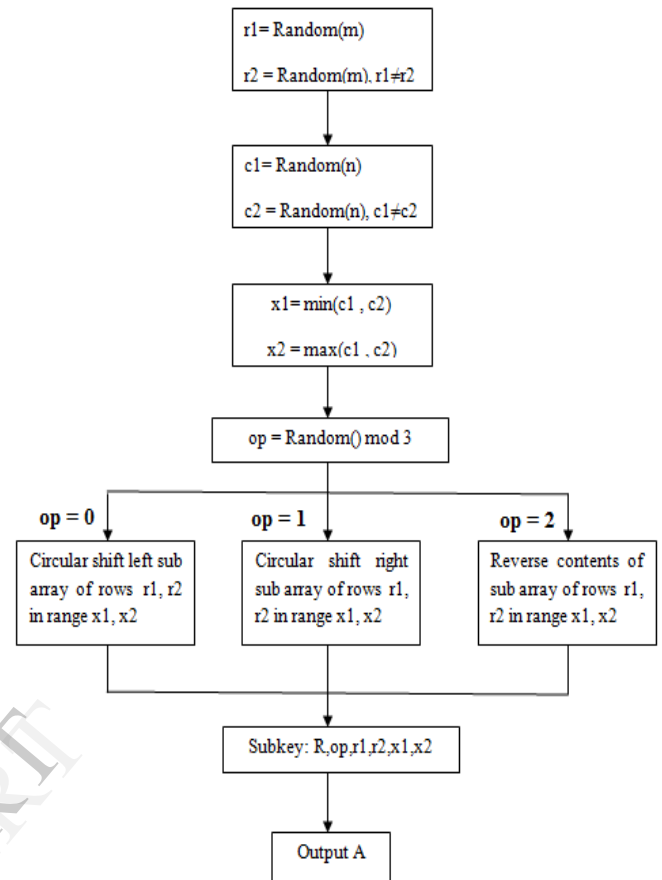


Fig 3: Row Transformation Process

2) **Column Transformation:** Column Transformation algorithm is shown in Fig 4, Two Columns are selected randomly,  $c1 = \text{Random}(n), c2 = \text{Random}(n)$  similarly two random values of rows  $r1, r2$  are selected i.e.  $r1 = \text{Random}(m), r2 = \text{Random}(m)$ , to determine the range of columns on which transformation has to be performed. The constraint here is  $c1 \neq c2$  and  $r1 \neq r2$ . Let  $x1 = \min(r1, r2), x2 = \max(r1, r2)$ , Here  $x1$  and  $x2$  becomes lowest index and highest index of the sub array selected in the columns  $c1$  and  $c2$ . Let  $op = \text{Random}() \bmod 3$ , hence  $op$  takes three possible values 0, 1 and 2. Thus three column operations are performed on the matrix. If  $op = 0$ , then circular upward shift operation is performed on columns  $c1$  and  $c2$  in the sub portion of range  $x1$  to  $x2$ . If  $op = 1$ , then circular downward shift operation is performed on columns  $c1$  and  $c2$  in the sub portion of range  $x1$  to  $x2$ . If  $op = 2$ , then perform reverse operation on the sub array of  $c1$  and  $c2$  in the range from  $x1$  to  $x2$ . For each operation a sub-key is constructed and recorded in a key file.

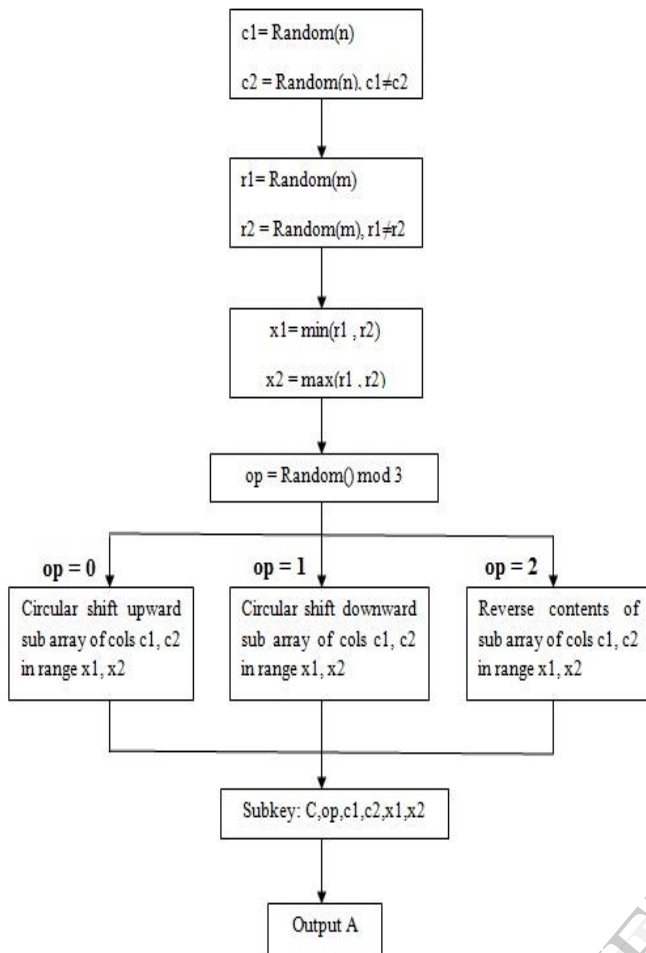


Fig 4: Column Transformation Process

### III. DECRYPTION PROCESS

Decryption is the reverse process of Encryption. Decryption process is done by reading the operations in the key file in reverse order and applying necessary operations on the matrix, which converts cipher text to get plain text. The cipher text is converted into its equivalent binary form and then it's arranged into a matrix of same order in encryption as  $m$  and  $n$ . The Fig 5 and Fig 6 show how the decryption process is done.

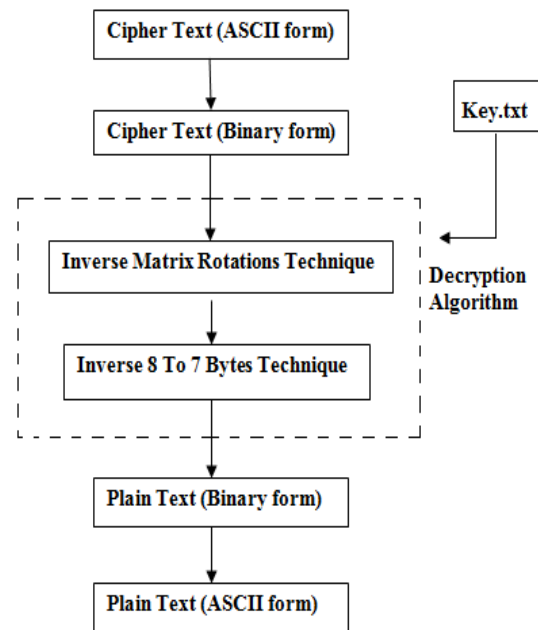


Fig 5: Block Diagram for Decryption Process

#### Algorithm for Decryption process:

Steps for Decryption Process explained in brief as follows.

1. The key file is partitioned by the R (row) operations and C (column) operations. Each line in key file can be considered as one sub key.
2. The sub keys are decrypted one by one from last sub key to first sub key in key file.
3. For each sub key  $T$ ,  $op$ ,  $\alpha_1$ ,  $\alpha_2$ ,  $\beta_1$ ,  $\beta_2$  values are obtained, where  $T$  represents either R or C operation, 'op' represents either 0 or 1 or 2 (if  $T$  is R then 'op' is shiftleft (0) or shiftright (1) or shiftreverse (2), if  $T$  is C then 'op' is shiftupward (0) or shiftdownward (1) or shiftreverse (2)). Based on 'T' value either R or C operation is decoded which are given as  $A = \text{InverseRowTrans}(A)$  and  $A = \text{InverseColTrans}(A)$ .
4. In  $\text{InverseRowTrans}(A)$  depending on the value of 'op' rotations are performed. For '0' RightShift, for '1' LeftShift and for '2' Reverse operations are performed on columns.
5. In  $\text{InverseColTrans}(A)$  depending on the value of 'op' rotations are performed. For '0' DownwardShift, for '1' UpwardShift and for '2' Reverse operations are performed on rows.
6. The steps from 2 to 5 are done until the key file is completed and end of process matrix  $A$  contains the required message in binary form.
7. The output binary message is given as input to "Inverse 8-To-7 Bytes Technique" and finally gets required message in binary form then converts

into its ASCII equivalent form which is called as required original message (i.e. Plain Text).

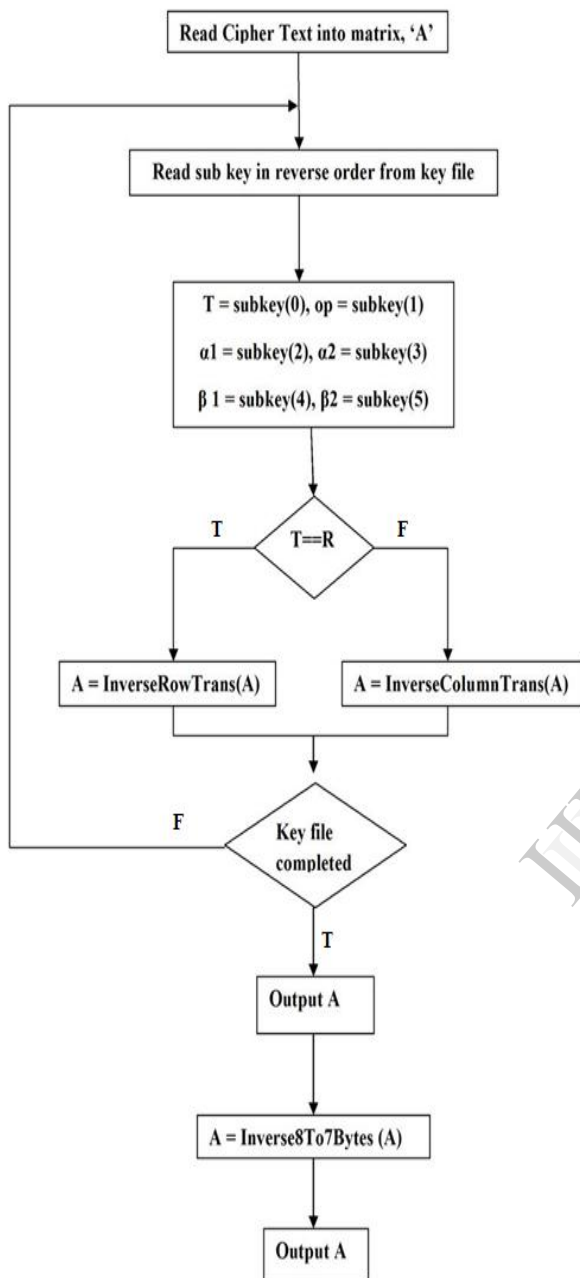


Fig 6: Block Diagram for MRBC Decryption Technique

#### IV. EXPERIMENTAL RESULTS

We have applied this proposed system on different sizes of data to encryption and decryption. Let's consider a sample data and apply MRBC technique as shown in Fig 7 and Fig 8.

##### A. Encryption process example

Let's consider a sample data and apply MRBC Encryption Technique on sample data which is shown in Fig 7.

In this example, the Original message (size 70 Bytes) is converted into its equivalent binary form and then given as input to MRBC technique. In MRBC technique, the binary equivalent of Original message given as input to "8-To-7 Bytes" Technique, which converts each 8 bytes of text to 7 bytes of text. So, the advantage of this technique is it reduces the size of data. The output of "8-To-7 Bytes" technique is given as input to "Matrix Rotations (MR)" technique which changes the form of the data. Finally, the output of MR technique (in Binary form) is converted into its ASCII equivalent and called as Cipher Text (size 67 bytes).

##### B. Decryption process example

The Cipher text which is get from MRBC encryption technique as output and given as input to MRBC Decryption Technique. The total decryption process was shown in Fig 8.

The Cipher text (size 67 Bytes) is converted into its equivalent binary form and then given as input to MRBC Decryption algorithm. In MRBC decryption technique, binary equivalent of cipher text is given as input to "Inverse Matrix Rotations (MR)" technique. In "Inverse MR Technique", depending on sub key value of 'T' we can perform either "Inverse row transformation" or "Inverse Column transformation". Output of "Inverse MR Technique" is given as input to "Inverse 8-To-7 technique". In Inverse 8-to-7 technique, each 7 bytes of text becomes 8 bytes of text. Finally, we obtained Plain text in binary form as output from "Inverse MR Technique" and then convert it into its ASCII equivalent form and called as Plain Text or Original message (size 70 Bytes).

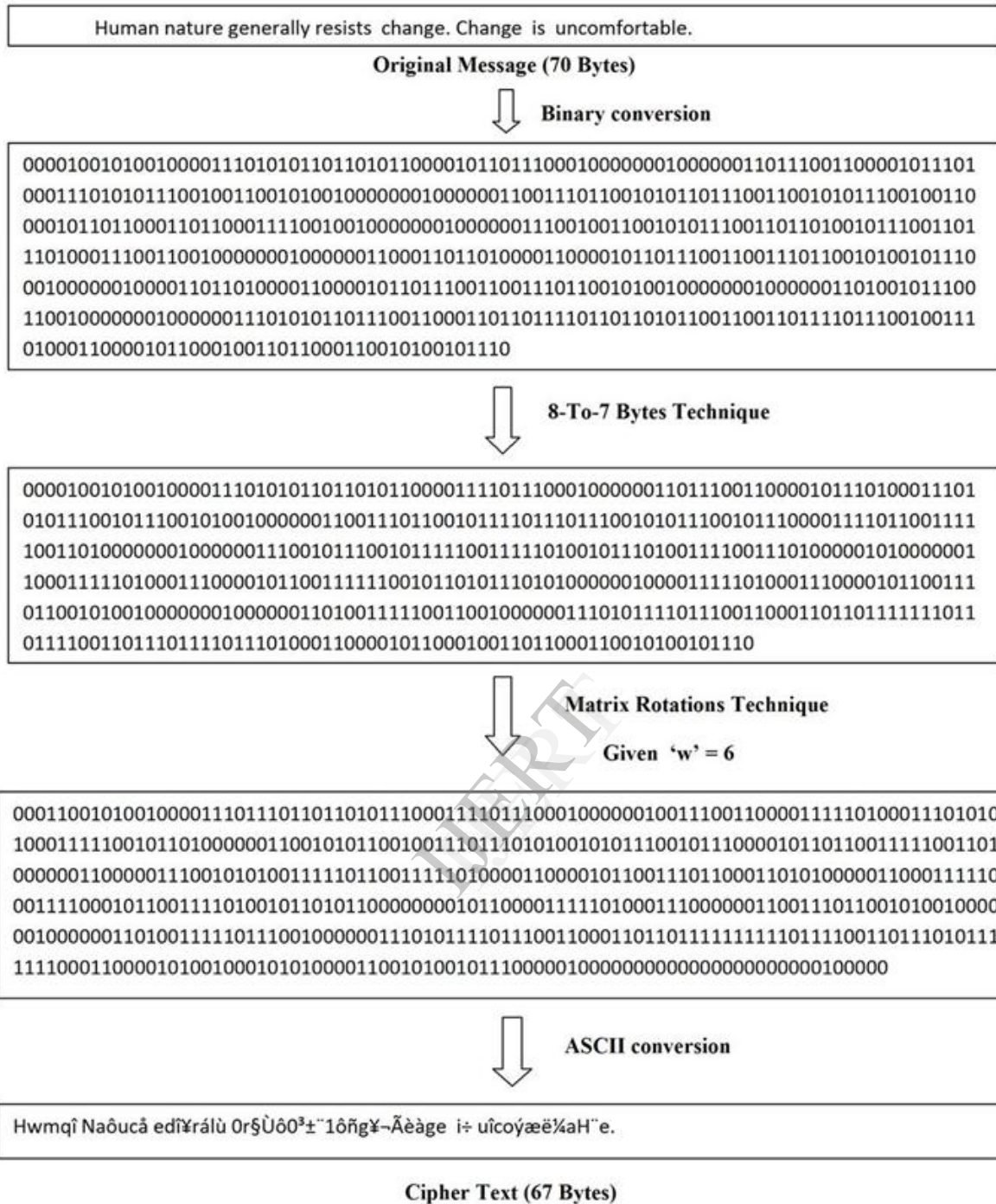


Fig 7: Encryption process with example

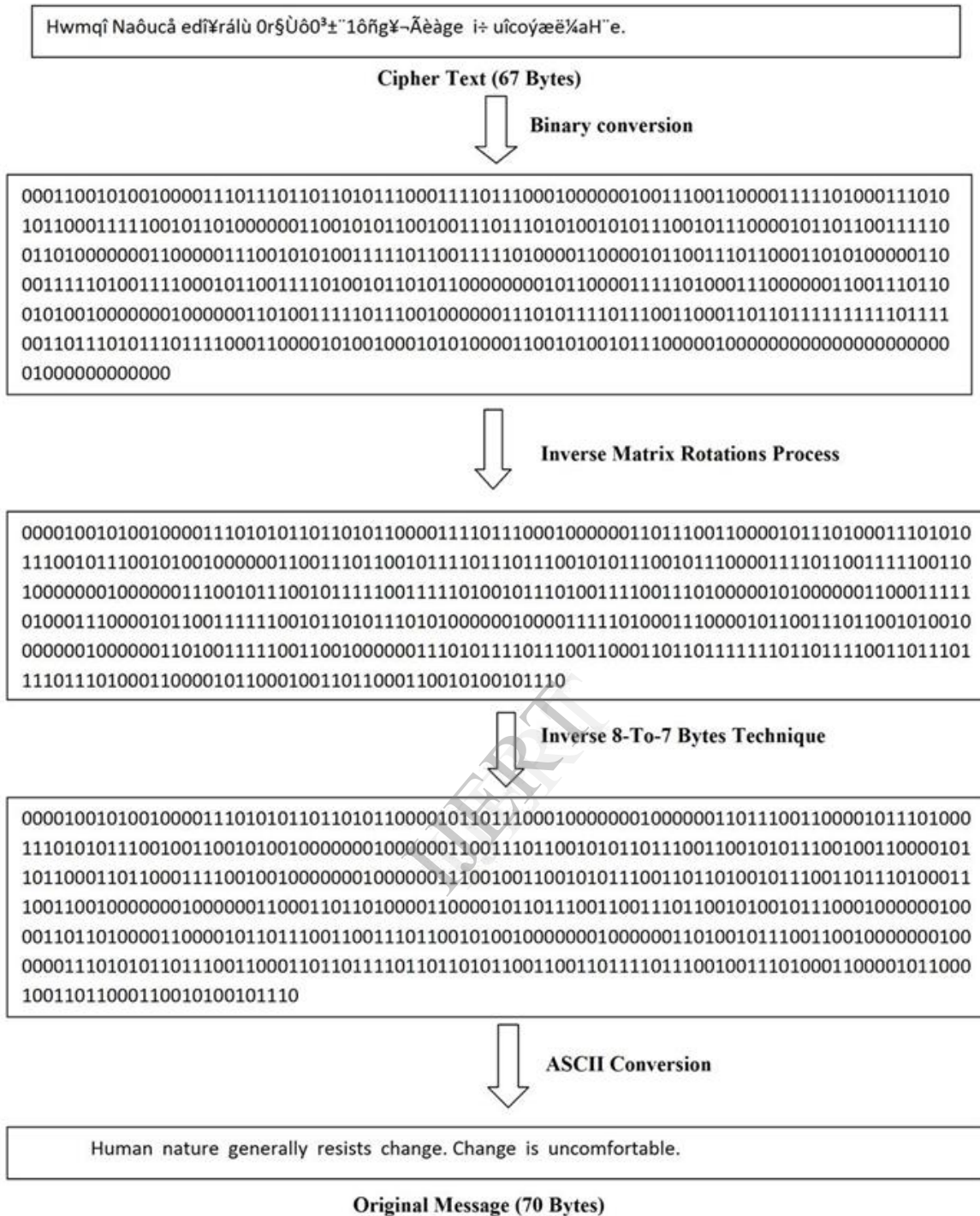


Fig 8: Decryption process with example

### C. Analysis

After encryption process the size of the data is reduced and after decryption process the size of the data is increases. The following Table 1 represented the various sizes of different data sets after encryption and decryption process.

TABLE 1

REPRESENTED THE VARIOUS SIZES OF DIFFERENT DATA SETS AFTER ENCRYPTION AND DECRYPTION PROCESS

S.No	Data Size (in Bytes)	After Encryption (Cipher Text in Bytes)	After Decryption (Plain Text in Bytes)
1.	64	61	64
2.	128	113	128
3.	256	232	256
4.	512	450	512
5.	1024	904	1024
6.	2048	1800	2048
7.	4096	3613	4096
8.	5120	4513	5120
9.	10240	8978	10240
10.	20480	17956	20480

### V. CONCLUSION

In this paper we presented an implementation of MRBC encryption algorithm. The main advantage of this algorithm was reduces size of data as well as change the form of data. The results show that the new theme has very fast encoding and safer with reduces the size of data. In our future work with this methodology was, MRBC technique is applicable for 0-127 ASCII values only. So, we can extend our methodology to remaining ASCII values also.

### REFERENCES

- [1] S.-J.-b. BAO Guan-jun and JI SHI-ming, (2002) "Magic cube transformation and its application in digital image encryption", Computer Applications, 11:22-25.
- [2] X.Y.CHEN Tao, (2005) "Design and implementation of encryption algorithm based on n-dimension magic cube", Journal of Information, 2:13-14.
- [3] C.C.LUO Pu, (2006) "An image encryption algorithm based on chaotic sequence", Journal of Information, 12.
- [4] F.Y.LI Min, (2005) "A New class of digital image scrambling algorithm based on the method of queue transformation", Computer Engineering, 01(31):148-149.
- [5] M. Kiran Kumar, S. Mukthyar Azam, "Efficient Digital Encryption Algorithm based on Matrix Scrambling Technique", IJNSA, Vol2, No.4, Oct-2010.
- [6] <http://en.wikipedia.org/wiki/Encryption>.
- [7] Z.L.Z.X. feng and FAN Jiu-lun, (2007) "A digital image encryption algorithm based on chaotic sequences", Microelectronics & computer, 2.
- [8] Ankita Agarwal (2012), "Secret key Encryption algorithm using Genetic algorithm", IJARCSSE April 2012.
- [9] Deepak nagde, Raviraj Patel, (2013) "New Approach for Data Encryption using two way Crossover", IJCSIT Vol. 4(1), 2013.
- [10] William Stallings "Cryptography and Network security", 3<sup>rd</sup> edition, Prentice-Hall Inc., 2005.
- [11] William Stallings, "Cryptography and Network Security: Principles & Practices", second edition.

### AUTHORS

**Sanikommu Krishna Reddy<sup>1</sup>** is pursuing Master of Technology in Computer Science and Engineering from JNTU Kakinada. He has received Bachelor of Technology in Information Technology form Acharya Nagarjuna University in 2012. His research interests are Information Security, Web Technology and Image Processing.



**R. Sudha Kishore<sup>2</sup>** is working as Assistant Professor in VVIT, Andhra Pradesh, INDIA. He has received B.Tech and M.Tech degrees from JNTU Hyderabad. This author has overall 9 years teaching experience and guided more than 10 innovative projects as a part of his academic work. His research interests are Image processing, Information security, security algorithms.

