ISSN:0975-9646

V.U.K. Sastry et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (1) , 2012, 3119 - 3128

A Modified Feistel Cipher involving a pair of key matrices,Supplemented with Modular Arithmetic Addition and Shuffling of the plaintext in each round of the iteration process

[1]V.U.K. Sastry, [2]K. Anup Kumar

[1]*Director School of Computer Science and Informatics, Dean(R & D), Dean (Admin), Department of Computer Science and Engineering, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, 501301, Andhra Pradesh, India.*

[2]*Associate Professor, Department of Computer Science and Engineering, Sreenidhi Institute of Science and Technology, Ghatkesar, Hyderabad, 501301, Andhra Pradesh, India*

Abstract: In this paper, we have offered a generalization to the classical Feistel cipher. Here we have taken the plaintext in the form of a rectangular matrix. This matrix is decomposed into two parts wherein each one is a square matrix. In these, one of the parts is multiplied with a pair of keys K and L. In each round of the iteration process, we have carried out a thorough shuffling of the binary bits of the resulting plaintext, so that, confusion and diffusion are created intensively. The cryptanalysis carried out in this investigation clearly indicate that this cipher is a strong one, and it can be comfortably applied for the security of information.

Key words: Encryption, Decryption, Key matrix, Mix, Permute and Modular Arithmetic Addition.

1. Introduction

In a recent investigation [1], we have modified the classical Feistel cipher [2] by converting the plaintext string into a matrix and by including a pair of key matrices as multiplicants of a portion of the plaintext matrix. In this, in addition to the usual XOR operation in Feistel cipher [3], we have introduced blending of the plaintext (multiplied by key matrices) in each round of the iteration process. The process of blending under consideration includes, in a way, both mixing and permuting of the binary bits of the key matrix and the plaintext matrix. This takes care of the features diffusion and confusion which are highly essential in the development of a cipher.

In the present paper, our objective is to modify the Feistel cipher by involving modular arithmetic addition [4]. In this analysis, we have included the function Shuffle ( ) which governs the shuffling process (a combination of mixing and permutation). This procedure is a variant of blending.

In what follows, we mention the plan of the paper. In section 2, we have introduced the development of the cipher, and presented the flow chart and the algorithms which describe the encryption and the decryption processes. In section 3, we have discussed an illustration of the cipher and studied the avalanche effect. In section 4, we have performed the cryptanalysis. Finally in section 5, we have spelt out the details of the computation carried out in this analysis and drawn conclusions.

2. Development of the cipher

Let us consider a plaintext P containing $2m^2$ characters. On using EBCIDIC code, P can be brought into the form of a pair of square matrices called P_0 and Q_0. The size of each matrix is m. We take a pair of square matrices called K and L as key matrices. Let the elements of K and L be in the interval [0-255].

The encryption and the decryption of this cipher are governed by the relations

$$\left.\begin{aligned} &\mathbf{P_i = Q_{i-1}},\\ &\mathbf{Q_i = (\ P_{i-1} + (K\,Q_{i-1}L)\,)\ mod\ N}, \end{aligned}\right\}\ \mathbf{i = 1\ to\ n} \qquad \mathbf{(2.1)}$$

and

$$\left.\begin{aligned} &\mathbf{Q_{i-1} = P_i},\\ &\mathbf{P_{i-1} = (\ Q_i\ -\ (K\,P_i\,L)\ )\ mod\ N}, \end{aligned}\right\}\ \mathbf{i = n\ to\ 1} \qquad \mathbf{(2.2)}$$

where, P_i and Q_i are the portions of P in i th round of the iteration process

Here the number of rounds in the iteration process is denoted by n, and we take n=16. N is taken as 256 as we have employed the EBCIDIC code in the development of the cipher.

In this analysis, the equations of encryption [2.1] are supplemented with a function called Shuffle ( ). This is used for mixing and permuting the binary bits of the plaintext in each round of the iteration process.

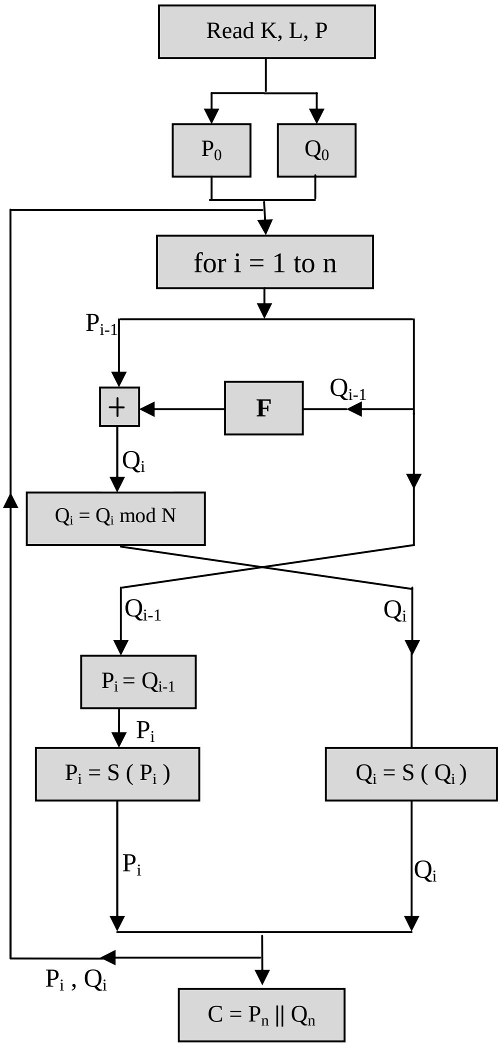

Fig 1. The process of Encryption

In the above flow chart, the function F includes the keys K and L as left and right multiplicants respectively

Correspondingly, we have associated the function IShuffle ( ) with the equations describing the decryption process (2.2). This function contains the reverse process of the function Shuffle ( ). The basic ideas of the Shuffling process are mentioned a little later.

In what follows, we present the flowcharts and the algorithms describing the process of encryption and the process of decryption.

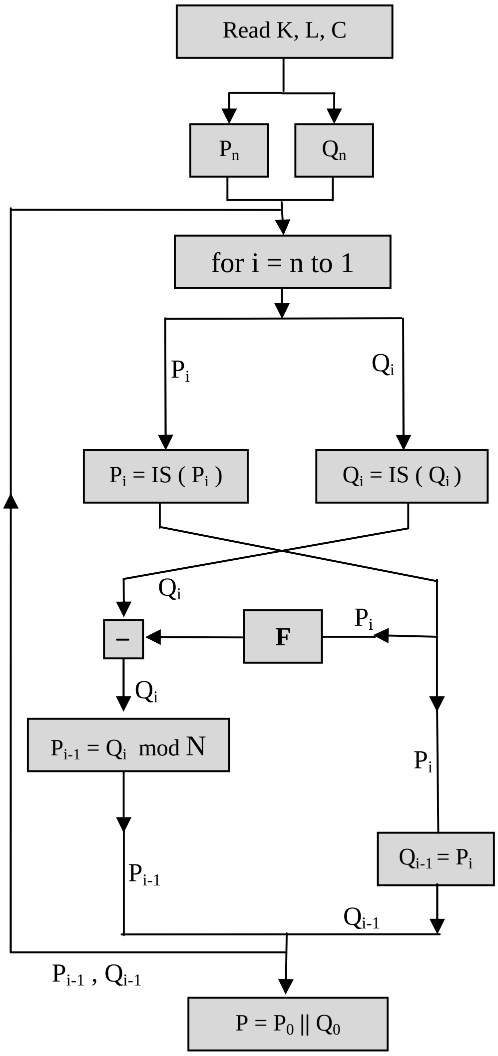

Fig 2. The process of Decryption

Algorithm for Encryption

1. Read P, K, n and N.
2. P_0 = Left half of P.
   Q_0 = Right half of P.
3. for i = 1 to n
begin
  $P_i = Q_{i-1}$
  $F = (K\ Q_{i-1}\ K)$
  $Q_i = (P_{i-1} + F) \bmod N$
  $P_i = S (P_i)$
  $Q_i = S (Q_i)$
end
4. $C = P_n \parallel Q_n$ /* $\parallel$ represents concatenation */
5. Write(C)

Algorithm for Decryption

1. Read C, K, n and N.
2. P_n = Left half of C
   Q_n = Right half of C
3. for i = n to 1
begin
  $P_i = IS (P_i)$
  $Q_i = IS (Q_i)$
  $Q_{i-1} = P_i$
  $F = (K\ P_i\ K) \bmod N$
  $P_{i-1} = (Q_i - F) \bmod N$
end
4. $P = P_0 \parallel Q_0$ /* $\parallel$ represents concatenation */
5. Write (P)

For the sake of elegance, in the flowcharts and the algorithms, we have represented the function Shuffle ( ) as S ( ), and the function IShuffle ( ) as IS ( ). The development of the function Shuffle can be described as follows

Consider a pair of square matrices U and V which are of size m. Let us assume that m is an even number, and m = 2r. Then the matrices U and V can be written as shown below.

$$U = \begin{bmatrix} U_{11} & U_{12} & \cdots & U_{1r} & U_{1(r+1)} & \cdots & U_{1m} \\ U_{21} & U_{22} & \cdots & U_{2r} & U_{2(r+1)} & \cdots & U_{2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ U_{m1} & U_{m2} & \cdots & U_{mr} & U_{m(r+1)} & \cdots & U_{mm} \end{bmatrix} \quad (2.3)$$

$$V = \begin{bmatrix} V_{11} & V_{12} & \cdots & V_{1r} & V_{1(r+1)} & \cdots & V_{1m} \\ V_{21} & V_{22} & \cdots & V_{2r} & V_{2(r+1)} & \cdots & V_{2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ V_{m1} & V_{m2} & \cdots & V_{mr} & V_{m(r+1)} & \cdots & V_{mm} \end{bmatrix} \quad (2.4)$$

Here, in both the matrices U and V, the first half and the second half are containing the columns 1 to r, and (r+1) to 2r (=m) respectively.

Let us swap the right half of U with the right half of V (that is from (r+1) th column to m th column). Thus we get

$$U = \begin{bmatrix} U_{11} & U_{12} & \cdots & U_{1r} & V_{1(r+1)} & \cdots & V_{1m} \\ U_{21} & U_{22} & \cdots & U_{2r} & V_{2(r+1)} & \cdots & V_{2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ U_{m1} & U_{m2} & \cdots & U_{mr} & V_{m(r+1)} & & V_{mm} \end{bmatrix} \quad (2.5)$$

$$V = \begin{bmatrix} V_{11} & V_{12} & \cdots & V_{1r} & U_{1(r+1)} & \cdots & U_{1m} \\ V_{21} & V_{22} & \cdots & V_{2r} & U_{2(r+1)} & \cdots & U_{2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ V_{m1} & V_{m2} & \cdots & V_{mr} & U_{m(r+1)} & \cdots & U_{mm} \end{bmatrix} \quad (2.6)$$

Now we introduce the subtle details of the shuffling process. In this, we have interposed the (r+1) the column of the above matrix in between first and second columns and the (r +2) th column in between third and fourth columns etc. , till we exhaust all the columns. Thus we rewrite the matrices U and V and get them in the form

$$U=\begin{bmatrix} U_{11} & V_{1(r+1)} & U_{12} & V_{1(r+2)} & \ldots\ldots & U_{1(r-1)} & V_{1(m-1)} & U_{1r} & V_{1m} \\ U_{21} & V_{2(r+1)} & U_{22} & V_{2(r+2)} & \ldots\ldots & U_{2(r-1)} & V_{2(m-1)} & U_{2r} & V_{1m} \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ U_{m1} & V_{m(r+1)} & U_{m2} & V_{m(r+2)} & \ldots\ldots & U_{m(r-1)} & V_{m(m-1)} & U_{mr} & V_{mm} \end{bmatrix} \quad (2.7)$$

and

$$V=\begin{bmatrix} V_{11} & U_{1(r+1)} & V_{12} & U_{1(r+2)} & \ldots\ldots & V_{1(r-1)} & U_{1(m-1)} & V_{1r} & U_{1m} \\ V_{21} & U_{2(r+1)} & V_{22} & U_{2(r+2)} & \ldots\ldots & V_{2(r-1)} & U_{2(m-1)} & V_{2r} & U_{1m} \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ V_{m1} & U_{m(r+1)} & V_{m2} & U_{m(r+2)} & \ldots\ldots & V_{m(r-1)} & U_{m(m-1)} & V_{mr} & U_{mm} \end{bmatrix} \quad (2.8)$$

For the sake of convenience, we have denoted (2.7) and (2.8) again as U and V only. Now we take

$U = [U_{ij}]$ i = 1 to m, j = 1 to m,

and

$V = [V_{ij}]$ i = 1 to m, j = 1 to m.

On writing each element of U and V in its binary form, we get

$$U = \begin{bmatrix} U_{111}\ U_{112}\ldots U_{118} & U_{121}\ U_{122}\ldots U_{128}\ \ldots\ldots & U_{1r1}\ U_{1r2}\ldots U_{1r8} & U_{1(r+1)1}\ U_{1(r+1)2}\ldots U_{1(r+1)8}\ldots\ldots & U_{1m1}\ U_{1m2}\ldots U_{1m8} \\ U_{211}\ U_{212}\ldots U_{218} & U_{221}\ U_{222}\ldots U_{228}\ \ldots\ldots & U_{2r1}\ U_{2r2}\ldots U_{2r8} & U_{2(r+1)1}\ U_{2(r+1)2}\ldots U_{2(r+1)8}\ldots\ldots & U_{2m1}\ U_{2m2}\ldots U_{2m8} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ U_{m11}\ U_{m12}\ldots U_{m18} & U_{m21}U_{m22}\ldots U_{m28}\ldots\ldots & U_{mr1}U_{mr2}\ldots U_{mr8} & \cdot\ {}_{+1)1}\ U_{m(r+1)2}\ldots U_{m(r+1)8}\ldots\ldots & \cdot\ U_{mm2}\ldots U_{mm8} \end{bmatrix} \quad (2.9)$$

and

$$V = \begin{bmatrix} V_{111}\ V_{112}\ldots V_{118} & V_{121}\ V_{122}\ldots V_{128}\ \ldots\ldots & V_{1r1}\ V_{1r2}\ldots V_{1r8} & V_{1(r+1)1}\ V_{1(r+1)2}\ldots V_{1(r+1)8}\ldots\ldots & V_{1m1}\ V_{1m2}\ldots V_{1m8} \\ V_{211}\ V_{212}\ldots V_{218} & V_{221}\ V_{222}\ldots V_{228}\ \ldots\ldots & V_{2r1}\ V_{2r2}\ldots V_{2r8} & V_{2(r+1)1}\ V_{2(r+1)2}\ldots V_{2(r+1)8}\ldots\ldots & V_{2m1}\ V_{2m2}\ldots V_{2m8} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ V_{m11}\ V_{m12}\ldots V_{m18} & V_{m21}V_{m22}\ldots V_{m28}\ldots\ldots & V_{mr1}V_{mr2}\ldots V_{mr8} & V_{m(r+1)1}\ V_{m(r+1)2}\ldots V_{m(r+1)8}\ldots\ldots & V_{mm1}V_{mm2}\ldots V_{mm8} \end{bmatrix} \quad (2.10)$$

Now we interchange the right half of U with the right half of V, and then shuffle the columns of the binary bits (as we have done earlier in (2.7) and (2.8)). Thus we get a pair of new matrices, which are once again called as U and V. On writing each 8 binary bits as a decimal number (of course taking the binary bits in a row wise manner), we obtain a pair of square matrices of size m, we call them as U and V. This completes the process involved in the function Shuffle ( ). It is readily possible to visualize the reverse process of Shuffling and to develop the function IShuffle ( ) very easily.

3. Illustration of cipher

Let us consider the following plaintext
Dear brother, I am very glad to know that you have joined as sub-inspector of police. When I was doing my B.Tech final year, you were in the training period. In that year, I did not get my scholarship as there were several political changes in our state. I tried to take bank loan but I was not successful. When I was thinking what I must do at that time, some bazaar rowdy came to my rescue and helped me by giving some money. When I was about to complete my B.Tech, he told me that the money was given by a naxalite, and I am to join

with them. Otherwise I will be murdered! So I joined them! (3.1)

Let us focus our attention on the first 128 Characters of the plaintext (3.1). This is given by

Dear brother, I am very glad to know that you have joined as sub-inspector of police. When I was doing my B.Tech final year, you (3.2)

On using the EBCIDIC code on (3.2), we get

$$P = \begin{bmatrix} 68 & 101 & 97 & 114 & 32 & 98 & 114 & 111 & 116 & 104 & 101 & 114 & 44 & 32 & 73 & 32 \\ 97 & 109 & 32 & 118 & 101 & 114 & 121 & 32 & 103 & 108 & 97 & 100 & 32 & 116 & 111 & 32 \\ 107 & 110 & 111 & 119 & 32 & 116 & 104 & 97 & 116 & 32 & 121 & 111 & 117 & 32 & 104 & 97 \\ 118 & 101 & 32 & 106 & 111 & 105 & 110 & 101 & 100 & 32 & 97 & 115 & 32 & 115 & 117 & 98 \\ 45 & 105 & 110 & 115 & 112 & 101 & 99 & 116 & 111 & 114 & 32 & 111 & 102 & 32 & 112 & 111 \\ 108 & 105 & 99 & 101 & 46 & 32 & 87 & 104 & 101 & 110 & 32 & 73 & 32 & 119 & 97 & 115 \\ 32 & 100 & 111 & 105 & 110 & 103 & 32 & 109 & 121 & 32 & 66 & 46 & 84 & 101 & 99 & 104 \\ 32 & 102 & 105 & 110 & 97 & 108 & 32 & 121 & 101 & 97 & 114 & 44 & 32 & 121 & 111 & 117 \end{bmatrix} \quad (3.3)$$

Thus we have

$$P_0 = \begin{bmatrix} 68 & 101 & 97 & 114 & 32 & 98 & 114 & 111 \\ 97 & 109 & 32 & 118 & 101 & 114 & 121 & 32 \\ 107 & 110 & 111 & 119 & 32 & 116 & 104 & 97 \\ 118 & 101 & 32 & 106 & 111 & 105 & 110 & 101 \\ 45 & 105 & 110 & 115 & 112 & 101 & 99 & 116 \\ 108 & 105 & 99 & 101 & 46 & 32 & 87 & 104 \\ 32 & 100 & 111 & 105 & 110 & 103 & 32 & 109 \\ 32 & 102 & 105 & 110 & 97 & 108 & 32 & 121 \end{bmatrix} \quad (3.4)$$

and

$$Q_0 = \begin{bmatrix} 116 & 104 & 101 & 114 & 44 & 32 & 73 & 32 \\ 103 & 108 & 97 & 100 & 32 & 116 & 111 & 32 \\ 116 & 32 & 121 & 111 & 117 & 32 & 104 & 97 \\ 100 & 32 & 97 & 115 & 32 & 115 & 117 & 98 \\ 111 & 114 & 32 & 111 & 102 & 32 & 112 & 111 \\ 101 & 110 & 32 & 73 & 32 & 119 & 97 & 115 \\ 121 & 32 & 66 & 46 & 84 & 101 & 99 & 104 \\ 101 & 97 & 114 & 44 & 32 & 121 & 111 & 117 \end{bmatrix} \quad (3.5)$$

Let us take the Keyes K and L in the form

$$K = \begin{bmatrix} 200 & 212 & 16 & 220 & 34 & 117 & 12 & 132 \\ 54 & 13 & 125 & 226 & 33 & 120 & 236 & 100 \\ 130 & 68 & 154 & 13 & 83 & 137 & 54 & 72 \\ 100 & 39 & 133 & 20 & 112 & 140 & 121 & 116 \\ 115 & 113 & 82 & 75 & 16 & 215 & 111 & 124 \\ 233 & 217 & 228 & 39 & 190 & 111 & 19 & 149 \\ 136 & 158 & 49 & 137 & 100 & 102 & 115 & 116 \\ 144 & 12 & 38 & 197 & 125 & 135 & 145 & 205 \end{bmatrix} \quad (3.6)$$

and

$$L = \begin{bmatrix} 133 & 192 & 66 & 123 & 36 & 155 & 53 & 164 \\ 05 & 09 & 129 & 155 & 56 & 187 & 116 & 205 \\ 215 & 270 & 150 & 15 & 42 & 116 & 94 & 82 \\ 105 & 249 & 203 & 12 & 222 & 175 & 204 & 218 \\ 168 & 219 & 10 & 66 & 88 & 210 & 174 & 136 \\ 154 & 182 & 176 & 50 & 113 & 116 & 02 & 39 \\ 126 & 144 & 82 & 215 & 108 & 118 & 194 & 146 \\ 15 & 72 & 54 & 130 & 112 & 146 & 195 & 110 \end{bmatrix} \quad (3.7)$$

On using P_0, Q_0, K and L , given by (3.4) to (3.7), and adopting the encryption algorithm given in section 2, we get the cipher text C in the form

$$C = \begin{bmatrix} 47 & 36 & 206 & 218 & 60 & 59 & 123 & 231 & 136 & 21 & 102 & 153 & 08 & 73 & 110 & 244 \\ 73 & 133 & 152 & 198 & 214 & 246 & 181 & 216 & 219 & 86 & 197 & 165 & 70 & 115 & 201 & 31 \\ 95 & 27 & 149 & 155 & 233 & 115 & 150 & 255 & 233 & 44 & 85 & 154 & 100 & 29 & 189 & 243 \\ 96 & 05 & 152 & 137 & 225 & 237 & 35 & 158 & 142 & 228 & 195 & 135 & 76 & 243 & 01 & 238 \\ 233 & 223 & 102 & 67 & 156 & 183 & 123 & 146 & 131 & 183 & 190 & 72 & 128 & 179 & 00 & 05 \\ 205 & 185 & 126 & 90 & 88 & 195 & 182 & 149 & 176 & 26 & 183 & 212 & 219 & 50 & 69 & 189 \\ 106 & 233 & 188 & 190 & 71 & 35 & 180 & 237 & 243 & 247 & 198 & 73 & 199 & 225 & 125 & 217 \\ 04 & 218 & 198 & 221 & 31 & 99 & 91 & 29 & 251 & 152 & 197 & 93 & 37 & 36 & 141 & 183 \end{bmatrix} . \quad (3.8)$$

On making use of the cipher text C given by (3.8), the keys K and L, given by (3.6) and (3.7), and the decryption algorithm, we get back the original plaintext (3.3).

Now let us examine the avalanche effect, which gives a qualitative picture about the strength of the cipher. To achieve this one, let us change the first character of (3.2) from D to E. As the EBCIDIC codes of these two characters are 68 and 69, we have a change of one binary bit in the plaintext.
Now on using the keys K and L, given by (3.6) and (3.7), the encryption algorithm and the modified plaintext(according to the change made), we get the cipher text C corresponding to the plaintext under consideration. This is given by

$$C = \begin{bmatrix} 70 & 219 & 194 & 242 & 76 & 237 & 163 & 193 & 37 & 187 & 209 & 38 & 42 & 205 & 50 & 14 \\ 222 & 249 & 226 & 02 & 204 & 99 & 107 & 123 & 90 & 236 & 109 & 171 & 98 & 210 & 163 & 57 \\ 228 & 143 & 175 & 141 & 202 & 205 & 244 & 185 & 203 & 127 & 244 & 150 & 42 & 205 & 50 & 14 \\ 222 & 249 & 226 & 02 & 204 & 68 & 240 & 246 & 145 & 207 & 71 & 114 & 97 & 195 & 166 & 121 \\ 128 & 247 & 116 & 239 & 179 & 33 & 206 & 91 & 189 & 201 & 65 & 219 & 223 & 36 & 64 & 89 \\ 128 & 02 & 230 & 220 & 191 & 45 & 44 & 97 & 219 & 74 & 216 & 13 & 91 & 234 & 109 & 153 \\ 34 & 222 & 181 & 116 & 222 & 95 & 35 & 145 & 218 & 118 & 249 & 251 & 227 & 36 & 227 & 240 \\ 190 & 236 & 130 & 109 & 99 & 110 & 143 & 177 & 173 & 142 & 253 & 204 & 98 & 174 & 146 & 146 \end{bmatrix} \quad (3.9)$$

On comparing (3.8) and (3.9), in their binary form, we find that, these cipher texts differ by 508 bits (out of 1024 bits).

Now let us study the effect of one bit change in the keys. To this end, we change the seventh row, first column element of the key matrix K from 136 to 137 As these two numbers differ by one binary bit, the key changes in one bit. On using the original plaintext (3.2), the modified key K, the other key L(with out any change) and the encryption algorithm, we obtain

$$C = \begin{bmatrix} 182 & 108 & 50 & 76 & 228 & 143 & 108 & 194 & 82 & 71 & 102 & 45 & 35 & 114 & 42 & 205 \\ 136 & 59 & 104 & 240 & 46 & 91 & 111 & 139 & 182 & 196 & 145 & 144 & 118 & 247 & 206 & 246 \\ 183 & 231 & 51 & 76 & 131 & 162 & 190 & 193 & 13 & 118 & 54 & 243 & 150 & 255 & 160 & 118 \\ 222 & 183 & 253 & 242 & 134 & 155 & 217 & 219 & 57 & 228 & 143 & 175 & 234 & 217 & 190 & 149 \\ 11 & 49 & 141 & 164 & 151 & 169 & 03 & 76 & 128 & 195 & 188 & 119 & 38 & 28 & 44 & 06 \\ 207 & 17 & 23 & 230 & 197 & 93 & 29 & 205 & 190 & 30 & 219 & 124 & 244 & 202 & 186 & 103 \\ 159 & 174 & 73 & 254 & 88 & 164 & 214 & 32 & 30 & 239 & 150 & 239 & 105 & 115 & 59 & 236 \\ 242 & 254 & 30 & 225 & 123 & 169 & 182 & 107 & 236 & 237 & 147 & 244 & 150 & 46 & 23 & 45 \end{bmatrix} \qquad (3.10)$$

After converting (3.8) and (3.10) into their binary form and comparing them, we notice that they differ by 516 bits (out of 1024 bits). This result also firmly indicates that the cipher is expected to be a potential one.

4. Cryptanalysis

In the development of a block cipher, cryptanalysis plays a vital role in deciding whether the cipher is a strong one or not. The methods that are used in cryptanalysis are

1. Cipher text only (Brute Force) attack.
2. Known Plaintext attack.
3. Chosen Plaintext attack.
4. Chosen Cipher text attack.

The first two methods are generally used in the literature, while the latter two methods are rarely utilized in deciding the strength of a cipher. As William Stallings [2] has pointed out that every cipher is to be developed so that it withstands the first two attacks.

Let us now consider the cipher text only attack. In this analysis we have two keys K and L, wherein each one is of size mxm. Thus, the total number of elements in both the keys put together is $2m^2$. Hence the size of the key space is

$$2^{16m^2} = \left(10^{3}\right)^{1.6m^2} = 10^{4.8m^2}$$

If we assume that, the time required for the encryption with one value of the key in the key space is $(10)^{-7}$ seconds, then the time needed for the execution of the encryption with all the possible keys in the key space is

$$10^{4.8m^2} \times 10^{-7} \text{ Seconds}$$

$$= \frac{10^{4.8m^2} \times 10^{-7}}{365 \times 24 \times 60 \times 60} \text{ Years}$$

$$= 10^{4.8m^2} \times 3.12 \times 10^{-15} \text{ Years}$$

$$= 3.12 \times 10^{(4.8m2 - 15)} \text{ Years}$$

Here in the present analysis, we have m = 8. Hence the time required is equal to

$$= 3.12 \times 10^{292.2} \text{ Years}$$

As this is enormously large, this cipher cannot be broken by the brute force attack.

Now let us examine the known plaintext attack, in this case, we know as many pairs of plaintext and cipher text as we require for the purpose of our analysis. In each round of the iteration process, as a part of the plaintext is multiplied with the key, on both the sides, and a through shuffling is carried out, after changing the sides, the relation between the cipher text and the plaintext at the end of the iteration process does not give any scope to find the key or a function of the key. Hence, the cipher cannot be broken by the known plaintext attack.

In the light of the above discussion we conclude that this cipher is considerably a strong one.

5. Computations and Conclusions

In the present investigation, we have developed a block cipher by modifying the Feistel cipher in which we have included modular arithmetic addition as a fundamental operation. In each round of the iteration process, we have shuffled the binary bits of the plaintext, operated with the key, in a thorough manner. The programs required in this analysis are written in C language.

The plaintext (3.1) is divided into five blocks, wherein each one is containing 128 characters. As the last block is containing 83 characters, we have added 45 blanks to make it a complete block of length 128. On using the keys K and L, given by (3.6) and (3.7), and the encryption algorithm presented in section 2, we get the cipher text C, corresponding to the entire plaintext (3.1), in the form

47	36	206	218	60	59	123	231
136	21	102	153	08	73	110	244
73	133	152	198	214	246	181	216
219	86	197	165	70	115	201	31
95	27	149	155	233	115	150	255
233	44	85	154	100	29	189	243
196	05	152	137	225	237	35	158
142	228	195	135	76	243	01	238
233	223	102	67	156	183	123	146
131	183	190	72	128	179	00	05
205	185	126	90	88	195	182	149
176	26	183	212	219	50	69	189
106	233	188	190	71	35	180	237
243	247	198	73	199	225	125	217
04	218	198	221	31	99	91	29
251	152	197	93	37	36	141	183
36	206	182	125	163	193	37	187
248	92	181	18	98	172	211	32
229	152	198	214	246	181	216	219
86	197	165	70	115	201	31	95
27	149	155	233	115	150	255	233
44	85	154	100	29	189	243	196
5	152	137	225	237	35	158	142
228	195	135	76	243	1	238	233
223	102	67	156	183	123	146	131
183	190	72	128	179	0	5	205
185	126	90	88	195	182	149	176
26	183	212	219	50	69	189	106
233	188	190	71	35	180	237	243
247	198	73	199	225	125	217	4
218	198	221	31	99	91	29	251
152	197	93	37	37	108	216	100
136	219	120	95	156	145	237	152
89	139	72	220	138	179	98	14
218	60	11	150	219	226	237	177
36	100	29	189	243	189	173	249
204	211	32	232	175	176	67	93
141	188	229	191	232	29	183	173
255	124	161	166	246	118	206	121
35	235	250	182	111	165	66	204
99	105	37	234	64	211	32	48
239	29	201	135	11	1	179	196
69	249	177	87	71	115	111	135
182	223	61	50	174	153	231	235
146	127	150	41	53	136	7	187
229	187	218	92	206	251	60	191
135	184	94	234	109	154	251	59
100	253	37	139	133	203	115	180
108	253	242	147	98	70	25	38
114	146	57	35	209	179	44	238
137	38	133	187	218	88	128	123
166	217	146	196	189	183	143	57
229	243	114	103	82	190	153	3
83	24	199	54	42	233	240	108
119	54	114	88	240	196	67	156
22	72	236	29	190	100	44	57
41	43	69	185	109	189	49	164
240	246	122	220	157	188	7	106
88	201	219	71	157	182	54	189
252	223	0	67	249	202	140	172
153	144	26	141	203	45	91	155
195	189	185	228	143	166	178	57
79	100	249	200	173	142	25	102
63	155	145	117	58	54	60	253
25	132	161	219	222	229	148	145
217	139	72	220	138	179	107	157
55	190	120	129	91	158	196	29
180	120	23	45	183	197	219	98
72	200	59	123	231	123	91	243
153	166	65	209	95	96	134	187
27	121	203	127	208	59	111	91
254	249	67	77	236	237	156	242
71	215	245	108	223	74	133	152
198	210	75	212	129	166	64	97
222	59	147	14	22	3	103	136
139	243	98	174	142	230	223	15
109	190	122	101	93	51	207	215
36	255	44	82	107	16	15	119
203	119	180	185	157	246	121	127
15	112	189	212	219	53	241	44

In the light of the discussion presented in the cryptanalysis, we have seen that the cipher is a strong one. Here it is to be noted that the strength of the cipher is achieved by the multiplication with the key and the shuffling carried out in each round of the iteration process. This cipher is quite comparable with the cipher discussed in [3,4].

5. References

[1] V. U. K. Sastry, K. Anup Kumar, "A Modified Feistel Cipher involving a pair of key matrices, supplemented with XOR operation, and Blending of the plaintext in each round of the iteration"

[2] William Stallings, Cryptography and Network Security, Principles and Practice, Third Edition, Pearson, 2003.

[3] V. U. K. Sastry, K. Anup Kumar, "A Modified Feistel Cipher involving a key as a multiplicant on both the sides of the plaintext matrix and supplemented with Mixing, Permutation and XOR Operation", IJCTA

[4] V. U. K. Sastry, K. Anup Kumar, "A Modified Feistel Cipher involving a key as a multiplicant on both the sides of the plaintext matrix and supplemented with Mixing, Permutation and Modular Arithmetic Addition", IJCTA

Authors profile:

Dr. V. U. K. Sastry is presently working as Professor in the Dept. of Computer Science and Engineering (CSE), Director (SCSI), Dean (R & D), SreeNidhi Institute of Science and Technology (SNIST), Hyderabad, India. He was Formerly Professor in IIT, Kharagpur, India and Worked in IIT, Kharagpur during 1963 – 1998. He guided 12 PhDs, and published more than 40 research papers in various international journals. His research interests are Network Security & Cryptography, Image Processing, Data Mining and Genetic Algorithms.

Mr. K. Anup Kumar is presently working as an Associate Professor in the Department of Computer Science and Engineering, SNIST, Hyderabad India. He obtained his B.Tech (CSE) degree from JNTU Hyderabad and his M.Tech (CSE) from Osmania university, Hyderabad. He is now pursuing his PhD from JNTU, Hyderabad, India, under the supervision of Dr. V.U.K. Sastry in the area of Information Security and Cryptography. He has 10 years of teaching experience and his interest in research area includes, Cryptography, Steganography and Parallel Processing Systems.