ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

Improved Cryptographic Technique by Square Matrix with Column Cells and Uniform Point Crossover on Binary Field

Ishrath Jahan[1], Udayini Chandana[2]

Department of Electronics and Communication Engineering, Stanley College of Engineering and Technology for Women, Hyderabad, India [1, 2]

Abstract: A new cryptographic algorithm is introduced. This technique uses three keys for encryption and decryption. A nearby square matrix with few column cells is used to place the input plain text in a unique manner. The left diagonal's positional value will be the key-1 with that key intermediate cipher text is produced. A 7-digit random number is generated as key-2. According to the digits of key-2 the section division, the block division process and the crossover point is finalized. Uniform point crossover is applied on the binary field of intermediate cipher text to produce complex final cipher text.

Keywords: Plain-text, Key, Intermediate Cipher-text, Cipher-text, Encryption, Decryption, Crossover.

I. INTRODUCTION

Cryptography is an indispensable tool for protecting information in computer systems. Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication. It is the technique by which the data is encrypted at the transmitter side and is decrypted again at the receiver side keeping the data secure from eavesdropper. It is method where the data is sent in the disguised form only to the intended recipients who can decrypt it and read the message. The encryption and decryption of the plain text (data) is done with the help of keys [1]. The keys are generated by the user or can be calculated or randomly generated depending upon the algorithm used. The data that is transmitted can be sent bit by bit or in blocks to the recipient. The block sizes and the key length are variable and can be fixed by the user at the beginning of the ciphering. Ciphering is the process of conversion of data into disguised form. Cryptographic technique using substitution followed by genetic function has been carried out in different ways [2]. Many genetic algorithm based encryption have been proposed describe a new symmetrical block ciphering system named ICIGA (Improved Cryptography Inspired by Genetic Algorithms) which generates a session key in a random process [3], [4]. ICIGA is an enhancement of the system (GIC) "Genetic algorithms Inspired Cryptography" [5]. Generation of block cipher and stream cipher [6].The Crossover operator has the significance in genetic function algorithms. Two parent blocks are considered and a crossover point is finalized according to the type of the crossover operation being performed. The crossover operation is performed to get the modified child blocks. In GAs different types of crossover are available [7], [8], [9]. A new algorithm for encryption and decryption is introduced. The algorithm is based on the process of substitution and genetic function. In this proposed model the number of letters of input plain text is placed into a cells in a specific manner, and key-1 is calculated. By using this key, the plain text is transformed into intermediate cipher text. A key-2 along with genetic function is used to obtain the final cipher text. By doing the inverse operation the plain text is obtained from the final cipher text.

II. THE PROPOSED TECHNIQUE

In the proposed technique, each letter of the input plain text is placed into a matrix of cells according to the number of letters in the input plain text. If the number of letters in the input plain text is a square number then they are placed in a square matrix. If the number of letter of input plain text is not a square number then a nearby least square matrix with column cells should be selected to place the plain text. The arrangement of letters of input plain text into a square matrix with column cells is shown in Figure-1.

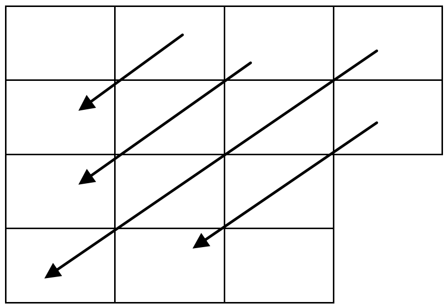

Figure-1. Pattern for arranging plain text in matrix of cells

The three keys are used in this technique. The key-1 is generated by adding positional value (A=1, B=2,……Z=26) of letters placed in left diagonal position of the matrix of cells. Each of the positional value of letters in the matrix is then added with Key-1 to generate intermediate cipher text. The intermediate cipher text will be the new string of characters different from the plain text. The intermediate cipher text will be converted into a binary code of 64-bits of ASCII value. A 7-digit random

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

number is generated as key-2. The first digit of the key-2 is the section number by which all the bits are divided into sections. The each section will be divided into blocks according to the last 6 digits of random number. Genetic function uniform point crossover is performed on blocks of bits of each section. If the total block number of a section is even, each block crossed over with the next block to produce level-1 modified blocks otherwise first block, N block (where N is odd) and second block, fourth block is crossed over and so on to produce Level 1 modified blocks. The number of blocks in each section will be key-3 of the proposed technique.

III. FLOW CHART REPRESENTATION OF UNIFORM POINT CROSSOVER GENETIC FUNCTION

The block diagram of crossover when the number of blocks in each section is even is shown in Figure – 2.
Each block is denoted by a number.
N=even number.

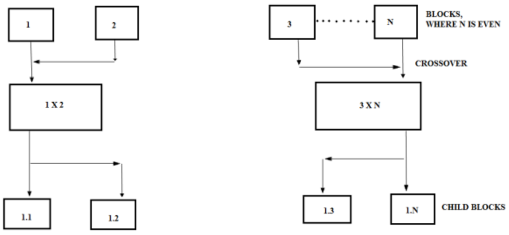

Figure – 2. Representation of crossover if the number of blocks is even

The block diagram of crossover when the number of blocks in each section is odd is shown in Figure – 3. Each block is denoted by a number. N=odd number.

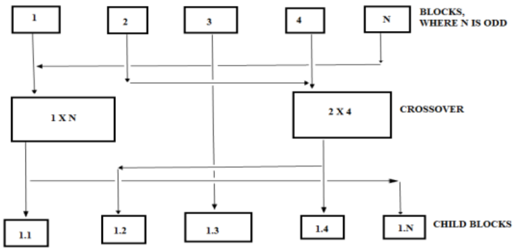

Figure – 3. Representation of crossover if the number of blocks is odd

IV. EXAMPLE

A. Encryption

Let the Plain text is: SPAINRUSSIAINDIAGERMANYSINGAPORE. Size of the plain text is 32. A nearby least square matrix to 32 is 5*5 is selected along with 7 column cells to place the plain text in the cells which is shown in the Figure-4.

S	P	I	U	A	A	A
A	N	S	I	G	N	N
R	S	N	E	Y	G	
I	D	R	S	A	O	
I	M	I	P	R	E	

Figure – 4. Arrangement of 32 characters of plain text into of cells

Key 1= Position Value of 'S' + Position Value of 'N' + Position Value of 'N' + Position Value of 'S'+ Position Value of 'R' =19+14+14+19+18=84 [Where, A=1…………….z=26]

The Key 1 will be added with each letter's position value to generate the intermediate cipher text.
S = 19 + 84 = 103 = **Y**, P = 16 + 84 = 100 = **V**, A = 1 + 84 = 85 = **G**, I= 9 + 84 = 93 = **O**, N = 14 + 84 = 98 = **T**, R = 18 + 84 = 102 = **X**, U = 21 + 84 = 105 = **A**, S = 19 + 84 = 103 = **Y**, S = 19 + 84 = 103 = **Y**, I = 9 + 84 = 93 = **O**, A = 1 + 84 = 85 = **G**, I = 9 + 84 = 93 = **O**, N = 14 + 84 = 98 = **T**, D = 4 + 84 = 88 = **J**, I = 9 + 84 = 93 = **O**, A = 1 + 84 = 85 = **G**, G = 7 + 84 = 91 = **M**, E = 5 + 84 = 89 = **K**, R = 18 + 84 = 102 = **X**, M = 13 + 84 = 97 = **S**, A = 1 + 84 = 85 = **G**, N = 14 + 84 = 98 = **T**, Y = 25 + 84 = 109 = **E**, S = 19 + 84 = 103 = **Y**, I = 9 + 84=93= **O**, N = 14 + 84 = 98 = **T**, G = 7 + 84 = 91 = **M**, A = 1 + 84 = 85 = **G**, P = 16 + 84 = 100 = **V**, O = 15 + 84 = 99 = **U**, R = 18 + 84 = 102 = **X**, E = 5 + 84 = 89 = **K**

Intermediate cipher text:
YVOAGGGGTYOMTTXYTKEMOJXYGUOSOVXK
Intermediate cipher text in cells is shown in Figure-5.

Y	V	O	A	G	G	G
G	T	Y	O	M	T	T
X	Y	T	K	E	M	
O	J	X	Y	G	U	
O	S	O	V	X	K	

Figure-5. Arrangement Intermediate cipher text in cells

Each letter of the matrix is represented in 64 bits Binary code of ASCII value. So that, total bits number of intermediate cipher text will be: 32* 64 bits =2048 bits as shown Table-1.

TABLE I
REPRESENTATION OF INTERMEDIATE CIPHER IN 64 BITS BINARY CODE OF ASCII VALUE

Character	Binary Code(64-bits) of ASCII value
Y	00000000000000000000000000000000 00000000000000000000000001011001
V	00000000000000000000000000000000 00000000000000000000000001010110
O	00000000000000000000000000000000 00000000000000000000000001001111
A	00000000000000000000000000000000 00000000000000000000000001000001
G	00000000000000000000000000000000 00000000000000000000000001000111
G	00000000000000000000000000000000 00000000000000000000000001000111
G	00000000000000000000000000000000 00000000000000000000000001000111

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

G	000000000000000000000000000000000 0000000000000000000000001000111
T	000000000000000000000000000000000 0000000000000000000000001010100
Y	000000000000000000000000000000000 0000000000000000000000001011001
O	000000000000000000000000000000000 0000000000000000000000001001111
M	000000000000000000000000000000000 0000000000000000000000001001101
T	000000000000000000000000000000000 0000000000000000000000001010100
T	000000000000000000000000000000000 00000000000000001010100
X	000000000000000000000000000000000 0000000000000000000000001011000
Y	000000000000000000000000000000000 0000000000000000000000001011001
T	000000000000000000000000000000000 0000000000000000000000001010100
K	000000000000000000000000000000000 0000000000000000000000001001011
E	000000000000000000000000000000000 0000000000000000000000001000101
M	000000000000000000000000000000000 0000000000000000000000001001101
O	000000000000000000000000000000000 0000000000000000000000001001111
J	000000000000000000000000000000000 0000000000000000000000001001010
X	000000000000000000000000000000000 0000000000000000000000001011000
Y	000000000000000000000000000000000 0000000000000000000000001011001
G	000000000000000000000000000000000 0000000000000000000000001000111
U	000000000000000000000000000000000 0000000000000000000000001010101
O	000000000000000000000000000000000 0000000000000000000000001001111
S	000000000000000000000000000000000 0000000000000000000000001010011
O	000000000000000000000000000000000 0000000000000000000000001001111
V	000000000000000000000000000000000 0000000000000000000000001010110
X	000000000000000000000000000000000 0000000000000000000000001011000
K	000000000000000000000000000000000 0000000000000000000000001001011

Binary representation of intermediate cipher text is:

00000000000000000000000000000000000000000000000
00000000001011001000000000000000000000000000000
00000000000000000000000000010101100000000000000
00000000000000000000000000000000000000000000100
11110000000000000000000000000000000000000000000
00000000000000100000100000000000000000000000000
00000000000000000000000000000001000111000000000
00000000000000000000000000000000000000000000000
01000111000000000000000000000000000000000000000
00000000000000000010001110000000000000000000000
00000000000000000000000000000000000100011100000
00000000000000000000000000000000000000000000000
00000101010000000000000000000000000000000000000
00000000000000000000001011001000000000000000000
00000000000000000000000000000000000000010011110
00000000000000000000000000000000000000000000000
00000000010011010000000000000000000000000000000
00000000000000000000000000101010000000000000000
00000000000000000000000000000000000000000001010
10000000000000000000000000000000000000000000000
00000000000001011000000000000000000000000000000
00000000000000000000000000000010110010000000000
00000000000000000000000000000000000000000000000
10101000000000000000000000000000000000000000000
00000000000000000100101100000000000000000000000
00000000000000000000000000000000001000101000000
00000000000000000000000000000000000000000000000
00001001101000000000000000000000000000000000000
00000000000000000000010011110000000000000000000
00000000000000000000000000000000000000100101000
00000000000000000000000000000000000000000000000
00000000101100000000000000000000000000000000000
00000000000000000000000001011001000000000000000
00000000000000000000000000000000000000000010001
11000000000000000000000000000000000000000000000
00000000000010101010000000000000000000000000000
00000000000000000000000000000100111100000000000
00000000000000000000000000000000000000000000001
01001100000000000000000000000000000000000000000
00000000000000001001111000000000000000000000000
00000000000000000000000000000000010101100000000
00000000000000000000000000000000000000000000000
00010110000000000000000000000000000000000000000
000000000000000000001001011

A 7-digit random number is generated as key-2. The key-2 generated is: 6543278. Depending upon the digits of key-2 the 2048 bits of intermediate cipher text will be divided into sections and blocks.

The first digit of key-2 is 6 i.e., they are divided into 6 sections and each section will be further divided into 5, 4, 3, 2, 7, 8 blocks respectively as 5, 4, 3, 2, 7, 8 are the next digits of key-2.

Section 1 is divided into 5 blocks.
Section 2 is divided into 4 blocks.
Section 3 is divided into 3 blocks.
Section 4 is divided into 2 blocks.
Section 5 is divided into 7 blocks.
Section 6 is divided into 8 blocks.

Each section contains (2048/6) = 341 bits
Discard the remainder (2048%6) = 2 bits for future use.

Last 2 bits '11' will be discarded from the intermediate cipher as the remainder is 2.

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

Section 1 (Each Block contains(341/5) bits or 68 bits with 1 remainder				
Block 1	Block 2	Block 3	Block 4	Block 5
00000000 00000000 00000000 00000000 00000000 00000000 00000000 01011001 0000	00000000 00000000 00000000 00000000 00000000 00000000 00000101 01100000 0000	00000000 00000000 00000000 00000000 00000000 00000000 01001111 00000000 0000	00000000 00000000 00000000 00000000 00000000 00000100 00010000 00000000 0000	00000000 00000000 00000000 00000000 00000000 01000111 00000000 00000000 00001

Section 2 (Each Block contains(341/4) bits or 85 bits with 1 remainder			
Block 1	Block 2	Block 3	Block 4
0000000000 0000000000 0000000000 0000001000 1110000000 0000000000 0000000000 0000000000 00000	0000000000 0000010001 1100000000 0000000000 0000000000 0000000000 0000000000 0000000001 00011	1000000000 0000000000 0000000000 0000000000 0000000000 0000000010 1010000000 0000000000 00000	0000000000 0000000000 0000000000 0000000101 1001000000 0000000000 0000000000 0000000000 000000

Section 3 (Each Block contains(341/3) bits or 113 bits with 2 remainder		
Block 1	Block 2	Block 3
000000000000000 100111100000000 000000000000000 000000000000000 000000000000000 000010011010000 000000000000000 00000000	000000000000000 000000000000000 101010000000000 000000000000000 000000000000000 000000000000000 000010101000000 00000000	000000000000000 000000000000000 000000000000000 101100000000000 000000000000000 000000000000000 000000000000000 000101100

Section 4 (Each Block contains(341/2) bits or 170 bits with 1 remainder	
Block 1	Block 2
1000000000000000000000000 0000000000000000000000000 0000000000101010000000000 0000000000000000000000000 0000000000000000000000000 0010010110000000000000000 0000000000000000000000000 00	0000000000000000001000101 0000000000000000000000000 0000000000000000000000000 0000000000010011010000000 0000000000000000000000000 0000000000000000000000000 0000001001111000000000000 0000000000

Section 5 (Each Block contains(341/7) bits or 48 bits with 5 remainder						
Block 1	Block 2	Block 3	Block 4	Block 5	Block 6	Block 7
00000 00000 00000 00000 00000 00000 00000 00100 10100 000	00000 00000 00000 00000 00000 00000 00000 00000 00000 000	00000 10110 00000 00000 00000 00000 00000 00000 00000 000	00000 00000 00000 00000 01011 00100 00000 00000 00000 000	00000 00000 00000 00000 00000 00000 00000 00100 01110 000	00000 00000 00000 00000 00000 00000 00000 00000 00000 000	00000 10101 01000 00000 00000 00000 00000 00000 00000 00000 000

Section 6 (Each Block contains(341/8) bits or 42 bits with 5 remainder							
Block1	Block2	Block3	Block4	Block5	Block6	Block7	Block8
0000 0000 0000 0000 1001 1110 0000 0000 0000 0000 00	0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 10	0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 00	0000 0000 0000 0000 0010 0111 1000 0000 0000 0000 00	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 10	1011 0000 0000 0000 0000 0000 0000 0000 0000 0000 00	0000 0000 0000 0000 0000 1011 0000 0000 0000 0000 00	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 010

The block number of each section is the crossover point. ‘X’ is the symbol represents the crossover operation. In the first section there are 5 blocks so the uniform crossover operation is performed from 5th bit as shown below.

Section 1: Number of Blocks 5 (Odd)

Block 1 X Block 5

$0000^{(1)}$ $0000^{(2)}$ $0000^{(3)}$ $0000^{(4)}$ $0000^{(5)}$ $0000^{(6)}$ $0000^{(7)}$ $0000^{(8)}$ $0000^{(9)}$ $0000^{(10)}$ $0000^{(11)}$ $0000^{(12)}$ $0000^{(13)}$ $0000^{(14)}$ $0101^{(15)}$ $1001^{(16)}$ $0000^{(17)}$

X

$\underline{0000}^{(1)}$ $\underline{0000}^{(2)}$ $\underline{0000}^{(3)}$ $\underline{0000}^{(4)}$ $\underline{0000}^{(5)}$ $\underline{0000}^{(6)}$ $\underline{0000}^{(7)}$ $\underline{0000}^{(8)}$ $\underline{0000}^{(9)}$ $\underline{0000}^{(10)}$ $\underline{0100}^{(11)}$ $\underline{0111}^{(12)}$ $\underline{0000}^{(13)}$ $\underline{0000}^{(14)}$ $\underline{0000}^{(15)}$ $\underline{0000}^{(16)}$ $\underline{0000}^{(17)}$ $\underline{1}^{(18)}$

$0000^{(1)}$ $\underline{0000}^{(1)}$ $0000^{(2)}$ $\underline{0000}^{(2)}$ $0000^{(3)}$ $\underline{0000}^{(3)}$ $0000^{(4)}$ $\underline{0000}^{(4)}$ $0000^{(5)}$ $\underline{0000}^{(5)}$ $0000^{(6)}$ $\underline{0000}^{(6)}$ $0000^{(7)}$ $\underline{0000}^{(7)}$ $0000^{(8)}$ $\underline{0000}^{(8)}$ $0000^{(9)}$
(Block 1.1)

$\underline{0000}^{(9)}$ $0000^{(10)}$ $\underline{0000}^{(10)}$ $0000^{(11)}$ $\underline{0100}^{(11)}$ $0000^{(12)}$ $\underline{0111}^{(12)}$ $0000^{(13)}$ $\underline{0000}^{(13)}$ $0000^{(14)}$ $\underline{0000}^{(14)}$ $0101^{(15)}$ $\underline{0000}^{(15)}$ $1001^{(16)}$ $\underline{0000}^{(16)}$ $0000^{(17)}$ $\underline{0000}^{(17)}$ $\underline{1}^{(18)}$
(Block 1.5)

Block 2 X Block 4

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0101 0110 0000 0000

X

$\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0100}$ $\underline{0001}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$

0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 **(Block 1.2)**

$\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0100}$ 0000 $\underline{0001}$ 0101 $\underline{0000}$
0110 $\underline{0000}$ 0000 $\underline{0000}$ 0000 $\underline{0000}$ **(Block 1.4)**
Block 1.3 is same as Block 3 of section 1.

Section 2: Number of Blocks 4 (Even)

Block 1 X Block 2

000 000 000 000 000 000 000 000 000 000 000 000 100 011 100 000 000 000 000 000 000 000 000 000 000 000 000 000 0

X

$\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{100}$ $\underline{011}$ $\underline{100}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{000}$ $\underline{010}$ $\underline{001}$ $\underline{1}$

000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{100}$ 000 $\underline{011}$ 000 $\underline{100}$ 000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{000}$ 000 $\underline{000}$ 100 $\underline{000}$ 011 $\underline{000}$ 1 **(Block 2.1)**

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

00000 000 000 000 000 000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000 000 000 000 010 000
001 0 1 **(Block 2.2)**

Block 3 X Block 4
100 000 000 000 000 000 000 000 000 000 000 000 000
000 000 000 000 000 000 010 101 000 000 000 000 000
000 000 0
X
000 000 000 000 000 000 000 000 000 000 000 000 010
110 010 000 000 000 000 000 000 000 000 000 000 000
000 000 00

100 000 000 000 000 000 000 000 000 000 000 000 000
000 000 000 000 000 000 000 000 000 000 000 000 010
000 110 0
(Block 2.3)

00 010 000 000 000 000 000 000 000 000 010 000 101
000 000 000 000 000 000 000 000 000 000 000 000 000
000 000 0 00 **(Block 2.4)**

Section 3: Number of Blocks 3 (Odd)
Block 1 X Block 3
00 00 00 00 00 00 00 01 00 11 11 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 01 00 11 01 00 00 00 00 00 00 00 00 00 00 00 00 00 0
X
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 01 01 10 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 10
0

00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 11
00 11 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 01 00 01 00 10 00 00 00 00 00 00 0
(Block 3.1)

0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 01 00 00 00 11 00 01 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 01 0
10 0 **(Block 3.3)**

Block 3.2 is same as Block 2 of section 3.

Section 4: Number of Blocks 2 (Even)
Block 1 X Block 2
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 1 0 1 1 0 0 0 0 0 0 0 0 0 00 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
X
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0

1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0

1 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 1 0 1 **(Block 4.1)**

0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0
0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 0 1
0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
0 0 0 0 0 0 **(Block 4.2)**

Section 5: Number of Blocks 7 (Odd)
Block 1 X Block 7
000000 000000 000000 000000 000000 000000 010010
100000
X
000001 010101 000000 000000 000000 000000 000000
000000 00000
000000 000001 000000 010101 000000 000000 000000
000000 **(Block 5.1)**
000000 000000 000000 000000 010010 000000 100000
000000 00000 **(Block 5.7)**

Block 2 X Block 6
000000 000000 000000 000000 000000 000000 000000
000000
X
000000 000000 000000 000000 000000 000000 000000
000000
000000 000000 000000 000000 000000 000000 000000
000000 **(Block 5.2)**
000000 000000 000000 000000 000000 000000 000000
000000 **(Block 5.6)**

Block 3 X Block 5
000001 011000 000000 000000 000000 000000 000000
000000
X
000000 000000 000000 000000 000000 000000 010001
110000
000001 000000 011000 000000 000000 000000 000000
000000 **(Block 5.3)**
000000 000000 000000 000000 000000 010001 000000
110000 **(Block 5.5)**
Block 5.4 is same as Block 4 of section 5.

Section 6: Number of Blocks 8 (Even)
Block 1 X Block 2
0000000 0000000 0010011 1100000 0000000 0000000
X
0000000 0000000 0000000 0000000 0000000 0001010

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

0000000 0000000 0000000 0000000 0010011 0000000
(Block 6.1)
1100000 0000000 0000000 0000000 0000000 0001010
(Block 6.2)

Block 3 X Block 4
0110000 0000000 0000000 0000000 0000000 0000000
X
0000000 0000000 0000100 1111000 0000000 0000000
0110000 0000000 0000000 0000000 0000000 0000100
(Block 6.3)
0000000 1111000 0000000 0000000 0000000 0000000
(Block 6.4)

Block 5 X Block 6
0000000 0000000 0000000 0000000 0000000 0000010
X
1011000 0000000 0000000 0000000 0000000 0000000
0000000 1011000 0000000 0000000 0000000 0000000
(Block 6.5)
0000000 0000000 0000000 0000000 0000010 0000000
(Block 6.6)

Block 7 X Block 8
0000000 0000000 0000001 0110000 0000000 0000000
X
0000000 0000000 0000000 0000000 0000000 0000000 10010
0000000 0000000 0000000 0000000 0000001 0000000
(Block 6.7)
0110000 0000000 0000000 0000000 0000000 0000000 10010 **(Block 6.8)**

Now concatenate all the modified blocks and discarded bits'11' to produce the final cipher text. The total number of bits in final cipher text is 2048. The 2048 bits will be divided into 32 blocks each of 64 bits. The decimal equivalent of each number is shown below:

1) 0000000000000000000000000000000000000000 000000000000000000000 = 0
2) 0000000000000000000000000000000000000000 000000000000000000000 = 0
3) 0000000000000000000000000000000000000000 000000000000001001111 = 79
4) 0000000000000000000000000000000000100000 000010101000001100000 = 67194976
5) 0000000000000000000000000000001000000011 100000000000000000101 = 1081081861
6) 0000100100000000000000000000000000000000 000000000001000000110 = 6485183463413519 42
7) 0010000000000000000000000000010000001100010 000000000000000000000 = 2305843026599084032
8) 0000000000000000000000000000000000000000 000000000010000001011 = 1035
9) 0000000000000000000000000000000000000000 000000000000000000000 = 0
10) 0000000000100001100000100000000000000000 000000010000101000000 = 4715805371539776
11) 0000000000000000000000000000000000000000 000000000000000000000 = 0
12) 0000000100000011001100000000000000000000 00000000000000000001= 72954795526193152
13) 0000010001001000000000000000000000000000 000000000000001010100 = 19281035904679941
14) 0000000000000000000000000000000000000000 000000000000001010100 =84
15) 0000000000000000000000000000000000000000 000000000000010000001 =129
16) 1000100000000000000000000000000000000000 000000000001000101001 = 9799832789158199849
17) 0000000000000000000000000000000010000000100 010000000000000000000 = 2156396544
18) 0000000000000000000000000000000000000000 000000001000100010000 =4368
19) 0000000000000000000000000000000010000010100 010000000000000000000 = 2189950976
20) 0000000000000000000000000000000000000000 000000001000001000101=4165
21) 0000000000000000000000000000000010000010101 010000000000000000000 = 2192048128
22) 0000000000000000000000000000000100000001010 100000000000000000000 = 4316987392
23) 0000000000000000000000000000000000000000 000000000000001000000 =64
24) 0110000000000000000000000000000000000000 000000000000001011001 = 6917529027641081945
25) 0000000000000000000000000000000000000000 000000001000100000011= 4355
26) 0000000000000000000000000000000000000000 000000000000000000000 = 0
27) 0000000000000100100000000100000000000000000 000000000000000000000 = 1267187151011840
28) 0000000100111000000011000000000000000000000 000000000000001010011 = 85594781199106131
29) 0000000000000000000000000000000000001000000 000111100000000000000 = 134463488
30) 0000000000000000000000001011000000000000000 000000000000000000000 = 755914244096
31) 0000000000000000000000000001000000000000000 000000000000000000000 = 34359738368
32) 0000000100000000110000000000000000000000000 000000000000001001011 = 72268700270461003

The final cipher text is:
0 0 79 67194976 1081081861 6485183463413519 42 2305843026599084032 1035 0 4715805371539776 0 72954795526193152 19281035904679941 84 129 9799832789158199849 2156396544 4368 2189950976 4165 2192048128 4316987392 64 6917529027641081945 4355 0 1267187151011840 85594781199106131 134463488 755914244096 34359738368 72268700270461003

B. Decryption

At the decryption side, the input is the final cipher text obtained from the encryption.

Key-1= 84, key-2 = 6543278. Again by using key-2 the cipher text is divided into sections and blocks. Each section consists of (2048/6) = 341 bits and the remainder '11' is discarded for future use.

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

Section 1 (Each Block contains(341/5) bits or 68 bits with 1 remainder				
Block 1	Block 2	Block 3	Block 4	Block 5
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 0000	00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 0000	00000000 00000000 00000000 00000000 00000000 00000000 01001111 00000000 0000	00000000 00000000 00000000 01000000 00010101 00000110 00000000 00000000 0000	00000000 00000000 01000000 01110000 00000000 00000101 00001001 00000000 0000 1

Section 2 (Each Block contains(341/4) bits or 85 bits with 1 remainder			
Block 1	Block 2	Block 3	Block 4
0000000000 0000000000 0000000000 0001000000 1100010000 0000000000 0000000000 0010000001 10001	0000000000 0000000000 0000000000 0000000000 0000000000 0000000000 0000000000 0000010000 00101	1000000000 0000000000 0000000000 0000000000 0000000000 0000000000 0000000000 0000001000 01100	0001000000 0000000000 0000000000 1000010100 0000000000 0000000000 0000000000 0000000000 000000

Section 3 (Each Block contains(341/3) bits or 113 bits with 2 remainder		
Block 1	Block 2	Block 3
000000000000 000000000000 000001000000 110011000000 000000000000 000000000000 000000000000 000000010001 001000000000 00000	000000000000 000000000000 000000101010 000000000000 000000000000 000000000000 000000000000 000000000010 101000000000 00000	000000000000 000000000000 000000000000 000000001000 000110001000 000000000000 000000000000 000000000000 000000000010 0010100

Section 4 (Each Block contains(341/2) bits or 170 bits with 1 remainder	
Block 1	Block 2
100000000000000000000000 000000000100000001000100 000000000000000000000000 000000000000000000000000 000000000000000000001000 100010000000000000000000 000000000000000001000001 01	000100000000000000000000 000000000000000000000000 000000000000000000000000 001000001000101000000000 000000000000000000000001 000001010101000000000000 000000000000000000000000 000

Section 5 (Each Block contains(341/7) bits or 48 bits with 5 remainder						
Block 1	Block 2	Block 3	Block 4	Block 5	Block 6	Block 7
00000 00000 01000 00001 01010 00000 00000 00000 00000 000	00000 00000 00000 00000 00000 00000 00000 00000 00000 000	00000 10000 00011 00000 00000 00000 00000 00000 00000 000	00000 00000 00000 00000 01011 00100 00000 00000 00000 000	00000 00000 00000 00000 00000 00000 01000 10000 00110 000	00000 00000 00000 00000 00000 00000 00000 00000 00000 000	00000 00000 00000 00000 00000 10010 00000 01000 00000 00000 000

Section 6 (Each Block contains(341/8) bits or 42 bits with 5 remainder							
Block1	Block2	Block3	Block4	Block5	Block6	Block7	Block8
0000 0000 0000 0000 0000 0000 0000 0010 0110 0000 00	1100 0000 0000 0000 0000 0000 0000 0000 0000 001010	0110 0000 0000 0000 0000 0000 0000 0000 0000 0001 00	0000 0001 1110 0000 0000 0000 0000 0000 0000 0000 00	0000 0001 0110 0000 0000 0000 0000 0000 0000 0000 00	0000 0000 0000 0000 0000 0000 0000 0000 0100 0000 00	0000 0000 0000 0000 0000 0000 0000 0000 0010 0000 00	0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 010

The uniform point crossover is applied on the blocks of every section to generate intermediate cipher text.

Section 1: Number of Blocks 5 (Odd)

Block 1 X Block 5

$0000^{(1)}$ $0000^{(2)}$ $0000^{(3)}$ $0000^{(4)}$ $0000^{(5)}$ $0000^{(6)}$ $0000^{(7)}$ $0000^{(8)}$ $0000^{(9)}$ $0000^{(10)}$ $0000^{(11)}$ $0000^{(12)}$ $0000^{(13)}$ $0000^{(14)}$ $0000^{(15)}$ $0000^{(16)}$ $0000^{(17)}$

X

$0000^{(1)}$ $0000^{(2)}$ $0000^{(3)}$ $0000^{(4)}$ $0100^{(5)}$ $0000^{(6)}$ $0111^{(7)}$ $0000^{(8)}$ $0000^{(9)}$ $0000^{(10)}$ $0000^{(11)}$ $0101^{(12)}$ $0000^{(13)}$ $1001^{(14)}$ $0000^{(15)}$ $0000^{(16)}$ $0000^{(17)}$ $1^{(18)}$

$0000^{(1)}$ $0000^{(3)}$ $0000^{(5)}$ $0000^{(7)}$ $0000^{(9)}$ $0000^{(11)}$ $0000^{(13)}$ $0000^{(15)}$ $0000^{(17)}$ $0000^{(2)}$ $0000^{(4)}$ $0000^{(6)}$ $0000^{(8)}$ $0000^{(10)}$ $0101^{(12)}$ $1001^{(14)}$ $0000^{(16)}$ **(Block 1.1)**

$0000^{(2)}$ $0000^{(4)}$ $0000^{(6)}$ $0000^{(8)}$ $0000^{(10)}$ $0000^{(12)}$ $0000^{(14)}$ $0000^{(16)}$ $0000^{(1)}$ $0000^{(3)}$ $0100^{(5)}$ $0111^{(7)}$ $0000^{(9)}$ $0000^{(11)}$ $0000^{(13)}$ $0000^{(15)}$ $0000^{(17)}$ $1^{(18)}$ **(Block 1.5)**

Block 2 X Block 4

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

X

0000 0000 0000 0000 0000 0000 0100 0000 0001 0101 0000

0110 0000 0000 0000 0000 0000

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0101 0110 0000 0000**(Block 1.2)**

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0100 0001 0000 0000 0000 0000**(Block 1.4)**

Block 1.3 is same as Block 3 of section 1.

Section 2: Number of Blocks 4 (Even)

Block 1 X Block 2

000 000 000 000 000 000 000 000 000 000 000 100 000 011 000 100 000 000 000 000 000 000 000 000 100 000 011 000 1

X

00 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 010 000 001 0 1

000 000 000 000 000 000 000 000 000 000 000 000 100 011 100 000 000 000 000 000 000 000 000 000 000 000 000 000 0 **(Block 2.1)**

000 000 000 000 000 100 011 100 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 010 001 1 **(Block 2.2)**

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

Block 3 X Block 4
100 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 010 000 110 0
X
00 010 000 000 000 000 000 000 000 000 010 000 101 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 0 00

100 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 010 101 000 000 000 000 000 000 000 0
(Block 2.3)
000 000 000 000 000 000 000 000 000 000 000 000 010 110 010 000 000 000 000 000 000 000 000 000 000 000 000 000 00 **(Block 2.4)**

Section 3: Number of Blocks 3 (Odd)
Block 1 X Block 3
00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 11 00 11 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 01 00 10 00 00 00 00 00 00 0
X
0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 11 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 01 0 10 0
00 00 00 00 00 00 00 01 00 11 11 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 11 01 00 00 00 00 00 00 00 00 00 00 00 00 00 0
(Block 3.1)
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 10 0
(Block 3.3)

Block 3.2 is same as Block 2 of section 3.
Section 4: Number of Blocks 2 (Even)
Block 1 X Block 2
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1
X
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 0 1 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 **(Block 4.1)**

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 **(Block 4.2)**

Section 5: Number of Blocks 7 (Odd)
Block 1 X Block 7
000000 000001 000000 010101 000000 000000 000000 000000
X
000000 000000 000000 000000 010010 000000 100000 000000 00000
000000 000000 000000 000000 000000 000000 010010 100000 **(Block 5.1)**
000001 010101 000000 000000 000000 000000 000000 000000 00000 **(Block 5.7)**
Block 2 X Block 6
000000 000000 000000 000000 000000 000000 000000 000000
X
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000 **(Block 5.2)**
000000 000000 000000 000000 000000 000000 000000 000000 **(Block 5.6)**

Block 3 X Block 5
000001 000000 011000 000000 000000 000000 000000 000000
X
000000 000000 000000 000000 000000 010001 000000 110000
000001 011000 000000 000000 000000 000000 000000 000000 **(Block 5.3)**
000000 000000 000000 000000 000000 000000 010001 110000 **(Block 5.5)**

Block 5.4 is same as Block 4 of section 5.
Section 6: Number of Blocks 8 (Even)
Block 1 X Block 2
0000000 0000000 0000000 0000000 0010011 0000000
X
1100000 0000000 0000000 0000000 0000000 0001010
0000000 0000000 0010011 1100000 0000000 0000000
(Block 6.1)
0000000 0000000 0000000 0000000 0000000 0001010
(Block 6.2)

Block 3 X Block 4
0110000 0000000 0000000 0000000 0000000 0000100
X
0000000 1111000 0000000 0000000 0000000 0000000
0110000 0000000 0000000 0000000 0000000 0000000
(Block 6.3)
0000000 0000000 0000100 1111000 0000000 0000000
(Block 6.4)
Block 5 X Block 6
0000000 1011000 0000000 0000000 0000000 0000000
X

Copyright to IJARCCE

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

0000000 0000000 0000000 0000000 0000010 0000000
0000000 0000000 0000000 0000000 0000000 0000010
(Block 6.5)
1011000 0000000 0000000 0000000 0000000 0000000
(Block 6.6)

Block 7 X Block 8
0000000 0000000 0000000 0000000 0000001 0000000
X
0110000 0000000 0000000 0000000 0000000 0000000 10010
0000000 0000000 0000001 0110000 0000000 0000000
(Block 6.7)
0000000 0000000 0000000 0000000 0000000 0000000 10010
(Block 6.8)

Concatenate all the modified blocks along with the discarded bits ‘11’ to produce the intermediate cipher text. Now the total 2048 bits of intermediate cipher text will be divided into 32 blocks of 64bits.

TABLE II
REPRESENTATION OF 64 BITS BINARY CODE OF INTERMEDIATE CIPHER IN CHARACTERS

Binary Code(64-bits) of ASCII value	Character
0000000000000000000000000000000000 000000000000000000000001011001	Y
0000000000000000000000000000000000 000000000000000000000001010110	V
0000000000000000000000000000000000 000000000000000000000001001111	O
0000000000000000000000000000000000 000000000000000000000001000001	A
0000000000000000000000000000000000 000000000000000000000001000111	G
0000000000000000000000000000000000 000000000000000000000001000111	G
0000000000000000000000000000000000 000000000000000000000001000111	G
0000000000000000000000000000000000 000000000000000000000001000111	G
0000000000000000000000000000000000 000000000000000000000001010100	T
0000000000000000000000000000000000 000000000000000000000001011001	Y
0000000000000000000000000000000000 000000000000000000000001001111	O
0000000000000000000000000000000000 000000000000000000000001001101	M
0000000000000000000000000000000000 000000000000000000000001010100	T
0000000000000000000000000000000000 0000000000000001010100	T
0000000000000000000000000000000000 000000000000000000000001011000	X
0000000000000000000000000000000000 000000000000000000000001011001	Y
0000000000000000000000000000000000 000000000000000000000001010100	T
0000000000000000000000000000000000 000000000000000000000001001011	K
0000000000000000000000000000000000 000000000000000000000001000101	E
0000000000000000000000000000000000 000000000000000000000001001101	M
0000000000000000000000000000000000 000000000000000000000001001111	O
0000000000000000000000000000000000 000000000000000000000001001010	J
0000000000000000000000000000000000 000000000000000000000001011000	X
0000000000000000000000000000000000 000000000000000000000001011001	Y
0000000000000000000000000000000000 000000000000000000000001000111	G
0000000000000000000000000000000000 000000000000000000000001010101	U
0000000000000000000000000000000000 000000000000000000000001001111	O
0000000000000000000000000000000000 000000000000000000000001010011	S
0000000000000000000000000000000000 000000000000000000000001001111	O
0000000000000000000000000000000000 000000000000000000000001010110	V
0000000000000000000000000000000000 000000000000000000000001011000	X
0000000000000000000000000000000000 000000000000000000000001001011	K

Intermediate cipher text is:

YVOAGGGGTYOMTTXYTKEMOJXYGUOSOVXK
Substitute the intermediate cipher text by its positional values and subtract it by key-1 to obtain the plain text. As the subtraction of intermediate cipher by key-1 results a negative number, adding the intermediate cipher text with 78, 104 is desirable (26+26+26 = 78, 26+26+26+26 =104). Y = (25+78) – 84 = 19 = **S**, V = (22+78) – 84 = 16 = **P**, O = (15+78) – 84 = 9 = **I**, A = (1+104) – 84 = 21 = **U**, G = (7+78) – 84 = 1 = **A**, G = (7+78) – 84 = 1 = **A**, G = (7+78) – 84 = 1 = **A**, G = (7+78) – 84 = 1 = **A**, T = (20+78) – 84 = 14 = **N**, Y = (25+78) – 84 = 19 = **S**, O = (15+78) – 84 = 9 = **I**, M = (13+78) – 84 = 7 = **G**, T = (20+78) – 84 = 14 = **N**, T = (20+78) – 84 = 14 = **N**, X = (24+78) – 84 = 18 = **R**, Y = (25+78) – 84 = 19 = **S**, T = (20+78) – 84 = 14 = **N**, K = (11+78) – 84 = 5= **E**, E = (5+104) – 84 = 25 = **Y**, M = (13+78) – 84 = 7 = **G**, O = (15+78) – 84 = 9 = **I**, J = (10+78) – 84 = 4 = **D**, X = (24+78) – 84 = 18 = **R**, Y = (25+78) – 84 = 19 = **S**, G = (7+78) – 84 = 1 = **A**, U = (21+78) – 84 = 15 = **O**, O = (15+78) – 84 = 9 = **I**, S = (19+78) – 84 = 13 = **M**, O = (15+78) – 84 = 9 = **I**, V = (22+78) – 84 = 16 = **P**, X = (24+78) – 84 = 18 = **R**, K = (11+78) – 84 = 5 = **E**

Now, each character obtained is placed in the cells and the plain text is obtained from the proposed technique as shown in Figure-6.

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

S	P	I	U	A	A	A
A	N	S	I	G	N	N
R	S	N	E	Y	G	
I	D	R	S	A	O	
I	M	I	P	R	E	

Figure-6. Arrangement of decrypted characters in the cells to obtain the plain text

Plaintext:
SPAINRUSSIAINDIAGERMANYSINGAPORE

V. SYNTHESIS RESULTS AND COMPARISON

The synthesis results for transmitting a 16 characters input plain text BIOTECHNOLOGICAL using three different crossover operations i.e., single, double and uniform point crossover is observed. The synthesis results and comparison is shown in the Table-3.

TABLE-3
SYNTHESIS RESULTS OF VARIOUS CROSSOVER OPERATIONS

Device Utilization	Single Point Crossover	Double Point Crossover	Uniform Point Crossover
No. of slices (14752)	11877	4646	4660
No. of 4 input LUTs (29504)	22572	8657	8676
No. of Bonded IOBs	2048	2048	2048
Combinational Path Delay in ns	68.332	54.820	54.764

The synthesis results for transmitting a 18 characters input plain text ALLROADSLEADTOROME using three different sizes of matrix i.e., square, rectangular, matrix with different number of cells (as used in the proposed technique) is observed. The synthesis results and comparison is shown in the Table-4.

TABLE-4
SYNTHESIS RESULTS OF DIFFERENT SIZE OF MATRICES

Device Utilization	Square Matrix	Rectangular Matrix	Least Square Matrix with Column Cells
No. of slices (14752)	17218	12497	12427
No. of 4 input LUTs (29504)	32796	23762	23646
No. of Bonded IOBs	3200	2304	2304
Combinational Path Delay in ns	66.010	68.644	64.619

The synthesis results for transmitting a input plain text using different digits of key-2 is observed. The synthesis results shown below in the Table-5.

TABLE-5
SYNTHESIS RESULTS OF DIFFERENT DIGITS OF KEY-2

Device Utilization	Different Key-2 Used			
	4532	65743	7654321	843217695
No. of slices (14752)	11072	11072	11072	11072
No. of 4 input LUTs (29504)	21069	21069	21069	21069
No. of Bonded IOBs	2048	2048	2048	2048
Delay in ns	64.119	64.119	64.119	64.119

VI. CONCLUSION

From the synthesis results of Section V, it is observed that

1) By using the uniform point crossover the more complex cipher text is generated with minimum device utilization and better delay performance. [TABLE-3]

2) To transmit a string of characters which is not a square number for example 18 characters, a 5*5 square matrix is used. Out of the 25 available cells of the square matrix only 18 cells are used to place the characters and the other cells are kept empty to transmit 64 bits of 0(zero) unnecessarily.
A rectangular matrix of size 6*3 is also used to tranmit the 18 characters.
A nearby least square matrix to 18 is 4*4 square matrix with 2 column cells is used to tranmit the same 18 characters which results in minimum device utilization with better delay performance [TABLE-4]

3) The algorithm is also performed using different digits of key-2. It is observed that using various values of key-2 does not change the delay or device utilization ratio but using a larger key-2 value consisting of different numbers in it makes the data more secure. [TABLE-5]

4) Thus in the proposed technique a uniform point crossover operation is performed which uses the nearby least square matrix with column cells according to the number of characters in the input plain text. The uniform point crossover has been proved to generate complex cipher text which makes the algorithm susceptible from the attacker. A 7-digit key-2 with different numbers in its digit is used to make the algorithm more secure.

TABLE-6
SYNTHESIS RESULTS OF PROPOSED AND EXISTING ALGORITHM

Device Utilization	Existing Algorithm	Proposed Algorithm
No. of slices	26168	21982
No. of 4 input LUTs	49758	41765
No. of Bonded IOBs	4608	4096
Combinational Path Delay in ns	59.012	53.074

The Table-6 shows the synthesis results for transmitting 32 characters using the proposed algorithm and compares it with the existing algorithm. The existing algorithm[1] uses the square matrix with single point crossover and a 5-digit key-2 for transmission of 32 characters. It is observed that by using the proposed technique the device utilization ratio is comparatively less with minimum combinational path delay.

VII. FUTURE SCOPE

The cryptographic algorithm can be performed using different crossover operators to obtain less combinational path delay with minimum device utilization. The different size of matrices can be tried for implementation. The fitness test can be applied to take the fittest modified block to generate more complex cipher text. Different types of crossover operations can be performed on blocks of different sections to make the algorithm more complex.

REFERENCES

[1] Dr. Subhranil Som, Ms. Mandira Banerjee,"Cryptographic Technique by Square Matrix and Single Point Crossover on Binary Field", 2013.
[2] S. Som, M. Banerjee, "Cryptographic Technique Using Substitution through Circular Path Followed By Genetic Function", CCSN-2012, 1st International conference on Computing, Communication and Sensor Network, November 22nd and 23rd, 2012, Roukela, India. Accepted
[3] Poonam Garg, "Genetic algorithms and simulated annealing: a comparison between three approaches for the crypto analysis of transposition cipher" IMT, INDIA-2004.
[4] A.J.Bagnall, "The Applications of Genetic Algorithms in Cryptanalysis", School of Information Systems, University Of East Anglia, 1996.
[5] N.Koblitz, "A Course in Number Theory and Cryptography", Springer-Verlag, New York, Inc., 1994.
[6] Menzes A. J., Paul, C., Van Dorschot, V., Vanstone, S. A.,
[7] "Handbook of Applied Cryptography", CRS Press 5th Printing; 2001.
[8] National Bureau Standards, "Data Encryption Standard (DES)," FIPS Publication 46; 1977.
[9] Tragha A., Omary F., Mouloudi A.,"ICIGA: Improved Cryptography Inspired by Genetic Algorithms", Proceedings of the International Conference on Hybrid Information Technology (ICHIT'06), pp. 335-341, 2006.
[10] Melanie Mitchell, "An introduction to Genetic Algorithms". A Bradford book.
[11] H. Bhasin and S. Bhatia, "Application of Genetic Algorithms in Machine learning", IJCSIT, Vol. 2 (5), 2011.
[12] Dr. G. Raghavendra, Nalini N, "a new encryption and decryption algorithm combining the features of genetic algorithm (GA) and cryptography" NIE, Mysore.
[13] A. J. Bagnall, "the application of genetic algorithms in cryptanalysis" School of information system, University of East Anglia, 1996.

BIOGRAPHIES

Ishrath Jahan received the B. Tech degree in Electronics and Communication Engineering from Shadan Womens College of Engineering and Technolo gy. She is pursuing Masters in Embedded Systems from Stanley College of Engineering and Technology for Women, Hyderabad, India.

Udayini Chandana received the B.E. degree in Electronics and Communication Engineering from Andhra University, India and the M.S. degree in Electrical Engineering from the University of Houston, Texas, USA, in 1994 and 1996 respectively. She worked in the software industry for a period of more than 4 years on various platforms as a System Analyst. Since 2011, she has been with Stanley College of Engineering and Technology for Women, Hyderabad, India, where she is currently an Assistant Professor. Her research interests include Digital Electronics and Image Processing Systems.