ECE 530 spring 2017

Project Report

A Triple Hill Cipher Algorithm implementation to increase security of the encrypted binary data

Navaneeth Gowda Thandavamurthy
A20378763

nthandavamurthy@hawk.iit.edu

Abstract

In the field of the data communication the security of the encrypted binary data is a crucial task. In this project, the HILL CIHER algorithm, a symmetric key encryption algorithm is used. It is a classical symmetric cipher based on matrix transformation. **Triple Hill cipher algorithm** that consists of three stages of the hill cipher is proposed considering each stage a block cipher with the block length of 128 bits and 256 bits' key.

To increase the security of message, message to be encrypted is processed by the by the block cipher in three stages and each stage consisting of eight rounds with different eight keys and in each round three operations are implemented: matrix multiplication of plaintext and cipher matrix multiplication, XOR operation and a stir operation. Following the triple DES algorithm, three stage implementations is proposed.

1. Introduction

In the present scenario security of the data transmitted over the network or the channel is very important to avoid the risk of unauthorized access. Cryptography deals with the encryption of the data so that it is transmitted over the channel securely. The plain text, which must be encrypted is processed to the cipher text, which is the encrypted form of the plain text using a specific algorithm. At the receiver's side decryption is performed to obtain back the plain text.

There are two types of cryptosystems: one is the asymmetric and other is the symmetric. In the asymmetric type, two keys are used and in symmetric algorithm only one key is used. In the project, Hill cipher algorithm is implemented, which involves the generation of the key matrix and multiplying the key matrix with the plain text matrix to obtain the cipher text. Implementation involves the key generator, multiplier and the other implementations to enhance the security.

2. Implementation:

Implementation of the algorithm is as shown in the figure below,

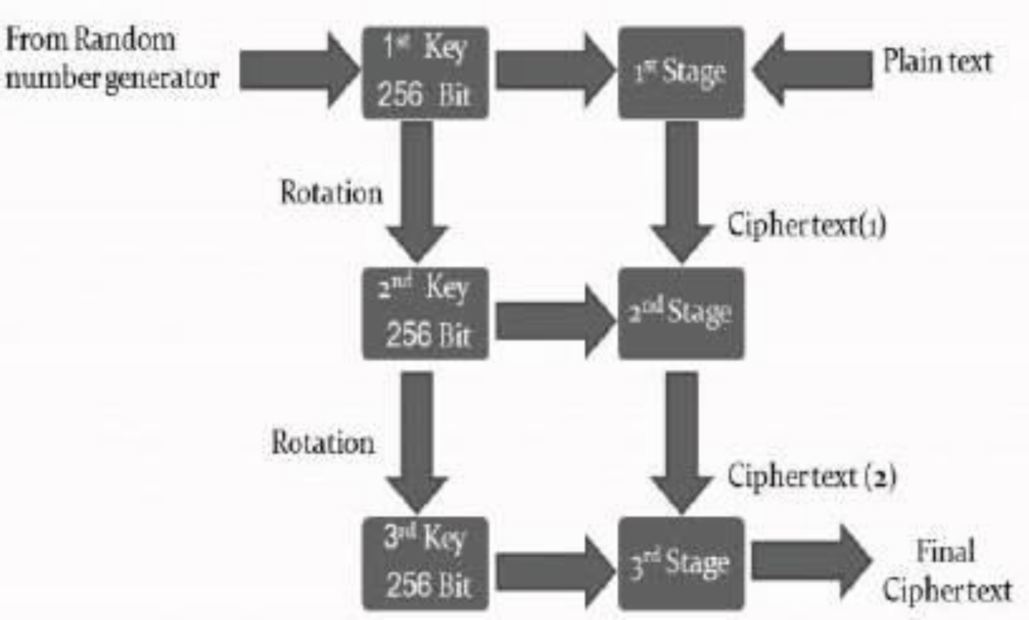


Fig1: Algorithm implementation

As shown in the above figure, a random number generator is used to generate the key sequence which is 256 bit. Then the obtained key is rotated once to get second key and the third key is obtained by rotating the first key twice. Then the plain text is encrypted thrice using these three keys.

a) Random Number Generator:

Random number generators are the logic devices designed to generate a random sequence of signals. There are two types of the number generators, true random number generator and the pseudo random number generator.

In the project, a pseudo random generator is designed to obtain the key and as explained below:

The first 256-bit key is generated using the Linear Feedback shift register made of sequential shift registers with combinational feedback logic connected to it which can generate a sequence of the binary values in the pseudo random manner. A four-bit LFSR is as shown in the figure below:

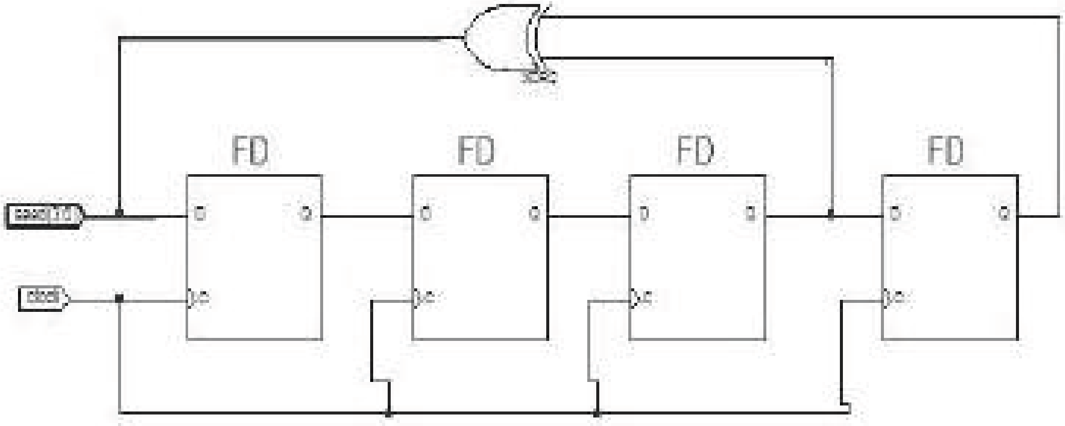
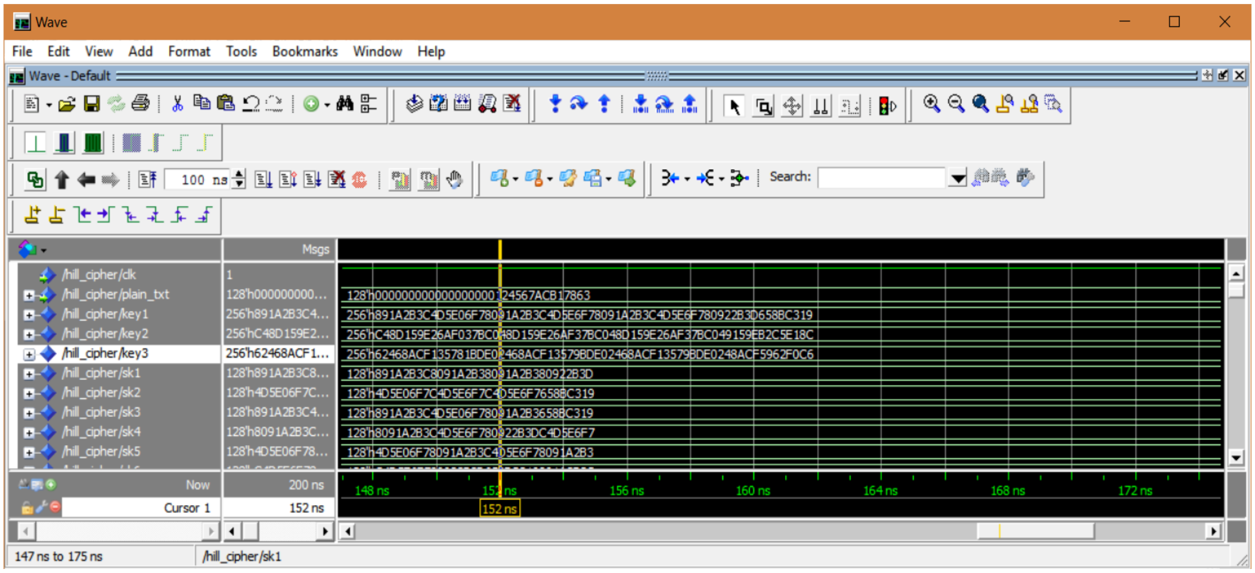


Fig2: four bit LFSR

Feedbacks around an LFSR's shift register are connected to taps and these taps constitutes for the xor. The taps selected determines the values generated. The output of the LFSR depends on the seed value connected to the shift registers. In the developed project 123456789ABC0DEF0123456789ABCDEF0123456789ABCDEF0124567ACB17 8632 is the seed value used based on which the keys shown in the simulation below is obtained:



```
/hill_cipher/clk        1
/hill_cipher/plain_txt  128'h000000000...  128'h00000000000000000001 24567ACB17863
/hill_cipher/key1       256'h891A2B3C4...  256'h891A2B3C4D5E06F78091A2B3C4D5E6F78091A2B3C4D5E6F780922B3D658BC319
/hill_cipher/key2       256'hC48D159E2...  256'hC48D159E26AF037BC048D159E26AF37BC048D159E26AF37BC049159EB2C5E18C
/hill_cipher/key3       256'h62468ACF1...  256'h62468ACF135781BDE02468ACF13579BDE02468ACF13579BDE0248ACF5962F0C6
/hill_cipher/sk1        128'h891A2B3C8...  128'h891A2B3C8091A2B38091A2B380922B3D
/hill_cipher/sk2        128'h4D5E06F7C...  128'h4D5E06F7C4D5E6F7C4D5E6F7658BC319
/hill_cipher/sk3        128'h891A2B3C4...  128'h891A2B3C4D5E06F78091A2B3658BC319
/hill_cipher/sk4        128'h8091A2B3C...  128'h8091A2B3C4D5E6F780922B3DC4D5E6F7
/hill_cipher/sk5        128'h4D5E06F78...  128'h4D5E06F78091A2B3C4D5E6F78091A2B3
Now       200 ns
Cursor 1  152 ns
147 ns to 175 ns    /hill_cipher/sk1
```

b) Encryption Algorithm implementation:

To implement the security algorithm, show in the fig1, in each stage as depicted in the consists of the eight rounds of encryption using the sub keys generated form the key obtained from random number generator. Each stage is as shown in the figure below:

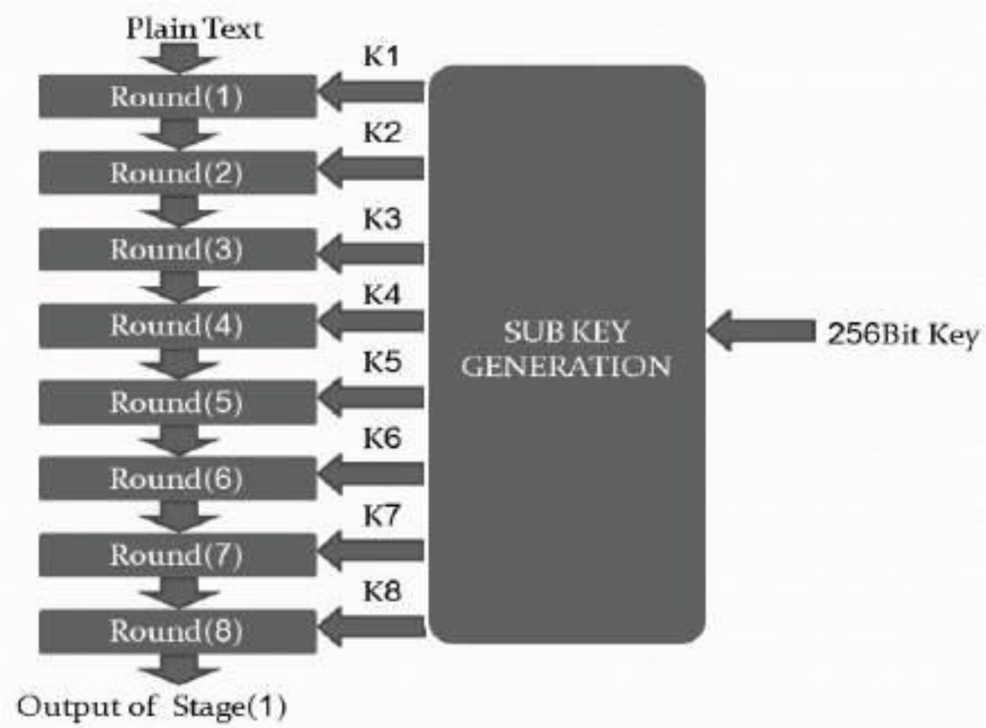
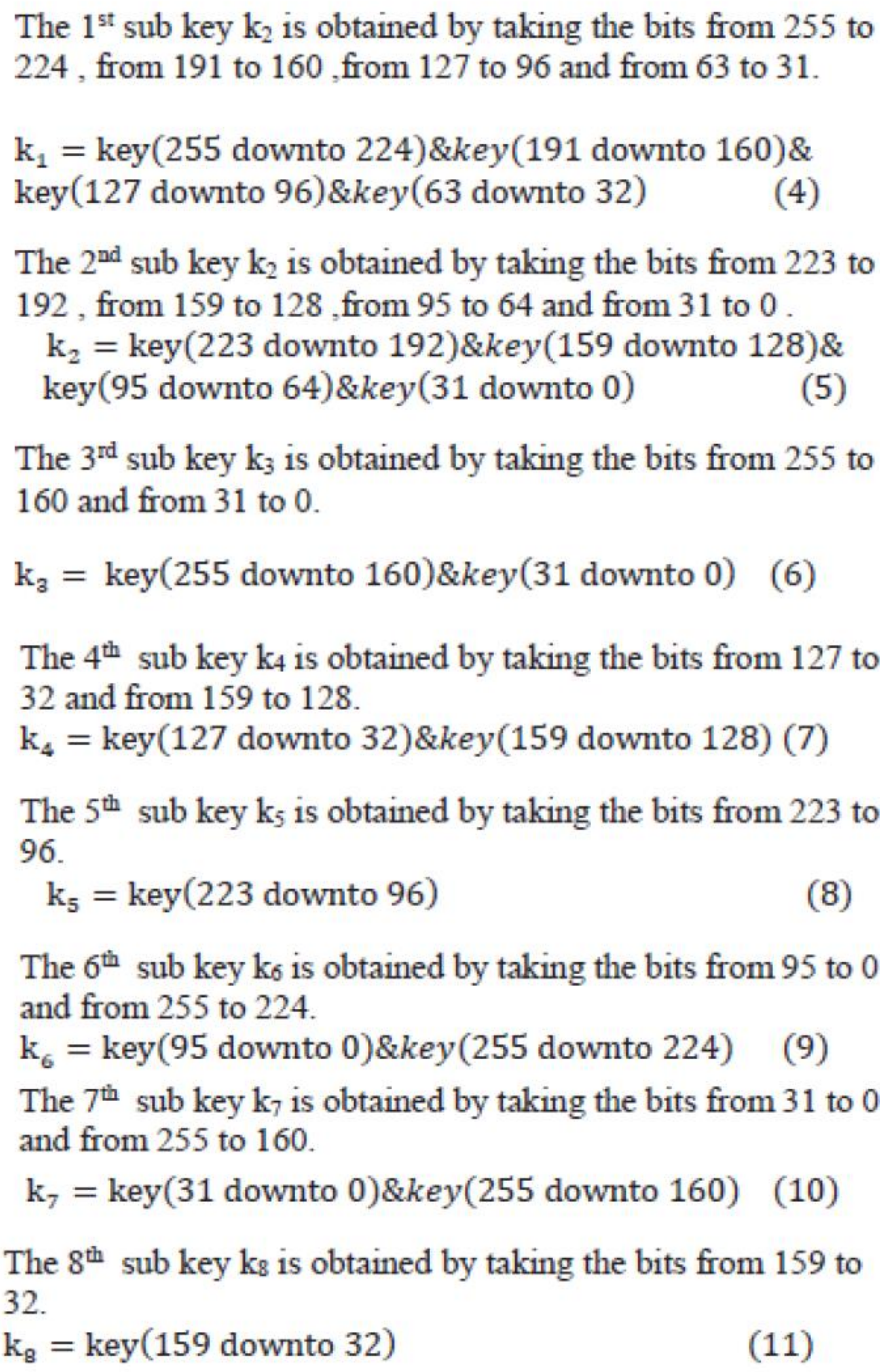


Fig3: encryption in each stage

To generate the sub key following algorithm is followed:



The 1st sub key k_2 is obtained by taking the bits from 255 to 224 , from 191 to 160 ,from 127 to 96 and from 63 to 31.

$$k_1 = \text{key}(255 \text{ downto } 224)\&key(191 \text{ downto } 160)\& \text{key}(127 \text{ downto } 96)\&key(63 \text{ downto } 32) \qquad (4)$$

The 2nd sub key k_2 is obtained by taking the bits from 223 to 192 , from 159 to 128 ,from 95 to 64 and from 31 to 0 .

$$k_2 = \text{key}(223 \text{ downto } 192)\&key(159 \text{ downto } 128)\& \text{key}(95 \text{ downto } 64)\&key(31 \text{ downto } 0) \qquad (5)$$

The 3rd sub key k_3 is obtained by taking the bits from 255 to 160 and from 31 to 0.

$$k_3 = \text{key}(255 \text{ downto } 160)\&key(31 \text{ downto } 0) \qquad (6)$$

The 4th sub key k_4 is obtained by taking the bits from 127 to 32 and from 159 to 128.

$$k_4 = \text{key}(127 \text{ downto } 32)\&key(159 \text{ downto } 128) \qquad (7)$$

The 5th sub key k_5 is obtained by taking the bits from 223 to 96.

$$k_5 = \text{key}(223 \text{ downto } 96) \qquad (8)$$

The 6th sub key k_6 is obtained by taking the bits from 95 to 0 and from 255 to 224.

$$k_6 = \text{key}(95 \text{ downto } 0)\&key(255 \text{ downto } 224) \qquad (9)$$

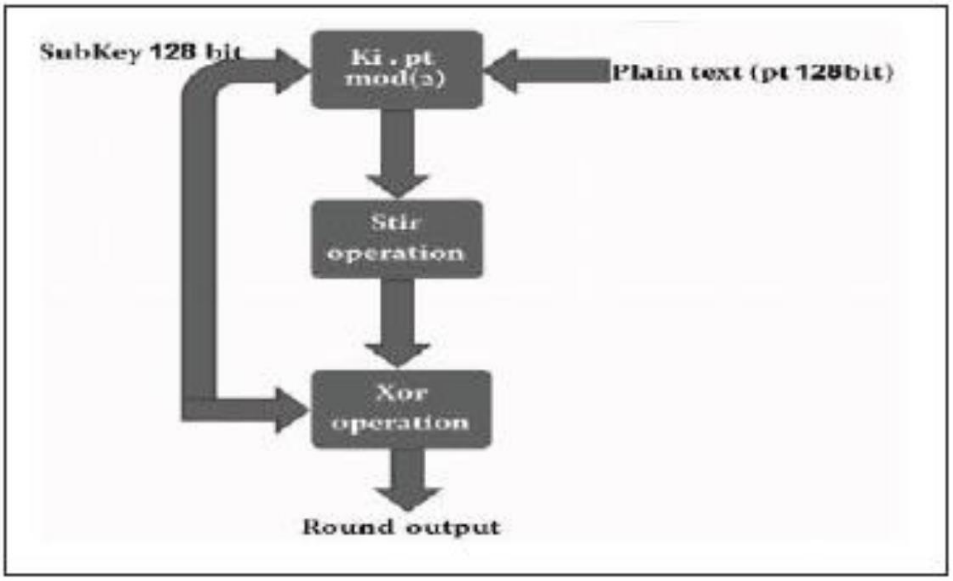
The 7th sub key k_7 is obtained by taking the bits from 31 to 0 and from 255 to 160.

$$k_7 = \text{key}(31 \text{ downto } 0)\&key(255 \text{ downto } 160) \qquad (10)$$

The 8th sub key k_8 is obtained by taking the bits from 159 to 32.

$$k_8 = \text{key}(159 \text{ downto } 32) \qquad (11)$$

c) Each round implementation:



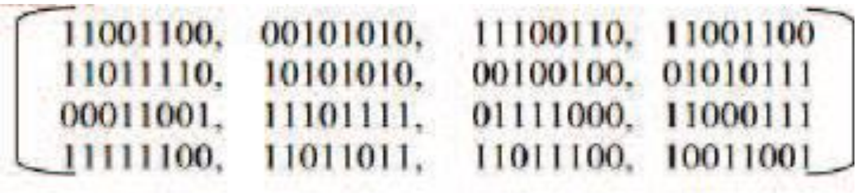
As in the fig 3, in each round the sub keys generated using the above algorithm is used crate a matrix and it is multiplied by the plain text.
The matrix multiplication is performed as shown in the figure below

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{21} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \text{mod}(26)$$

Fig4: Matrix multiplication

The above figure shows the matrix multiplication, which is the basic form of the hill cipher implementation. in the implementation AND is considered as multiplication and XOR is considered as the addition operation for the matrix multiplication implementation.

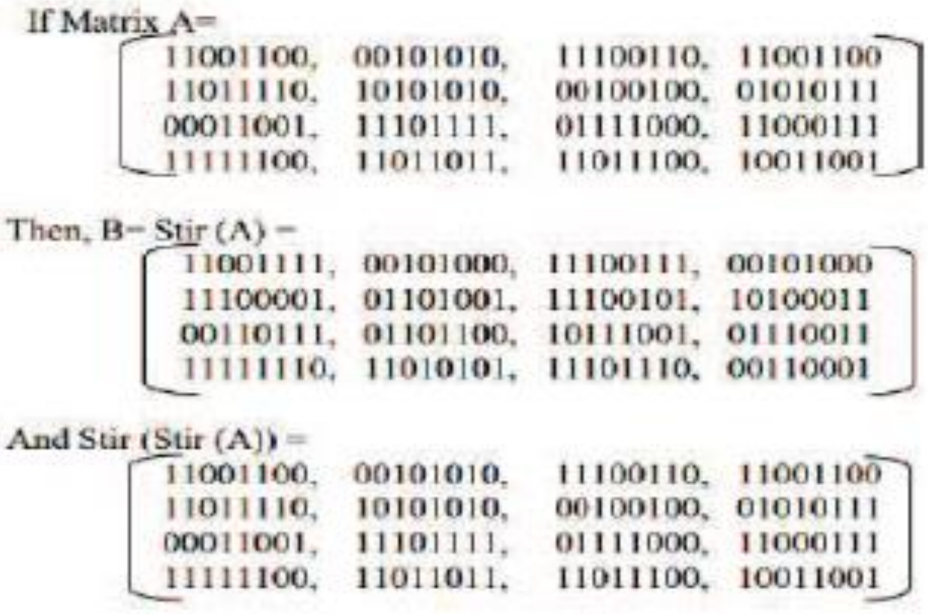
Then the output of the matrix multiplication is used to perform a reversible operation called as the STIR. To explain the stir operation, consider the following matrix



$$\begin{bmatrix} 11001100, & 00101010, & 11100110, & 11001100 \\ 11011110, & 10101010, & 00100100, & 01010111 \\ 00011001, & 11101111, & 01111000, & 11000111 \\ 11111100, & 11011011, & 11011100, & 10011001 \end{bmatrix}$$

- First byte is formed by selecting the first two bits of all the row elements.
- Second byte of each row is obtained by concatenating 3 and 4th bits all row elements.
- Third byte in each row is obtained by concatenating 5th and 6th bits of all row elements.
- Fourth byte in each row is obtained by concatenating 7th and 8th bits of all row elements.

Stir operation is a reversible operation. An example is shown in the below figure:



If Matrix A=

$$\begin{bmatrix} 11001100, & 00101010, & 11100110, & 11001100 \\ 11011110, & 10101010, & 00100100, & 01010111 \\ 00011001, & 11101111, & 01111000, & 11000111 \\ 11111100, & 11011011, & 11011100, & 10011001 \end{bmatrix}$$

Then, B= Stir (A) =

$$\begin{bmatrix} 11001111, & 00101000, & 11100111, & 00101000 \\ 11100001, & 01101001, & 11100101, & 10100011 \\ 00110111, & 01101100, & 10111001, & 01110011 \\ 11111110, & 11010101, & 11101110, & 00110001 \end{bmatrix}$$

And Stir (Stir (A)) =

$$\begin{bmatrix} 11001100, & 00101010, & 11100110, & 11001100 \\ 11011110, & 10101010, & 00100100, & 01010111 \\ 00011001, & 11101111, & 01111000, & 11000111 \\ 11111100, & 11011011, & 11011100, & 10011001 \end{bmatrix}$$

Fig5: Stir operation example

After obtaining the stir matrix, it is XORed with the sub key to get the that stage output. Thus, this is repeated for 8 times to get the first cipher text. Then this cipher text is used as the input to the next stage and the same operations are repeated to get the second cipher and finally the third cipher is obtained by repeating the same procedures for the cipher 2 obtained after the second iteration.

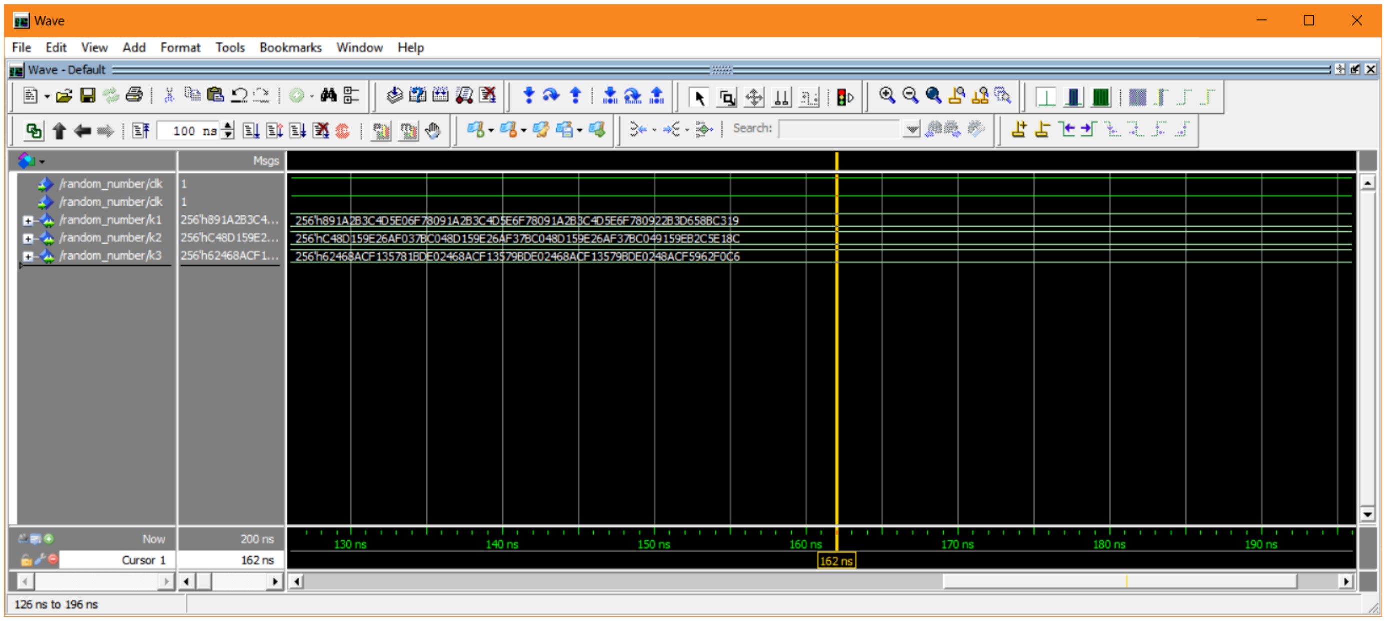
The three operations of the round is repeated for eight times with different sub keys. To perform the triple cipher this is performed three times.

The summary of the algorithm is as follows,
Encryption process:

- Read the plaintext and it is represented as 128-bit binary number in a 4X4 matrices.
- The 128-bit sub keys are generated from the 256-bit key obtained from the random number generator.
- Obtain the cipher text after performing the 8 stage operations three times.

3. Simulation and results:

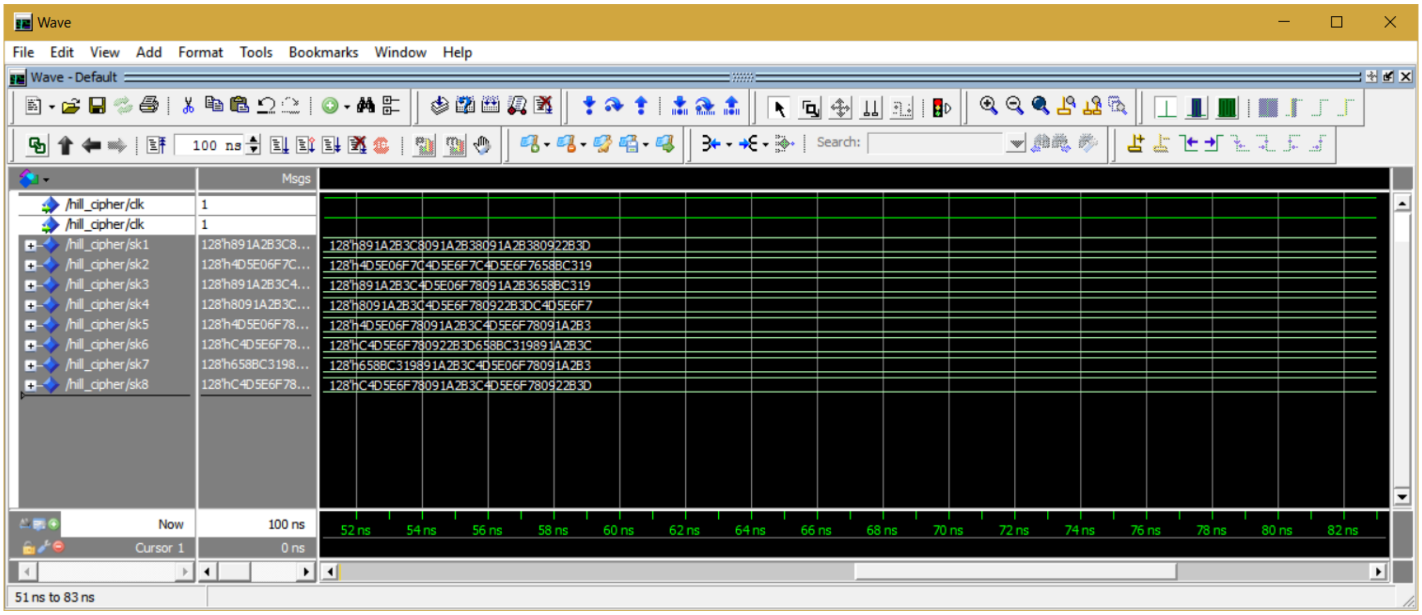
a) Simulation result of the random number generator:



```
/random_number/clk   1
/random_number/clk   1
/random_number/k1    256'h891A2B3C4...   256'h891A2B3C4D5E06F78091A2B3C4D5E6F78091A2B3C4D5E6F780922B3D658BC319
/random_number/k2    256'hC48D159E2...   256'hC48D159E26AF037BC048D159E26AF37BC048D159E26AF37BC049159EB2C5E18C
/random_number/k3    256'h62468ACF1...   256'h62468ACF135781BDE02468ACF13579BDE02468ACF13579BDE0248ACF5962F0C6
```

In the above figure it can be identified that the three keys are generated which are used to generate the cipher text at three stages.

b) Simulation results of the sub key generation:



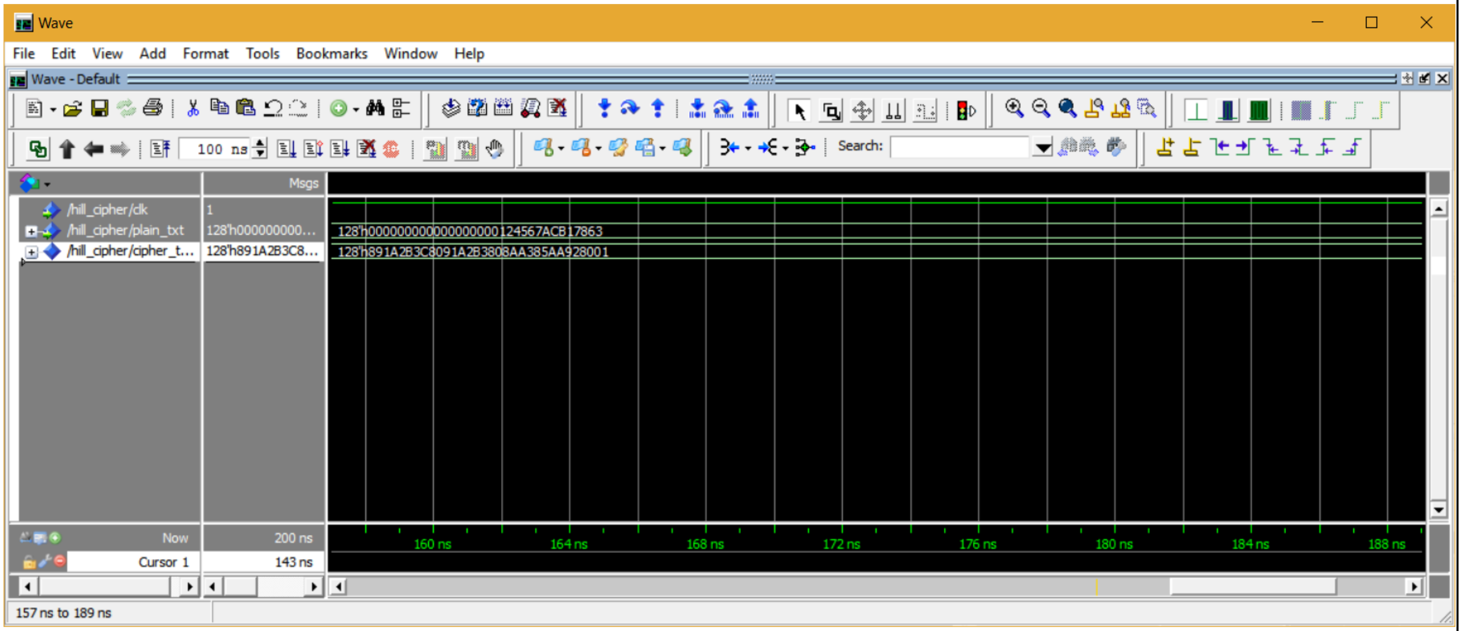
```
/hill_cipher/clk     1
/hill_cipher/clk     1
/hill_cipher/sk1     128'h891A2B3C8...   128'h891A2B3C8091A2B38091A2B380922B3D
/hill_cipher/sk2     128'h4D5E06F7C...   128'h4D5E06F7C4D5E6F7C4D5E6F7658BC319
/hill_cipher/sk3     128'h891A2B3C4...   128'h891A2B3C4D5E06F78091A2B3658BC319
/hill_cipher/sk4     128'h8091A2B3C...   128'h8091A2B3C4D5E6F780922B3DC4D5E6F7
/hill_cipher/sk5     128'h4D5E06F78...   128'h4D5E06F78091A2B3C4D5E6F78091A2B3
/hill_cipher/sk6     128'hC4D5E6F78...   128'hC4D5E6F780922B3D658BC319891A2B3C
/hill_cipher/sk7     128'h658BC3198...   128'h658BC319891A2B3C4D5E06F78091A2B3
/hill_cipher/sk8     128'hC4D5E6F78...   128'hC4D5E6F78091A2B3C4D5E6F780922B3D

Now       100 ns
Cursor 1  0 ns

51 ns to 83 ns
```

Form the above figure it can be identified that the eight different sub keys are generated.

c) Stage one output:

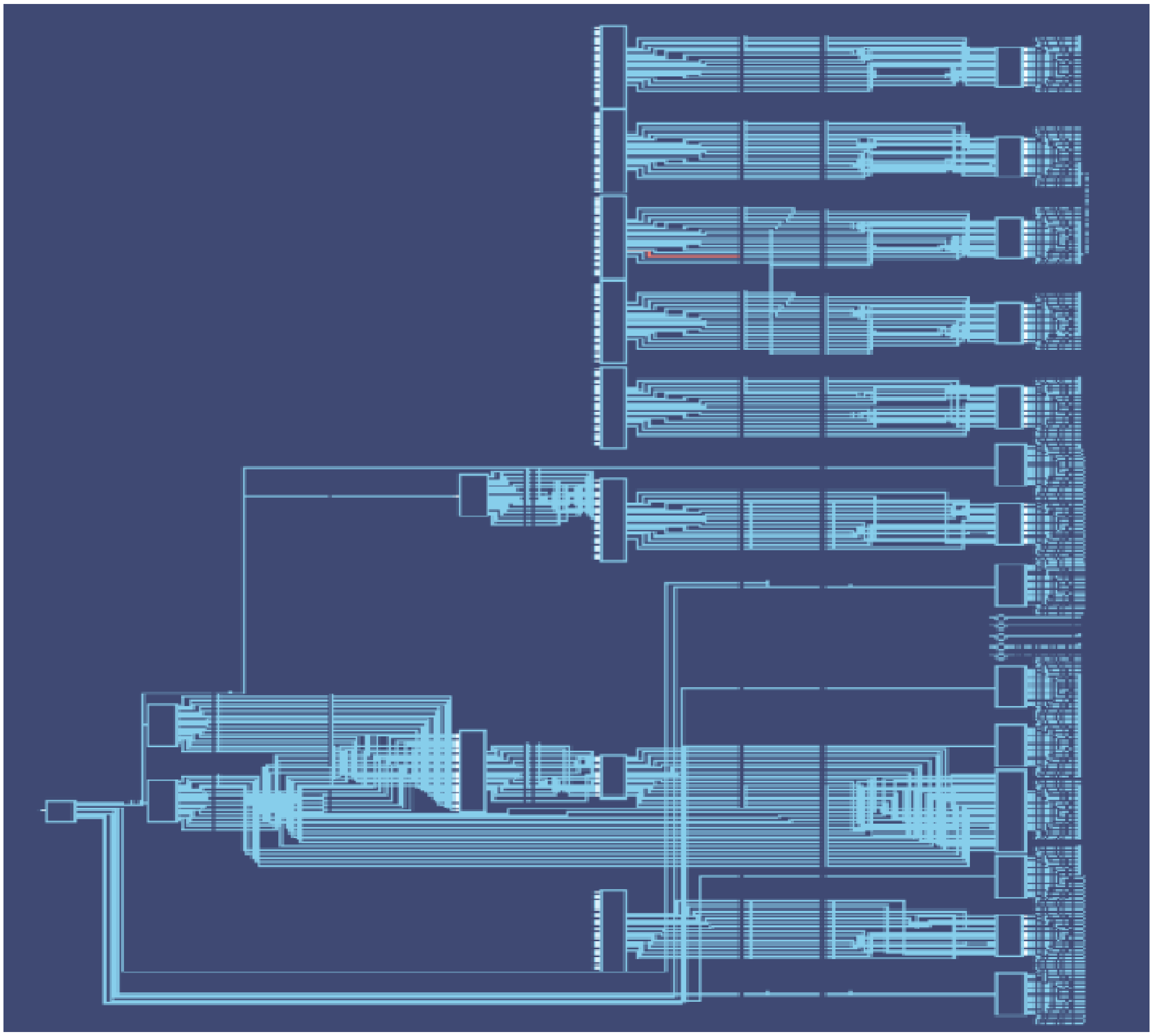


```
/hill_cipher/clk          1
/hill_cipher/plain_txt    128'h000000000...   128'h000000000000000000124567ACB17863
/hill_cipher/cipher_t...  128'h891A2B3C8...   128'h891A2B3C8091A2B3808AA385AA928001

Now       200 ns
Cursor 1  143 ns

157 ns to 189 ns
```

Data flow of the design:



4. Conclusion:

The algorithm discussed in the above is implemented in the VHDL and the output is verified. Since it requires a huge number of computations it can be implemented on the hardware to minimize the computation time and also to improve the performance of the algorithm. In the paper considered the only the encryption is considered and in the further work decryption can be implemented.

5. Reference

A Triple Hill Cipher Algorithm Proposed to Increase the Security of Encrypted Binary Dataand its Implementation Using FPGA Ashraf A.M. Khalaf, Mona S. Abd El-karim,Hesham F. A. Hamed
Department of Electronics & Communications Engineering, Faculty of Engineering, Minia University, Minia, Egypt.

A.F.A. Abidin, O.Y. Chuan and M.R.K. Ariffin A Novel Enhancement Technique of the Hill Cipher for Effective Cryptographic Purposes *Journal of Computer Science* 7 (5): 785-789, 2011.

Novel modified Hill cipher algorithm, Bhibhudendra Acharya, GirijaSankarRath, and Sarat Kumar Patra.

GandharbaSwain,andSaroj Kumar Lenka, A technique for secrect communication using new block cipher, *International Journal of Security and Its Applications*,Vol. 6, No. 2, April,2012.

Ahmed Desoky, AnjuPanickerMadhusoodhanan, *Bitwise Hill Crypto System*