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
ashkhalaf@yahoo.com,engmona1889@gmail.com,hfah66@yahoo.com

***Abstract*—Encrypted binary data security is an important task in the field of data communication systems since many decades. In this paper,we study the security problem and present a proposed triple hill cipher algorithm and its implementation on FPGA to encrypt any binary data such as images , audio , video … etc. The proposed algorithm uses three stages of a modified hill cipher to make the algorithm more robust and gives high level security of the data,each stage is considered a block cipher with a block length of 128 bits and key length of 256 bits. The message to be encrypted is processed by this block cipher in three stages. The keys are taken from random number generator. The proposed algorithm is promising to give better security.**

***Keyword*—hill cipher , 256 bit key ,cryptography, VHDL, FPGA.**

I. INTRODUCTION

Nowadayssecurity becomes an important feature with the growth of electronic communication systems. Cryptography is one of the methods used to protect data from unauthorized access and being stolen [1]. Cryptography is the science and study of secret writing. A cipher is a secret method of writing , whereby plaintext is transformed into cipher text .The process of transforming plaintext into cipher text is called encryption. The reverse process of transforming cipher text into plaintext is called decryption. Both encryption and decryption are controlled by a cryptographic key or keys [2][3].

There are two types of cryptosystem, which are symmetric cryptosystem and asymmetric cryptosystem. In Symmetric cryptosystem, the sender and recipient share the same key. It means the same key is used for encryption and decryption. In Asymmetric cryptosystem, different keys are used. A public key is used by sender to encrypt the message while the recipient used a private key to decrypt it [1][2].

In this paper we focus on hill cipher which is a type of symmetric cryptosystem.

The hill cipher was first described in 1929 by its inventor, the mathematician Lester S. Hill, in the journal of the american mathematical monthly (Eisenberg, 1998).[1][3][4]

The hill cipher is a classical symmetric cipher based on matrix transformation. It has several advantages including its resistance to frequency analysis and simplicity due to the fact that it uses matrix multiplication and inversion for encryption and decryption. However, it succumbs to the known plaintext attack [5] and as such there have been efforts to strengthen the cipher through the use of various techniques which have improved the security of the cipher quite significantly [6],[7],[8].

In this paper, we present a proposed *triple hill cipher algorithm* which consists of three stages of hill cipher ,each stage is considered a block cipher with a block length of 128 bits and key length of 256 bits. The message to be encrypted is processed by this block cipher in three stages to increase the security. The keys are taken from random number generator. Each stage consists of eight rounds with different eight keys , in each round three operations are implemented ; key and plaintext matrix multiplication , stir operation and XOR operation . this will be discussed in details in section III .we expect to achieve an algorithm more robust to cryptanalysis as we will use three layers of security.

The reason for using three stages is that, we used the concept of the common triple DES algorithm which is standardized by ANS X9.52 and is used to enhance the DES algorithm [12].

II. THE BASIC CONCEPT OF THE CLASSICAL HILL CIPHER

Hill Cipher was the first polygraphic cipher in which the key (K), plain text (P), and cipher text (C) are represented in the form of matrices. The basic method of encryption and decryption is represented by the following equations[9][10].

Manuscript received July 9 , 2015. This work was self-supported, and a follow-up of the invited journal to the accepted conference paper of the 17th International Conference on Advanced Communication Technology, and without Grants (Self -support).
Ashraf M. Khalaf is with the Faculty of Engineering, Department of Electrical Engineering. (Corresponding author, Phone: +20 86 2355261; fax: +20 86 2346674; e-mail: ashkhalaf@yahoo.com).
Mona S. Abd El-karim, is with the Faculty of Engineering, Departement of Electrical Engineering. (Phone: +201116123919; e-mail: engmona1889@gmail.com).
Hesham F. A. Hamedis with the Faculty of Engineering, Department of Electrical Engineering. (E-mail:hfah66@yahoo.com).

$$C = (K\,P) \bmod (26) \quad (1)$$

$$P = (K^{-1}\,C) \bmod (26) \quad (2)$$

where K^{-1} is the inverse of key matrix

Here modulo 26 arithmetic is used as the study was performed on the English alphabet, where each letter (Alphabet) is allotted a number generally starting from 0 in a continuous sequence one after the other as shown in Fig. 1 [3].

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Fig. 1 Alphabet Numbering

As shown in figure 1, Alphabets are numbered as A equals 0, B equals 1,... , and Z equals 25, but this is not a fixed requirement of the cipher. The encryption of plain text takes n successive plain text letters and substitutes them for n cipher text letters. In case n = 3, the encryption can be expressed in terms of the matrix multiplication as follows[3]

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{21} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \bmod(26) \quad (3)$$

III. THE PROPOSED TRIPLE HILL CIPHER ALGORITHM

As shown in Fig. 2 the proposed Triple Hill Cipher algorithm consists of three stages ,where plaintext is encrypted three times using three different 256 bits keys.

Keying options :

- Option 1: k_1 ,k_2 and k_3 are different (not equal).
- Option 2: k_1 ,k_2 are different and k_3 equal to k_1, this option is considered double encryption ,but it is stronger than simply hill cipher encrypting twice, e.g with k_1 and k_2 because it protects against meet in the middle attack[13],
- Option 3: k_1 ,k_2 and k_3 are the same, this option is considered the least secure one.

In our proposed algorithm we used the first option 1, the key of the first stage is taken from random number generator ,then 1st key is rotated one time to get the 2nd key and two times to get the 3rd key or we can take the three keys from different generators.

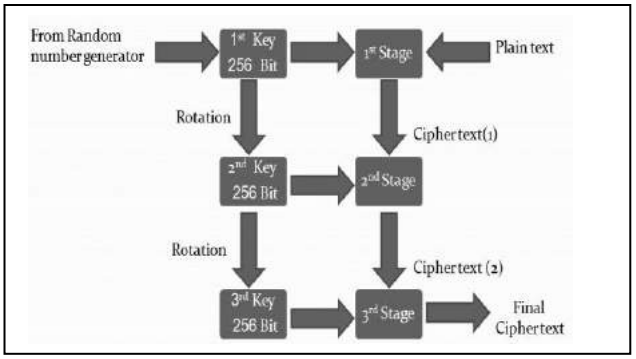

Fig. 2 The Proposed Triple Hill Cipher Algorithm

Random number generators (RNGs)

Is a computational or physical devices designed to generate a random sequence of numbers or symbols. There are two types of these generators:

- True Random Number Generators (TRNGs) which divided into two categories physical and nonphysical TRNGs[14]. Physical TRNGs use nondeterministic effects of electronic circuits such as shot noise from zener diode, inherent semiconductors thermal noise, and free running oscillators. They produce a truly random numbers. Their outputs depend only on physical or nonphysical process not on any initial value.
- Pseudo Random Number Generators (PRNGs) is considered an algorithm to generate a sequence of numbers that appear random. The sequence is not truly random in which it is completely determined by an initial value called a seed. There are several techniques used to perform PRNGs such as Linear Feedback Shift register, Linear Congruential Generator and Blum BlumShub[15].

In this paper we will use PRNG with the Linear Feedback Shift register (LFSR).

Linear Feedback Shift Register (LFSR)

A LFSR is made of sequential shift-register with combinational feedback logic connected to it which can generate a sequence of binary values in a pseudo-random manner.

Feedbacks around an LFSR's shift register are connected to the certain points (taps) of LFSR construction and constitute either XORing or XNORing these taps to provide taps back into the register.

The selection of taps determines how many values can be generated in a given sequence before the sequence is repeated. A certain tap arrangements lead to amaximal length sequences of $(2^n - 1)$. These settings are calculated for different lengths of LFSRs and are represented in[16].

In our algorithm, we need a 256 bit LFSR to get pseudo random keys to enhance the security of the proposed algorithm. Our cipher is symmetric (sender and receiver share the same key) so when we use random keys at transmitter we need to synchronize these keys with the receiver's keys, we can do that by using the same seed at transmitter and receiver.Here we will explain four-bit LFSR as example:

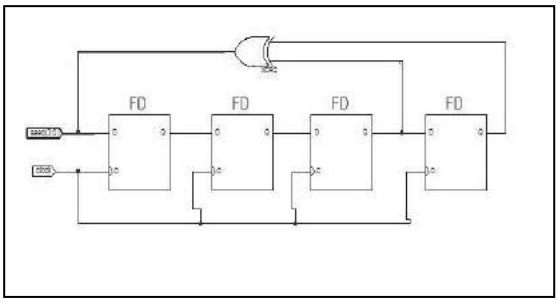

Fig. 3 Four-bit LFSR

As shown in Fig. 3, four-bit LFSR consists of four D-flip flops and xor gate in its shift path. The feedback taps are selected from taps 3 and 4. When LFSR seed is loaded to the four flip flops ,LFSR generates random numbers startingfrom an initial value selected by the user.

A. Steps of the proposed algorithm

Each stage in this algorithm includes eight rounds and sub keys generation module as shown in Fig. 4.

For the first stage ; plaintext is divided into blocks of lengths 128 bits ,each block is represented in 4 by 4 byte matrix .

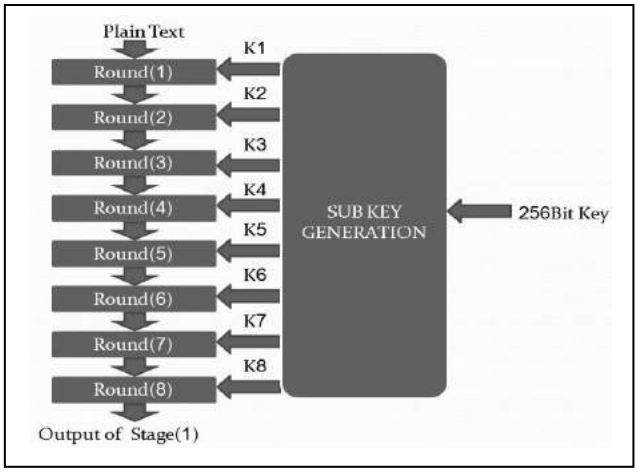

Fig. 4 Single Stage Diagram

Sub keys generation:

128 bits sub keys are generated from the 256 bits key as follow :

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.

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

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 .

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

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

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

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

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

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

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

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

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

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

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

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

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

Each of k_1 ,k_2 , k_3 ,k_4 ,k_5 ,k_6 ,k_7 and $k_8$is then represented in 4 by 4 byte matrices .

B. Operations in each round

Each round includes three operations as shown in Fig. 5.

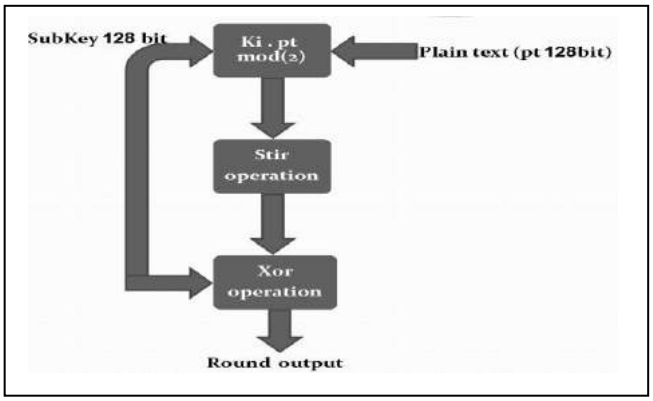

Fig. 5 Single Round Diagram

Operation 1 represents matrix multiplication of plaintext (pt) and sub keys (k_i), where k_i represents k_1 ,k_2 , k_3 ,k_4 ,k_5 ,k_6 ,k_7 and k_8.Here we encrypt any binary data ,so we use modulo 2 field , when values in the matrices are multiplied, bitwise AND is used and when values are added bitwise XOR is used[10][11].

At decryption process the same operation is performed but instead we use the inverse matrices of the sub keys.

Operation 2 represents the Stir operation

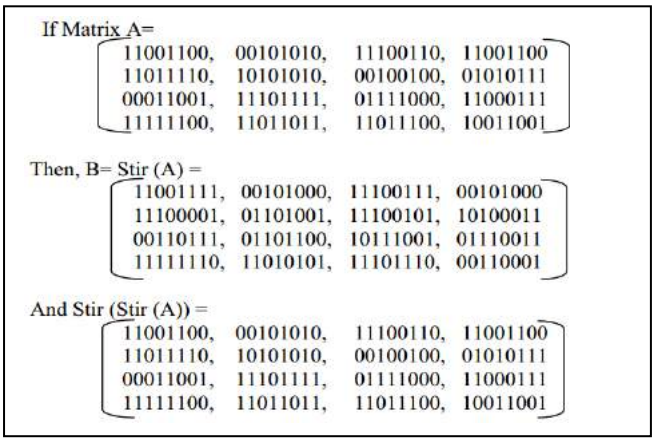

Fig. 6 Stir operation

As shown in Fig. 6, the stir operation is defined by the following steps:

- The 1st and 2nd bits from each byte in a row of A are combined to form the first byte of B in that row.
- The 3rd and 4th bits from each byte in a row of A are combined to form next byte of B in that row.
- The 5th and 6th bits from each byte in a row of A are combined to form next byte of B in that row.

- The 7th and 8th bits from each byte in a row of A are combined to form the last byte of B in that row.
- This stir operation is reversible, i.e. Stir(Stir(A))=A[9].

Operation 3 represents XOR operation
We perform XOR between sub keys k_i and the output of stir operation .it is performed as bit by bit XOR ,as example if we have $M = 11110000$ and $L = 00110011$ then $XOR(M, L) = 11000011$
XOR operation is reversible if $N = XOR(M, L)$
then$L = XOR(N, M)$ and $M = XOR(N, L)$.

The three operations of the round are repeated eight times with different sub keys for k_i=1:8 to perform one stage of the Triple Hill Cipher , so to perform the triple hill cipher we repeat the stage three times ,so we can achieve an algorithm more robust to cryptanalysis.

Summary of The Proposed Triple Hill Cipher Algorithm

i. Encryption process

1. Read the message (plaintext) as a binary and divide it into blocks of lengths 128 bits ,then is represented in 4 by 4 byte matrices
2. 128 bits sub keys matrices (k_i =1:8) are generated from 256 bits key
3. Find inverse of the sub keys matrices
4. If the matrices are noninvertiblechange the key and go to step 2
5.

$for\ m = 1:3$
{
$for\ i = 1:8$
{
$p_{m1} = (k_i . pt)\ mod(2)$
$p_{m2} = stir(p_{m1})$
$p_{m3} = XOR(k_i, p_{m2})$
$pt = p_{m3}$
}
}
$c = pt$

Where pt is the plain text ,c is the cipher text ,m is the number of stages and i is the number of rounds.

i. Decryption process

1. Prepare the inverse of the sub keys matrices k_i^{-1}.
2.

$for\ m = 1:3$
{
$for\ i = 1:8$
{
$c_{m1} = XOR(k_i, c)$
$c_{m2} = stir(c_{m1})$
$c_{m3} = (k_i^{-1} . c_{m2})\ mod(2)$
$c = c_{m3}$
}
}
$pt = c$

IV. SIMULATION AND IMPLEMENTATION RESULTS

A. Simulation results

The proposed algorithm is coded using VHDL language and simulation results taken from Modelsim6.0C as sown in the following figures:

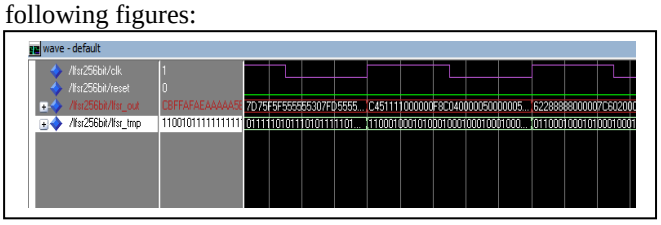

Fig. 7 Timing Simulation of The 256 bit LFSR

Fig. 7 shows timing simulation of the 256 bit LFSR which works as a pseudo random number generator ,as shown in figure in each clock cycle different bits are generated.

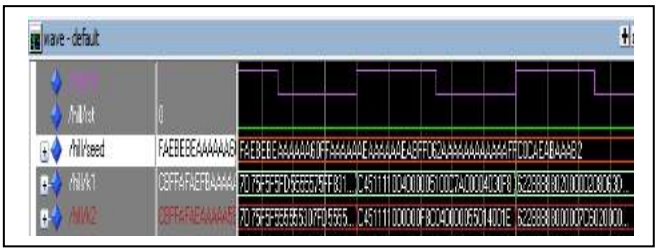

Fig. 8 Timing Simulation of the sub keys generation

Fig. 8 shows Timing simulation of sub keys generation process from the 256 bit key , as shown in figure in each clock cycle k_1 ,k_2 , k_3 ,k_4 ,k_5 ,k_6 ,k_7 and k_8 are generated with different values depending on the value of the 256 bit key used.

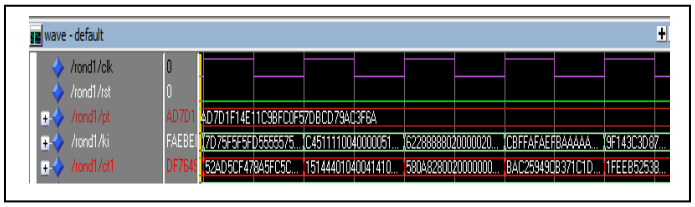

Fig.9 Timing simulation of one round

Fig. 9 shows timing simulation of one round ,wherept is the plain text , $k_i$is the 128 bit sub key and ct is the cipher text from one round ,as shown we have ct with different values and this process is repeated eight times in each stage.

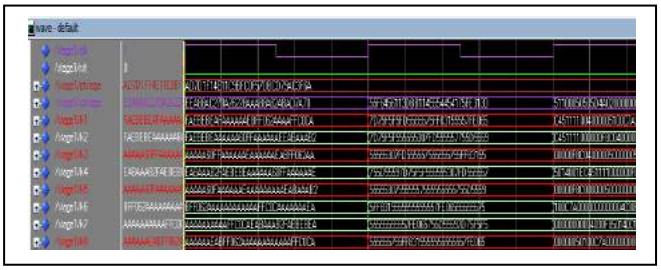

Fig.10 Timing Simulation of one stage

Fig. 10 shows the timing simulation of one stage from the encryption process, where ptstage is the plain text and ctstage is the cipher text from one stage, also we note the contents of the cipher text changes in each clock cycle and this process is repeated three times.

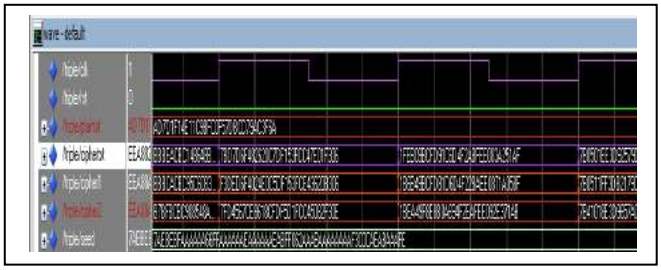

Fig. 11 Timing Simulation of the overall Proposed Triple Hill Cipher

Fig. 11 shows the timing simulation of the overall proposed algorithm, where plaintext is the input message needed to be encrypted, seed is the initial value of the pseudo random number generator to get the random 256 bit keys, cipher1 is the cipher text from the first stage,cipher2 is the cipher text from the second stage and ciphertext is the final cipher text which is the output of the third stage

Because of using random number generator to get keys we get cipher text with different values in each clock cycle which enhance encryption process and make the attacker's task more difficult.

We used this data during the simulation process,here data is arranged as vectors of bits.
256 bits first key:
1111101011101011111010111110101010101010101010101010
1010011000001111111110101010101010101010101010101110
1010101010101010101010101011101010101111111111110000
0110001010101010101010101010101010101010101010101010
1010101011111111110000001100101011101010101110101010
1010101010110010

Plaintext:
1111111111010101010101010101010101010101010101010101
0100000000001111111111010101010110101010100011000
1100110101010101010101010101010101

The outputs of the three stages is
Cipher text1:
1110111010101000100010001100001001010100101000100
0110001011101000101010101010100010001000100000010
1010111110101101111100101011000 0
Cipher text 2:
0101010011110010010001000110000100000000111011111
0011000101010100010101000101000001000100010000001
0111010011111010001100010101000 0
Cipher text 3 which also is the Final Cipher text of the proposed algorithm :
0010101001111011001010000011001010000011010001111
1001101011101010001010100010101000010100000100010
1001000001111111001100100011000 0
As shown from the simulation results,plain text is encrypted three times ,which proves that our proposed algorithm increases the difficulty in cryptanalysis .

B. Implementation results

FPGA implementation of the proposed algorithm was accomplished on Space-Grade Virtex-4QV XQR4VSX55-10CF1140 using Xilinx ISE Design Suite 13.2 as synthesis tool.The top level RTL schematic of the proposed algorithm as shown in figures is given to establish the fact that the HDL codes are synthesizable.

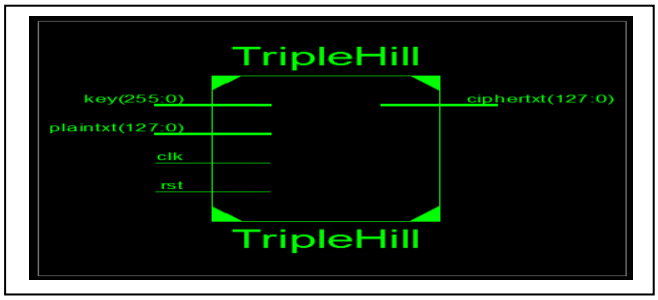

Fig. 12. Top Level RTL schematic of the proposed algorithm

Fig. 12 represents the complete hardware implementation of the proposed algorithm, as shown the data (plaintxt) is 128 bit which is ciphered using a 256 bit key. The rst bit is used to reset the module and clk bit is used to clock the design. The output is the 128 bit cipher text.

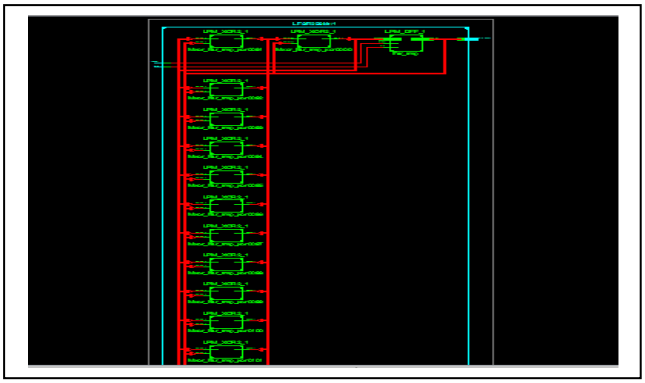

Fig. 13 Top Level RTL schematic of the 256 bit LFSR

Fig. 13 shows the components (actually it is a part of the components because we can't attach all components in the figure) of the 256 bit LFSR which acts as random number generator to give us the random 256 bit keys.

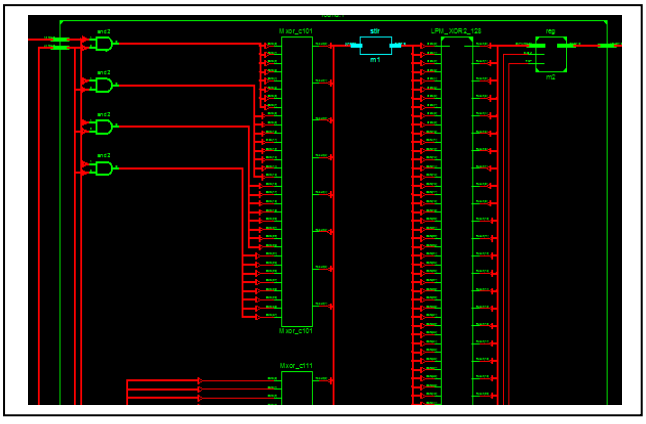

Fig. 14 Top Level RTL schematic of one Round

Fig. 14 shows the components of one round which consists of three operations; matrix multiplication which is the first part in the figure from the left, stir operation the second part and the xor operation the third part (also it is a part of the components because we can't attach all components in the figure).

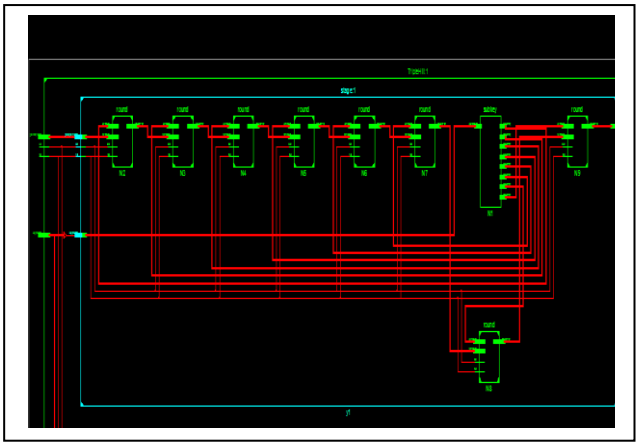

Fig. 15 Top Level RTL schematic of one Stage

Fig. 15 shows the components of one stage which consists of eight rounds and sub keys generation module which is the second part in the figure from the right.

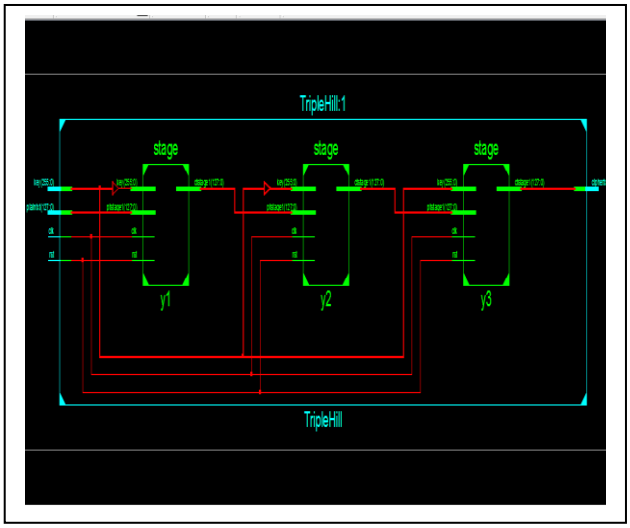

Fig. 16 Top Level RTL schematic of The overall triple Hill cipher

Fig. 16 shows the complete hardware of the proposed algorithm.

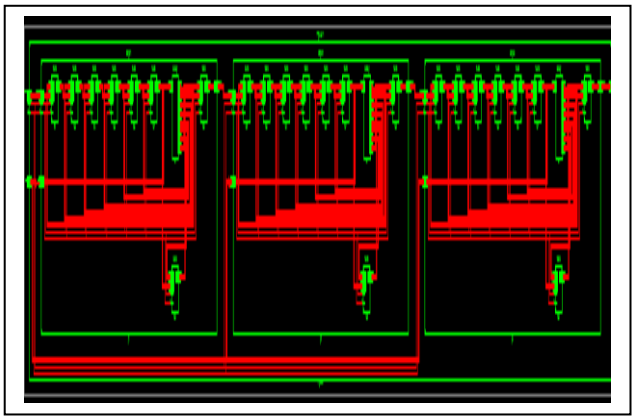

Fig. 17 Top Level RTL schematic of The overall triple Hill cipher cont.

Fig. 17 also shows the complete hardware of the proposed algorithm but with more details.

Design summary for the proposed algorithm is specified in Table I to set up details .

TABLE I
Design summary of the proposed algorithm

Logic Utilization	*Used*	*Available*	*Utilization*
Number of Slice Flip Flop	*3,072*	*49,152*	*6%*
Number of 4input LUTs	*9,222*	*49,152*	*18%*
Number of occupied slices	*4,636*	*24,576*	*18%*
Number of bonded IOBs	*514*	*640*	*80%*
Number of BUFG/BUFGCTRLs	*1*	*32*	*3%*
Average Fanout of Non-clock Nets	*3,77*		

Design summary in Table Igives the number of slice flip flops, number of 4 input LUT (Look Up Tables), number of BUFG/BUFGCTRLS, number of bonded IOBs and average fanout of non-clock network that used from the FPGA kit to implement the hardware of our design.

Power analysis of the proposed algorithm design

The total power dissipation is the sum of two types of power ;

- Quiescent (static) power is defined as the product of the power supply voltage and static current, which itself has two dual components: leakage currentand through current. Leakage currents are parasitic effects and are small in magnitude Through currents occur in normal operation and are due to transistors being continuously operated in their saturation region.
- Dynamic power which is frequency dependent and it has two components: the “capacitive” load power and the cell power. The latter is consumed internally bythe cell primitives. This component accounts for the power that is required to mainly charge and discharge the internal cell capacitance. “Capacitive” load power represents the currents required to charge the external loads driven by each cell. The overall dynamic power for an entire chip is given by [17]

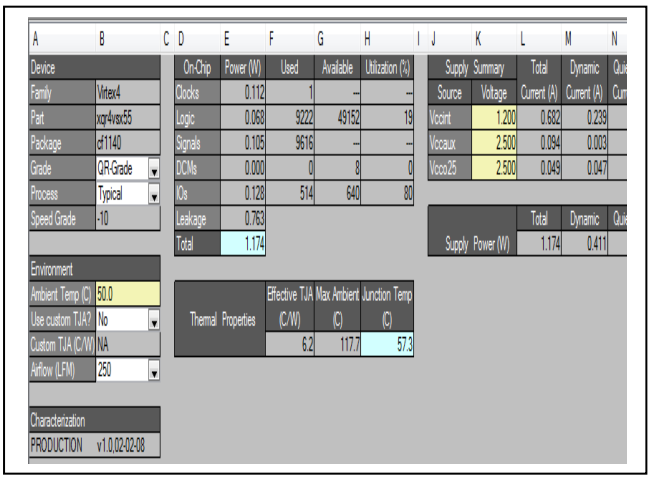

Fig. 18 Power analysis summary of the design with Clock frequency 25 MHZ

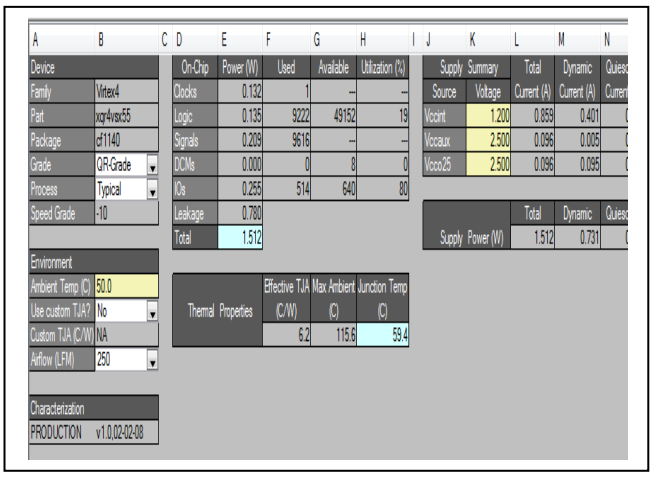

Fig.19 Power analysis summary of the design with Clock frequency 50 MHZ

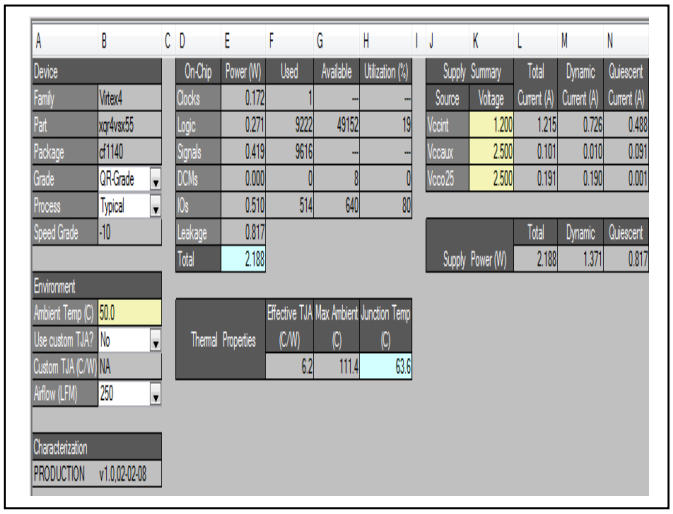

Fig. 20 Power analysis summary of the design with Clock frequency 100MHZ

Figs 18, 19 and 20 specifies the power analysis summary of the design at frequencies 25, 50 and 100 MHZ. As shown in figure the quiescent power is approximately constant at different frequencies, but the dynamic power changes with the frequency it increases as the frequency increases.

It is clear from these figures that the dynamic power increases as the frequency increases, so we need to know the maximum allowed frequency of this design. Here is the timing summary of that design, that from it we can know the maximum frequency:

- Minimum input arrival time before clock: 4.263ns
- Maximum output required time after clock: 4.677ns
- Minimum period: 1.894ns (Maximum Frequency: 528.067MHz)

Power analysis summary of the design at the maximum allowed clock frequency will be shown in Fig. 21

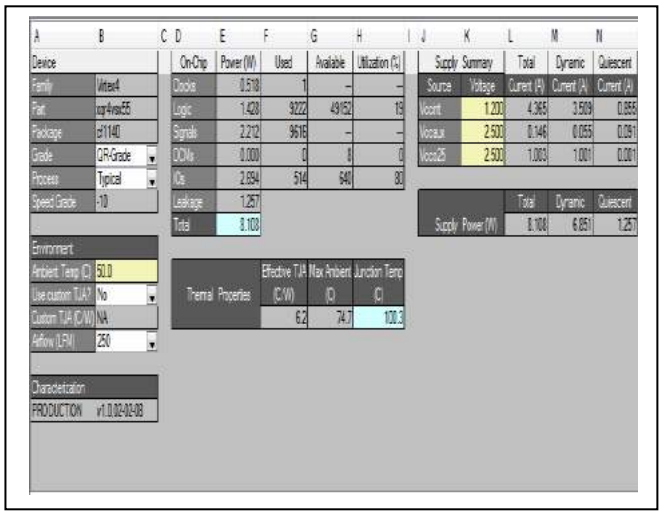

Fig. 21 Power analysis summary of the design with Clock frequency 528.067 MHZ

V. CONCLUSION

In the cryptanalysis of the classical hill cipher ,the known plaintext attack is performed using the method $k = p^{-1} c$ where p is an invertible matrix , so if an attacker has m distinct plaintext and cipher text ,can retrieve the key using the previous method .In the proposed triple Hill Cipher algorithm the known plain text attack becomes more difficult as the plain text is encrypted in eight rounds with eight different keys three times ,we can say that , we used 24 different keys which makes the task of the known plain text attack more difficult, there is no doubt that increasing the number of stages for example four stages instead of three increases the level of security but the overall performance of the algorithm become more slower. Also using LFSR as a random number generator to get random keys enhances the security of the proposed algorithm as the keys change more times.

REFERENCES

[1]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.

[2] William Stallings ,''*Cryptography and Network Security Principles and Practices*", Fourth Edition, Prentice Hall, November 16, 2005.

[3] Jasdeep Singh Bhalla,'' A Database Encryption Technique to Enhance Security Using Hill Cipher Algorithm", *International Journal of Engineering and Advanced Technology* (IJEAT), Vol. 2, No. 4, April 2013.

[4] M. Nordin A. Rahman, A. F. A. Abidin, MohdKamirYusof, N. S. M. Usop,'' Cryptography: A New Approach of Classical Hill Cipher", *International Journal of Security and Its Applications*,Vol. 7, No. 2, March, 2013.

[5]D.R. Stinson, "*Cryptography Theory and Practice*",Third Edition, Chapman and Hall/CRC, Pp.13-37, 2006.

[6] V. U. K. Sastry, D. S. R. Murthy, S. DurgaBhavani, "A Block Cipher Involving a Key Applied on Both the Sides of the Plain Text," *International Journal of Computer and Network Security* (IJCNS), Vol. 1, No. 1, Pp. 27 -30, Oct. 2009.

[7] V. U. K. Sastry, V. Janaki, "A Modified Hill Cipher with Multiple Keys", *International Journal of Computational Science*, Vol. 2, No. 6, 815-826, Dec. 2008.

[8] Bhibhudendra Acharya, GirijaSankarRath, and Sarat Kumar Patra, "Novel Modified Hill Cipher Algorithm,",*Proceedings of ICTAETS*, Pp. 126-130, 2008.

[9]GandharbaSwain,andSaroj Kumar Lenka,''A Technique for Secret Communication Using a New Block Cipher with Dynamic Steganography'', *International Journal of Security and Its Applications*,Vol. 6, No. 2, April, 2012.

[10] Ahmed Desoky, AnjuPanickerMadhusoodhanan,''*Bitwise Hill Crypto System*'',
DOI: 10.1109/ISSPIT.2011.6151539
[11] Ali Muhammad Ali Rushdi and Fares Ahmad Muhammad Ghaleb,''On Self-Inverse Binary Matrices Over the Binary Galois Field'',*Journal of Mathematics and Statistics* 9 (3): 238-248, 2013.
[12] D. Coppersmith, D. B. Johnson and S. M. Matyas, ''A proposed mode for triple-DES encryption",*IBM J. RES, DEVELOP*. VOL. 40 NO, 2 MARCH 1996.
[13] Ralph C. MerkleElxsi, ''On the Security of Multiple Encryption",*Technical Note Programming Techniques and Data Structures D. Mcllroy Editor, Int. Martin E. Hellman Stanford University, Pp. 465-467,* Volume 24 the ACM Number 7, July,*1981*.
[14]C¸ .K. Koc¸ (ed.) " *Cryptographic Engineering",* DOI 10.1007/978-0-387-71817-0 3, c Springer Science+Business Media, LLC 2009
[15] Jay Kumar, Sudhanshu Shukla, Dhiraj Prakash, Pratyush Mishra and Sudhir Kumar,'' Random Number Generator Using Various Techniques through VHDL'',*International Journal of Computer Applications in Engineering Sciences* ,VOL I, ISSUE II, JUNE 2011 , ISSN: 2231-4946.
[16] Douglas, J.S., 1997, "*HDL Chip Design*" 3rd, Doone Publications, Madison, AL, USA, ISBN 0-9651934-3-8, , pp. 179-187.
[17] HichemBelhadj, BehroozZahiri, Albert Tai and Actel Corporation,''Power-sensitive designtechniques on FPGA devices'', *International IC – Taipei Conference Proceedings*.http://www.eetasia.com.sci-hub.org/ARTICLES/2001JUL/2001JUL03_PL_POW_TAC.PDF.

Ashraf A. M. Khalaf (M'98) received his B.Sc. and M.sc. degrees in electrical engineering from Minia university, Egypt, in 1989 and 1994 respectively. He received his Ph.D in electrical engineering from Graduate School of Natural Science and Technology, Kanazawa university, Japan, in Marsh, 2000. He is currently works as an associate professor at electronics and communications engineering Department, Minia University, Egypt.. His research interest includes digital signal processing and its applications in communications, neural networks, and optical communications.

Mona S. Abd El-karimwas Born : 1-8-1989, she worka as a teaching assistant and is currently a master course student for M.Sc. degree in Electrical Engineering (Communication and Electronics), Faculty of Engineering, Minia University, El-Minia, Egypt.

Hesham F.A. Hamed received the B.Sc. degree in Electrical Engineering, the M.Sc. and Ph.D. degrees in Electronics and Communications Engineering from EL-Minia University, ELMinia, Egypt, in 1989, 1993, and 1997 respectively. He currently is Professor and a Dean of the faculty of Engineering EL-Minia University. From 1989 to 1993 he worked as a Teacher Assistant in the Electrical Engineering Department, ELMinia University. From 1993 to 1995, he was a visiting scholar at Cairo University, Cairo, Egypt. From 1995to 1997, he was a visiting scholar at Texas A&M University, College Station, Texas (with the group of VLSI). From 1997 to 2003, he was anAssistant Professor in the Electrical Engineering Department, EL-Minia University. From 2003 to 2005, he was Associate Professor in the sameUniversity. From 2005 to 2007, he was a Visiting Researcher at Ohio University, Athens, Ohio. He has published more than 65 papers and onebook chapter. His research interests include analog and mixed-mode circuit design, low voltage low power analog circuits, current mode circuits,nano- scale analog and digital integrated circuits design, and FPGA.