IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 05, Issue 01 (January. 2015), ||V2|| PP 45-50

Optimized FPGA Hardware Encryption Implementation using Public Key Cryptography

Murtada Mohamed Abdelwahab[1], Elsanosy M. Elamin[2],
Abdul Rasoul Jabar Alzubaidi[3]
[1]*Department of Electronic Engineering - Faculty of Engineering & Technology - University of Gezira*
[2]*Electrical Engineering Department, faculty of Engineering, University of Kordofan , Sudan*
[3]*Electronic Dept, Engineering Collage -Sudan University of Science and echnology,rasoul46@live.com*

Abstract: - Public key algorithm is a popular algorithm that used to provide a secure transmission of information .The proposed implementation aims to provide a high level of security and correctness. The presented implementation is a public key algorithm based on Field Programmable Gate Array (FPGA). The design needs two keys to perform the process of decryption, one of the keys is a private key. The design is optimized in terms of the device hardware consumption compared to other related work shown in this paper. The targeted device is Xilinx Virtex 5. The obtained simulation results were correct and reliable. The results show that the implementation has an efficient utilization of the available resources of FPGA.

Keyword: - FPGA, Encryption, Decryption, Private Key

I. INTRODUCTION

The proposed work is classified as asymmetric encryption implementation. This design aimed to provide secure data transfer. It is based on the use of the private key property. Unlike symmetric algorithm , the private key algorithm does not needs a complex computations to ensure the privacy of information. Instead of using a hard computations, it used two keys one for all users which can be called as public key and the other is used only by receiver called private key.In general encryption algorithms are divided into two categories:

- Symmetric key algorithms where the same key is used for encryption and decryption.
- Asymmetric key algorithms (Public-key cryptography), where two different keys are used for encryption and decryption.

The advantage of asymmetric over symmetric encryption is that the public key does not have the risk that its interception will compromise encrypted data. The basic types of asymmetric algorithm are:

- Public key : the private key is owned by the receiver.
- Digital signature: the private key is owned by the sender.

II. MATERIALS AND METHODS

Xilinx –project navigator, ISE 9.2i is the computer aided tool which used to create the model design, synthesizing and implementing (i.e Translate Map & Place and Route) VHDL code with FPGA device.
The implementation as shown in figure(1) consists of two keys one is public key used by all users to encrypt a block of plaintext and also needed to decrypt cipher input into plaintext, the other key is a private key used only by receiver and they are both consists of 64-bit.

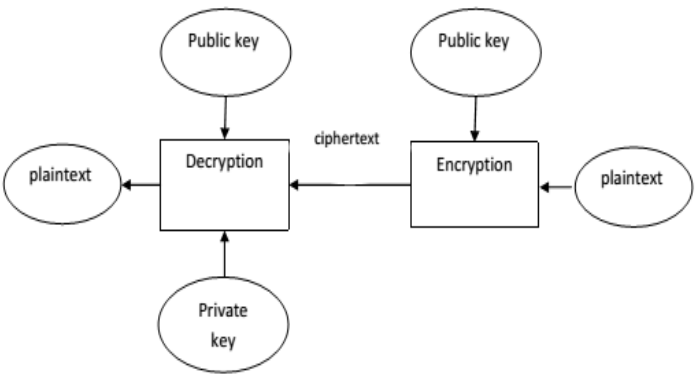

Fig.1: Operation method

The implementation algorithm is as other popular encryption algorithms composed of a combination of substitution and transposition encryption technique. The map of transposition is described in figure(2). Each byte changes position following the illustrated path in the chart.

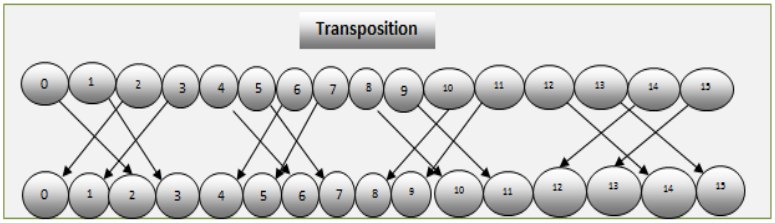

Fig.2. Transposition Mechanism

A. Encryption Algorithm:

The algorithm of encryption as shown in figure(3).The algorithm accepts a block of 128 bit and used a public key of 64 bit .It includes the following stages:

-**SubBytes**: where each block of data is divided into a group of bytes. The block consists of 128 bit and it can be considered as a set of array elements.

$$\text{S box} = \begin{pmatrix} B_0 & B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 & B_7 \\ B_8 & B_9 & B_{10} & B_{11} \\ B_{12} & B_{13} & B_{14} & B_{15} \end{pmatrix} \quad (1)$$

- **Transposition**: is the technique that used to change the positions of the input block elements. By using the chart in figure 1, a new arrangement of the s box elements is performed and it can be expressed as:

$$\text{S box} = \begin{pmatrix} B_2 & B_3 & B_0 & B_1 \\ B_6 & B_7 & B_4 & B_5 \\ B_{10} & B_{11} & B_8 & B_9 \\ B_{14} & B_{15} & B_{12} & B_{13} \end{pmatrix} \quad (2)$$

- **Addkey**: only public key is added at this stage. The key is added in two steps: a subbytes step and transposition step.

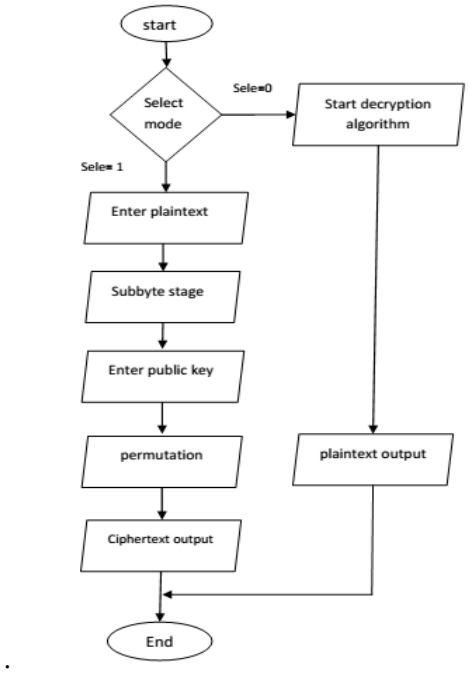

Fig.3. Encryption Architecture

The output cipher computations is described in figure(4).The output cipher is composed of three blocks.

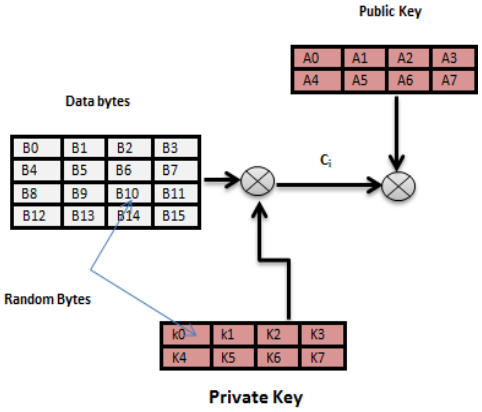

Fig. (4) Encoding process

The calculations are summarized in the following equations:

$$\mathbf{C_i = (B_i} \oplus \mathrm{K_i}) \qquad \mathbf{(3)}$$

$\mathbf{C}_i =$

Where $K_i$is theprivate key elemnts and B_i is therearranginging elemnts of s box .once the block of data is entered then C_i is composed directly inside the algorithm by adding (xor) the s box elements with the private key elements and it does not need any entering of private key by users. The final output cipher **C** is calculated by adding (xor) the public key$\mathbf{A_i}$ with $\mathbf{C_i}$.

$$\mathbf{C} = (\mathbf{C_i} \oplus \mathbf{A_j}) \qquad \mathbf{(4)}$$

B. Decryption Algorithm:

Decryption process performs using the reverse steps for the encryption algorithm. The algorithm simply can perform by using a reverse transposition for the input elements of cipher and keys. The two keys (private and public) must be added together in a correct form to decode the input cipher. It is not necessary to use any particular order for adding the two keys. Decryption algorithm is described in figure (5).The output calculation is expressed as:

$$\mathbf{P} = \mathbf{K_i} \oplus \mathbf{A} \oplus \mathbf{C} \qquad \mathbf{(5)}$$

Where P is the output plaintext .A_i is the public key. K_i is a private key used only in decryption stage.C is the input block cipher.

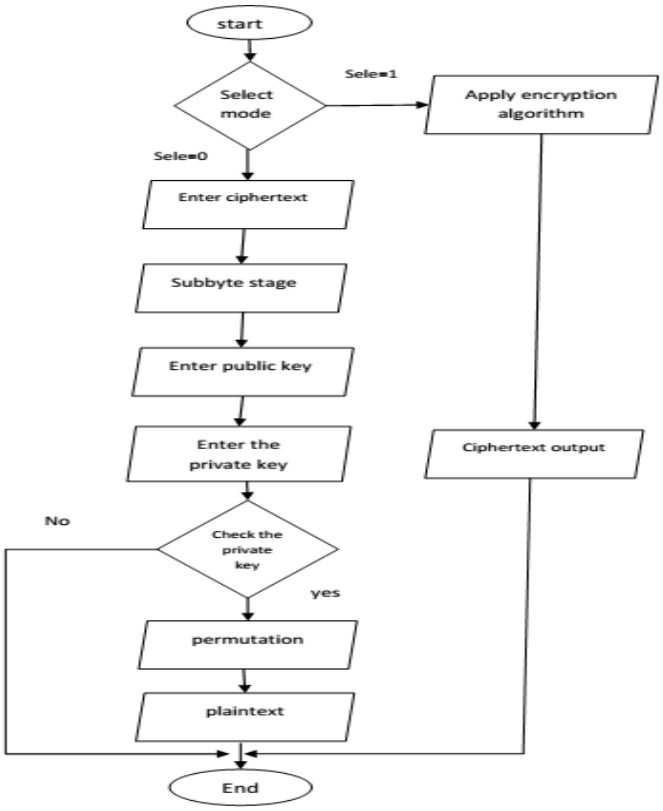

Fig .5. Decryption Architecture

III. RESULTS and DISCUSSIONS

When it comes to keeping data private, nothing beats protection with a strong encryption algorithm. The private key algorithm provides a high level of assurance. The purpose of simulation is to verify the correctness, assurance and reliability of the implementation. The obtained results of synthesis and simulation are presented in the following.

A. Encryption Results

Encryption test aims to verified the correctness and reliability of the implementation .The private key is applied internally without any need to add by users. The encryption mode is determined as shown in Figure (6) by sitting the selecting mode in the following state:

sele=1.

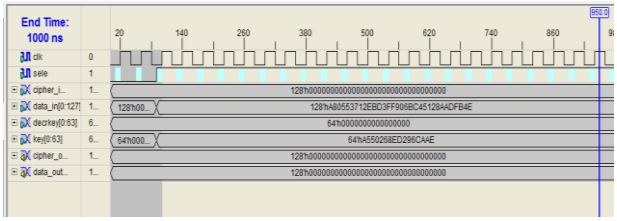

Fig .6. Encryption Mode .

Figure (7) illustrates the results of encryption stage using the previous inputs.

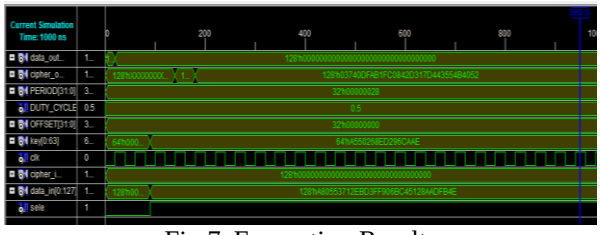

Fig.7. Encryption Results

B. Decryption Results

In public key encryption, a public key is made available to everyone. This key can encrypt messages, but it cannot decrypt them alone. The only person who can read a message is the recipient for whom it was encrypted. There are two cases can be studied when decoding is used, firstly when the recipient does not have the private key and secondly, the correct situation of decryption, when the recipient have the private key to decode the message.The following discusses the accuracy of decoding in the two cases in order to explain the benefit of using the private key:

Case1:

This test is done by using an incorrect private key(keydecr = incorrect value).The purpose of this simulation is to find out the level of secure reception that we can get it by using private key algorithm. The simulation inputs are shown in figure (8). Decryption mode is performed by using operation mode status as shown in the form (Sele=0).

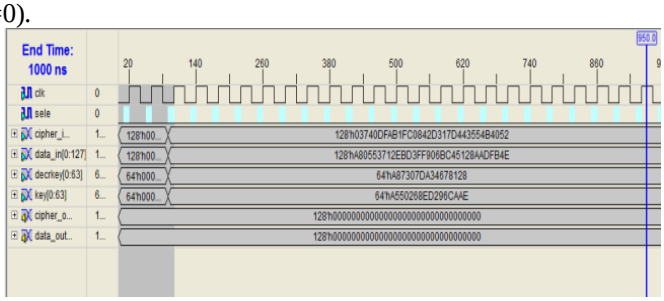

Fig.8. Decryption Mode (case1)

The results in figure (9) show that it is not possible for any person of deciphering the cipher without getting the correct private key, even if they were using the correct public key.

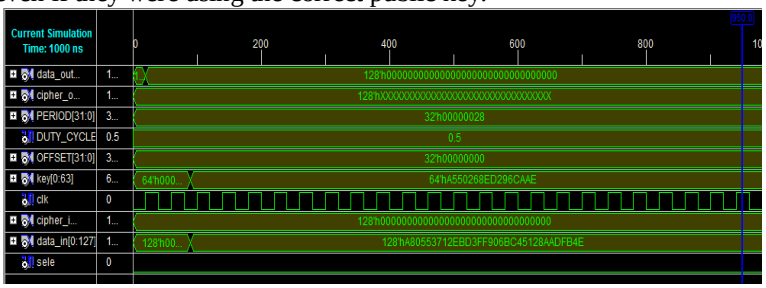

Fig .9. Decryption Results (case1)

Case2:

The purpose of this test is to prove the correctness and reliability for the output results. In this test ,simulation process is done for a correct decryption key (private key).At this case cipher is decoded correctly as shown in figure (10).

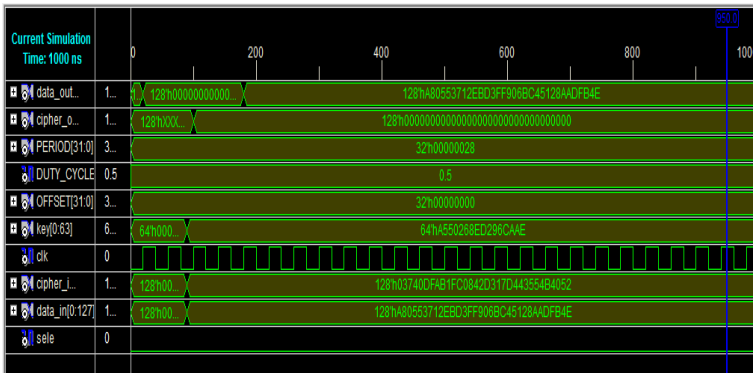

Fig .10. Decryption Results (case2)

In order to verify the output results, simulation performed by using the same output cipher of encryption in figure (9) as input. The results show that there is no doubt that the design gives a result of high accuracy and reliability.

C. Utilization Results

Synthesis is carried out using ISE9.2 and the target device is Xilinx FPGA Vertex5. TABLE I provides the obtained synthesis results. Synthesis is important operation to determine the efficiency of using the internal resources of the FPGA.

TABLE I: Utilization Results

Virtex5(target device xc5vlx110t package FF1738 speed-3)			
Logic utilization	**used**	**available**	**utilization**
Number of slices Registers	448	69120	0%
Number of slice LUTs	499	69120	0%
Number of fully used bit slices	360	587	61%
Number of occupied slices	328	17,280	1%
Number of bonded IOBs	642	680	94%
Number of BUFG/ $BUFGCTRL_S$	2	32	6%

The synthesis results show that the proposed design has an acceptable consumption area.

Timing Summary:

Maximum Frequency: 1045.260MHz

Minimum input arrival time before clock:

2.797ns

Maximum output required time after clock

2.703ns

The Delay Summary Report

The average connection delay for this design is: 2.016 ns

The maximum PIN delay is: 5.837ns

The average connection delay on the10 worst NETS is: 4.882 ns

TABLEII provides the comparison results of the proposed implementation and other related design.

TABLE II. Comparison results

Virtex 5 (xcv5lx110t)		
	our	Abduhadi[8]
Area utilization (%)	61%	84%

The comparison shows that the implementation is better in using the device internal resources than the implementation in abdulhadi [8].

IV. CONCLUSION

The provided design of public key cryptography or as it might be called as asymmetric cryptography presents a high security data transfer system. It provides authority to be included in the implementation by using a private key. The private key must be available only for the end user who should be entitled to receive the encrypted data.. The simulation results of implementation were verified and founded accurate while the synthesis results give that the design is efficient in using the resources of FPGA device.

REFERENCES

[1] Douglas.S,"Cryptography: Theory and Practice " ,CRC Press,1995.

[2] Dimitrios.M and Ioannis.P," Power consumption estimations vs measurements for FPGA-based security cores", in proceeding of International Conference on Reconfigurable Computing and FPGAs,pp 433- 437,Cancun ,Mexico,(2008).

[3] Gael.R, Francois.X.S, Jean.J. and Jean.D, "Compact and Efficient Encryption/Decryption Module for FPGA Implementation of AES", in Proceeding of International Conference on Information Technology: Coding and Computing, pp.339- 345, USA,(2004).

[4] http://www.icommcorp.com/downloads/Comparison AES vs 3DES.

[5] Wenbo. M, "Modern Cryptography Theory and Practice", Packard Company, Prentice Hall PTR,(2003).

[6] Prasun. G and Malabika.B," A Compact FPGA Implementation of Triple DES Encryption System with IP Core Generation and On-Chip Verification" , in Proceedings of International Conference on Industrial Engineering and Operations Management, Dhaka, Bangladesh,,(2010) .

[7] http://en.wikipedia.org/wiki/Public-key_cryptography.

[8] Abdulhadi.S,Thorsten.WGregor.M,Sorin.H and Falko.S,"ANovel Processor for McEliece Cryptosystem and FPGA platforms,international conference on application-specific Systems Architectures and processors',Boston,2009.