Design of AES Pipelined Architecture for Image Encryption/Decryption Module

Pravin V. Kinge[1], S.J. Honale[2], Prof. C.M. Bobade[3]
[1]Department of Electronics and Telecommunication Engineering, G.H. Raisoni College of Engineering, Amravati, India
[2,3]Faculty of Electronics and Telecommunication Engineering, G. H. Raisoni College of Engineering, Amravati, India

Article Info

Article history:

Received May 9, 2014
Revised Jul 23, 2014
Accepted Aug 12, 2014

Keyword:

AES
Ciphertext
FPGA
Plaintext
VHDL

ABSTRACT

The relentless growth of Internet and communication technologies has made the extensive use of images unavoidable. The specific characteristics of image like high transmission rate with limited bandwidth, redundancy, bulk capacity and correlation among pixels makes standard algorithms not suitable for image encryption. In order to overcome these limitations for real time applications, design of new algorithms that require less computational power while preserving a sufficient level of security has always been a subject of interest. Here Advanced Encryption Standard (AES), as the most widely used encryption algorithm in many security applications. AES standard has different key size variants, where longer bit keys provide more secure ciphered text output. The available AES algorithm is used for data and it is also suitable for image encryption and decryption to protect the confidential image from an unauthorized access. This project proposes a method in which the image data is an input to Pipelined AES algorithm through Textio, to obtain the encrypted image and the encrypted image is the input to Pipelined AES Decryption to get the original image. This project proposed to implement the 128,192 & 256 bit Pipelined AES algorithm for image encryption and decryption, also to compare the latency, efficiency, security, frequency & throughput. The proposed work will be synthesized and simulated on FPGA family of Xilink ISE 13.2 and Modelsim tool respectively in Very high speed integrated circuit Hardware Description Language.

Corresponding Author:

Pravin V. Kinge
PG Student, Department of Electronics and telecommunication Engineering,
G. H. Raisoni College of Engineering, Amravati
Email: Kinge.p.v@gmail.com

1. INTRODUCTION

In communication the data security is the big issue in various field so us government invited the new cryptography concept, ie AES algorithm yhe basic of AES Rijndael are in a mathematical concept called as Galois field theory. Similar to the way DES function, Rijndael also used the basic techniques of substitution and transposition (i.e. permutation). The key size and the plain text block size decide how many rounds need to be executed. One key differentiator between DES and provides for more optimized hardware and software implementation of the algorithm. AES algorithm has fix block size 128 bit and key size 128,192and 256 bit.

AES algorithm implemented by using hardware and software by using software it is easy to implemented the AES algorithm and it is easy low cost but it is not fully secured most secure. AES algorithm is applied data as well as image every image define in pixel concorn intensity value(digitel number) and location address in the form of row and column. The applications of the image processing have been commonly found in the Military communication, Forensics, Robotics, Intelligent systems etc. In this project, the Pipelined AES algorithm is proposed which is an efficient scheme for both hardware and software implementation.

AES algorithm

An encryption algorithm converts a plain text message into cipher text message which can be recovered only by authorized receiver using a decryption technique. The AES-Rijndael algorithm [4] is an iterative private key symmetric block cipher. The input and output for the AES algorithm each consist of sequences of 128 bits (block length). Hence Nb = Block length/32 = 4. The Cipher Key for the AES algorithm is a sequence of 128, 192 or 256 bits (Key length). In this implementation the key length to 128. Hence Nk = Key length/32 =4

initialization processes

- Expand the 16-byte key to get the actual key block to be used.
- Do one time initialization of the 16-byte plain text block(called as state).
- XOR the state with the key block.
- Apply s-box to each of the plain text bytes.
- Rotate row k of the plain text block(i.e. state) by k bytes.
- Perform a mix columns operation.
- XOR the state with the key block.

Encryption Process

The Encryption and decryption process consists of a number of different transformations applied consecutively over the data block bits, in a fixed number of iterations, called rounds. The number of rounds depends on the length of the key used for the encryption process. For key length of 128 bits, the number of iteration required are 10. (Nr = 10). As shown in Figure 1, each of the first Nr-1 rounds consists of 4 transformations: SubBytes(), ShiftRows(), MixColumns() & AddRoundKey().

Figure 1. AES Rijndael Describe step

There are four different transformations are described in detail below.

a) *Sub Bytes Transformation*:

It is a non-linear substitution of bytes that operates independently on each byte of the State using a substitution table (S box). This S-box which is invertible is constructed by first taking the multiplicative inverse in the finite field GF (2^8) with irreducible polynomial m(x) = x8 + x4+ x3 + x + 1. The element {00} is mapped to itself. Then affine transformation is applied (over GF (2)).

b) *Shift Rows Transformation:*

Cyclically shifts the rows of the State over different offsets. The operation is almost the same in the decryption process except for the fact that the shifting offsets have different values.

c) *Mix Columns Transformation:*

This transformation operates on the State column-by-column, treating each column as a four-term polynomial. The columns are considered as polynomials over GF (2^8) and multiplied by modulo x4 + 1 with a fixed polynomial a(x) = {03} x3+ {01} x2+ {02} x.

d) Add Round Key Transformation:

In this transformation, a Round Key is added to the State by a simple bitwise XOR operation. Each Round Key consists of Nb words from the key expansion. Those Nb words are each added into the columns of the State. Key Addition is the same for the decryption process.

Key Expansion:

Each round key is a 4-word (128-bit) array generated as a product of the previous round key, a constant that changes each round, and a series of S-Box lookups for each 32-bit word of the key. The Key schedule Expansion generates a total of Nb (Nr + 1) words.

The decryption process is direct inverse of the encryption process. All the transformations applied in encryption process are inversely applied to this process. Hence the last round values of both the data and key are first round inputs for the decryption process and follows in decreasing order.

2. RELATED WORK

The system makes AES key expansion which is used to generate multiple non-linear keys for the encryption process. This algorithm is suitable for image encryption in real time applications [1]. The data can be encrypted by 128 bit cipher key, through the use of cipher key with length 128, An efficient FPGA implementation of 128 bit block and 128 bit key AES algorithm has been presented [2]. They presented a low cost effective area cipher for encryption /decryption using 128 bit iterative architecture, after found that the amount of hardware resources has been optimize, One of the important Implementation of AES algorithm has been presented by Raneesha K, Rema Vellody and R nanda Kumar They compared two type of algorithm for speed of operation and observed that controller base approach [4]. Mg Suresh, Nataraj. K.R, concluded that the concept of Pipelined AES architecture can be practically implemented. It has been observed that the implementation of AES Encryption on the FPGA is successful and several data input. The AES algorithm is an iterative private key symmetric block cipher that can process data block of 128- bits through the use of cipher keys with key length 128,192 and 256 bits. An efficient FPGA implementation of 128 bit block and keys 128, 192 and 256 bits of AES –Rijindael algorithm has been presented [5].

3. WORKING

Initially the image is captured and then converted into the text file by using the MATLAB. This converted text file is used as a input through AES Textio and generate input.txt file, which is input to the VHDL code. The obtained file is encrypted by using the Advanced Encryption standard which is named as Ciphertext.txt this is again converted into image in MATLAB. The resultant image is Encrypted image. The Ciphertext.txt is now input to the AES Decryption to get original image. The AES is a symmetric key algorithm, in which both the sender and the receiver use a single key for encryption and decryption. AES defines the data block length to 128/192/256 bits. The image encryption and decryption of system is as shown in Figure 1.2.

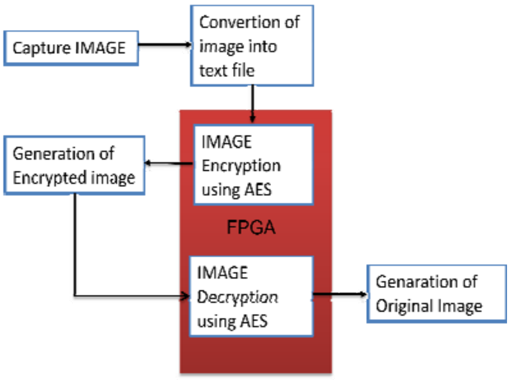
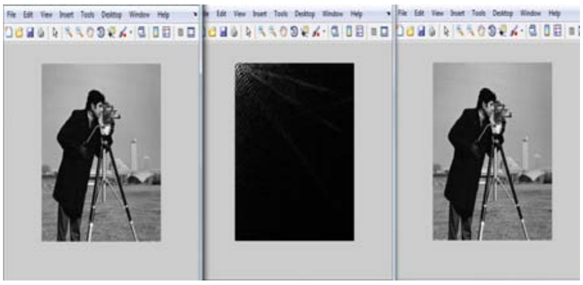

Capture IMAGE
Convertion of image into text file
Generation of Encrypted image
IMAGE Encryption using AES
FPGA
IMAGE Decryption using AES
Genaration of Original Image

Figure 1.2. System model for Image Encryption/Decryption

The receiver decrypts the encrypted image and gets the original image. At the receiver side image decryption is carried out by using AES itself .The key used for encryption and decryption must be the same. If they are different image will be lost. In decryption reverse of encryption is carried out. The input for decryption is the cipher text which is the output of AES encryption. This is then decrypted to get the output plain text. The output plain text is then converted to the original image by using MATLAB.

4. EXPERIMENTAL RESULT

Initially the image is captured and then converted into the text file by using the MATLAB. The converted text file is used as the input to AES Textio and generate ciphertext.txt file**.** This file is then input to MATLAB for showing Encrypted image.

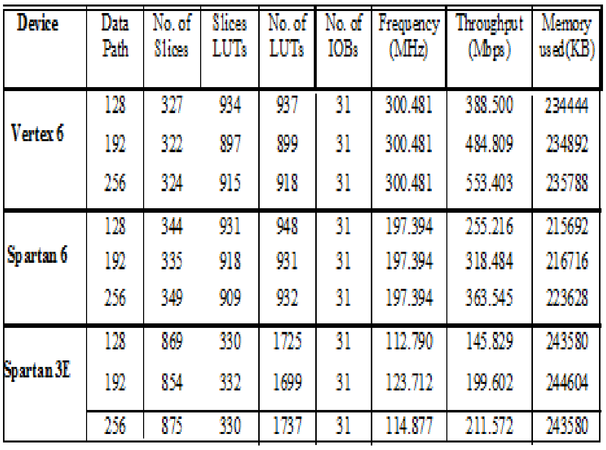


Original Image Encrypted Image Decrypted Image

For getting Original image decrypts the encrypted image through AES decryption Textio to get original image. At output side image decryption is carried out by using AES itself. The key used for encryption and decryption must be the same**.**

5. RESULT AND DISCUSSION

The described architecture was implemented in VHDL using the Model technology Modelsim simulator and synthesized, placed and routed using a target device of Xilinx (xilinx virtex XCV600E-6BG560, Spartan 6(XC6SLX25) and Spartan 3E starter kit FPGA). Four performances metrics such as the clocking frequency (MHz), the throughput (Mbps), the area (slices) and the total power consumption are computed. The results of the FPGA implementation are illustrated in following table



Device	Data Path	No. of Slices	Slices LUTs	No. of LUTs	No. of IOBs	Frequency (MHz)	Throughput (Mbps)	Memory used(KB)
Vertex 6	128	327	934	937	31	300.481	388.500	234444
	192	322	897	899	31	300.481	484.809	234892
	256	324	915	918	31	300.481	553.403	235788
Spartan 6	128	344	931	948	31	197.394	255.216	215692
	192	335	918	931	31	197.394	318.484	216716
	256	349	909	932	31	197.394	363.545	223628
Spartan 3E	128	869	330	1725	31	112.790	145.829	243580
	192	854	332	1699	31	123.712	199.602	244604
	256	875	330	1737	31	114.877	211.572	243580

From the above table it is show that our design is better than previous result. The main aim is to increase the throughput and decrease the latency of the AES so to increase the data rate and security.

6. CONCLUSION

An efficient subpipelined architecture of AES algorithm with its key expansion unit is presented. The key expansion architecture is suitable for 6 substages sub pipelined AES architecture Against prior implementations, this architecture uses composite field arithmetic in normal bases representation to reduce the required hardware. Image Encryption and Decryption using AES is designed and implemented to protect the confidential image data from an unauthorized access. A Successful implementation of AES algorithm is one of the best encryption and decryption standard available in market. It helps to explore the path to implement such an algorithm using VHDL code that is synthesized and simulated using the ISE 13.1 in Xilinx Family Spartan-6 (XC6SLX25), Vertex-6 & Spartan 3E. The Maximum Frequency achieved from the design is 385.239, 181.258 & 224.770 MHz and the throughput reaches the value of 1232.736, 580.02 & 719.264Mbit/sec for Encryption and Decryption.

The result shows that the design with the pipelining technology and special data transmission mode can optimize the chip area effectively. Meanwhile, this design reduces power consumption to some extent, for the power consumption is directly related to the chip area. Therefore the encryption device implemented in this method can meet some practical Applications like image encryption.

REFERENCES

[1] B. Subramanyan, Vivek. M. Chhabria, T.G. Sankar Babu, “Image Encryption Based On AES Key Expansion”, *Second International Conference on Emerging Applications of Information Technology, DOI 10.1.109/EAIT.2011.60, IEEE2011.*

[2] Hoang Trang, Nguyen Van Loi, “An efficient FPGA implementation of the advanced Encryption standard algorithm”, *978-1-4673-0309-5/12, IEEE 2012.*

[3] A. Amaar, I. Ashour and M Shiple, “Design and implementation a compact AES Architecture for FPGA Technology”, *World Academy of Science, Engineering and Technology 59, 2011.*

[4] Raneesha K, Rema Vellody and R nanda Kumar, “Hardware efficiency comparion of AES implementation”, *International Conference on Communication System and Network Technology. DOI 10.1109/CSNT.2012.187, IEEE 2012.*

[5] Mg Suresh, Dr. Nataraj. K.R, “Area Optimized and Pipelined FPGA Implementation of AES Encryption and Decryption”, *International Journal of Computational Engineering Research,* Vol. 2 Issue. 7, nov 2012

[6] National Institute of Standards and Technology (U.S.), "Data Encryption Standard (DES),"*FIPS Publication 46-3, NIST, 1999. Available at http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf*

[7] J. Yang, J. Ding, N. Li and Y.X. Guo, “FPGA-based design and implementation of reduced AES algorithm”, *IEEE Inter. Conf. Chal Envir Sci Com Engin (CESCE)., Vol. 02, Issue 5-6, pp. 67-70, Jun 2010.*

[8] National institute of standard and technology, “Federal information Procesing standaed publication 197, *the AES*”, Nov 2001.