ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementation of High Performance AES Using Minimal Resources

S.Bhanusree
M.Tech Student,
Department of ECE,
KITS for Women's, kodad, T.S, India.

Mr.B.Naresh Reddy
Associate Professor,
Department of ECE,
KITS for Women's, kodad, T.S, India.

ABSTRACT:

This paper presents a high speed, fully pipelined FPGA implementation of AES Encryption and Decryption (acronym for Advance Encryption Standard, also known as Rijndael Algorithm) which has been selected as New Algorithm by the National Institutes of Standards and Technology (NIST). The AES encryption & decryption algorithm is implemented on the FPGA. This has to have an interface with the PC. The C source for the encryption and decryption is already provided The algorithm is implemented to work in software and this is our baseline implementation. The application works in the following manner. The file is to be encrypted in software and transferred to the machine containing the FPGA. The file will be decrypted in hardware in that machine. Also the file is to be encrypted in hardware and decrypted in software. Our design mainly concentrates on the speed up along with silicon area optimization This implementation uses: Slices -1051 out of 33792 (3%), Slice Flip Flops - 1399 out of 67584 (2%), 4 input LUTs -1450 out of 67584 (2%), bonded IOBs - 390 out of 684 (57%) ,GCLKs -1 out of 16 (6%), Minimum period - 13.038ns (Maximum Frequency: 76.699MHz)

KEYWORDS:

AES, Cipher text, , FIPS, FPGA, Plaintext, pipelining.

INTRODUCTION:

Software implementation of AES algorithm is slower process (though easy). So the focal approach of our design on hardware platform is to attain speed up (i.e. high throughput No. of block processed per second) at the same time, silicon area optimization.

This paper is organized as follows. Section II provides brief overview of AES Algorithm. Section III focuses hardware architecture of our design. Section IV launches salient features. Section V discusses hardware software co-design. Section VI is endowed with experimental results on FPGA platform and finally section VII winds up with future work and conclusionThe main Objective of this project is to code a Data Encryption System using Advanced Encryption Standard (AES) Algorithm in Hardware Description Language and to test it according to a predetermined standard stimulus so that it meets requirements.

This standard specifies the Rijndael algorithm and, a symmetric block cipher that can process data blocks of 128 bits, using cipher keys with lengths of 128, 192, and 256 bits. Rijndael was designed to handle additional block sizes and key lengths, however they are not adopted in this standard. Throughout the remainder of this standard, the algorithm specified herein will be referred to as "the AES algorithm." The algorithm may be used with the three different key lengths indicated above, and therefore these different "flavors" may be referred to as "AES-128", "AES-192", and "AES-256".Security attacks against network are increasing significantly with time. Our communication media should also be secure and confidential. For this purpose, these three suggestions arrive in every one's mind:

(i) one can transmit the message secretly, so that it can be saved from hackers,
(ii) the sender ensures that the message arrives to the desired destination, and
(iii) the receiver ensures that the received message is in its original form and coming from the right sender. For this, one can use two techniques,:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

(i) one can use invisible ink for writing the message or can send the message through the confidential person, and
(ii) one can use a scientific approach called "Cryptography". Cryptography is the technique used to avoid unauthorized access of data. For example, data can be encrypted using a

PROPOSED ADVANCED: Encryption Standard

The AES algorithm is a round-based, symmetric block cipher. It processes data blocks of fixed size (128 bits) using cipher keys of length 128, 196 or 256 bits. Depending on the key used, it is usually abbreviated as AES-128, AES-196 or AES-256 respectively. In this project only AES-128 is considered, as it is the most popular variant of the algorithm. The functional blocks of the algorithm are Key expansion and encryption. In this project we are concentrating on the key generation algorithm. The initial 128-bit cipher key has to be expanded to new eleven round keys of same length. In order to produce a new round key, two transformations have to be performed, RotWord and SubWord. The first one simply cyclically shifts the bytes of the first 32-bit word ofthe previous key by one position to the left. SubWord on the other hand performs the SubBytes transformation to each byte ofthe rotated word. Simple bit wise xors are then needed in order to produce the final round key. The SubWord(SubBytes) transformation is implemented with a ROM (LUT).

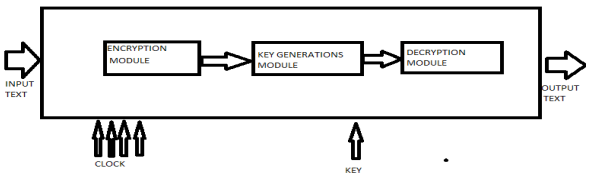


Fig 1 Architectured of Advanced Encryption Standard

The above figure (Fig No.2) shows the hardware I/O specification for the AES Encrypter and Decrypter. It requires a PLAIN / CIPHER TEXT which is of 128 bits length. Also KEY of length 128 bits (or 192 bits or 256 bits) is needed to be given. The control signals are START ENCRYPTION, START DECRYPTION, START KEY GENERATION, ENCRYPTION / DECRYPTION. These signals are used to control the proper sequencing of the modules. This controlling is done through a controller (not shown in the figure) which is interfaced with the Encrypter, Decrypter and the Key Expander.

The input data is processed using the key and the resultant data is available at the output terminal named CIPHER / PLAIN TEXT.The Key Expansion Module is common for both the Encrypted and Decrypted module.

AES Algorithm:

In cryptography, the Advanced Encryption Standard (AES), also known as Rijndael, is a block cipher adopted as an encryption standard by the US government. The cipher was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, and submitted to the AES selection process under the name "Rijndael", a portmanteau comprised of the names of the inventors. AES has a fixed block size of 128 bits and a key size of 128, 192 or 256 bits, whereas Rijndael can be specified with key and block sizes in any multiple of 32 bits, with a minimum of 128 bits and a maximum of 256 bits. AES operates on a 4×4 array of bytes, termed the state For encryption, each round of AES (except the last round) consists of four stages. a) SubBytes - a non-linear substitution step where each byte is replaced with another according to a lookup table (known as S Box). b) ShiftRows - a transposition step where each row of the state is shifted cyclically a certain number of steps. c) MixColumns - a mixing operation which operates on the columns of the state, combining the four bytes in each column using a linear transformation. d) AddRoundKey - each byte of the state is combined with the round key; each round key is derived from the cipher key using a key schedule. AES algorithm comprises of various rounds depending on the key size and blck size, Out of all the rounds the Pre- round comprises only AddRoundKey whereas the final round omits the MixColumns stage.

Number of rounds (Nr)	128-bit Data	192-bit Data	256-bit Data
128-bit Key	10	12	14
192-bit Key	12	12	14
256-bit Key	14	14	14

TABEL 1:In standard AES algorithm, there are four steps like SubByte, ShiftRow, MixColumn and Add Round Key in normal rounds. Our design highlights some following modifications:
i.Exclusion of Shift Row
ii.Pipelining for high Throughput
Optimizing the design to keep handy balance between Throughput and Silicon Area

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

EXCLUSION OF SHIFT ROW:

The ShiftRows step operates on the rows of the state; it cyclically shifts the bytes in each row by a certainoffset. For AES, the first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. For blocks of sizes 128 bits and 192 bits, the shifting pattern is the same. Row n is shifted left circular by n-1 bytes. In this way, each column of the output state of the ShiftRows step is composed of bytes from each column of the input state. (Rijndael variants with a larger block size have slightly different offsets). For a 256-bit block, the first row is unchanged and the shifting for the second, third and fourth row is 1 byte, 3 bytes and 4 bytes respectively—this change only applies for the Rijndael cipher when used with a 256-bit block, as AES does not use 256-bit blocks. The importance of this step is to avoid the columns being linearly independent, in which case, AES degenerates into four independent block ciphers.

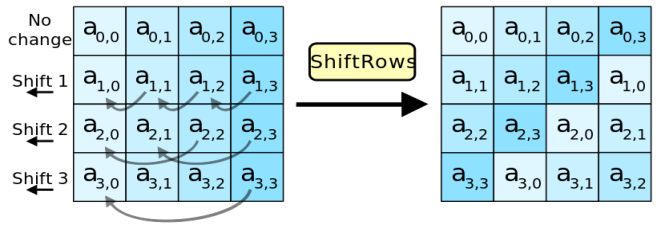


Fig2ShiftRows

PIPELINING FOR HIGH THROUGHPUT:

One focal constraint is that pipelining is only possible within each round (Not for whole rounds required for encryption/decryption). Next round can start only when previous round is totally completed as input data of the next round solely depend on the output of the previous rounds. So our design mainly concentrates on Pipelined Architecture Implementation of each round.

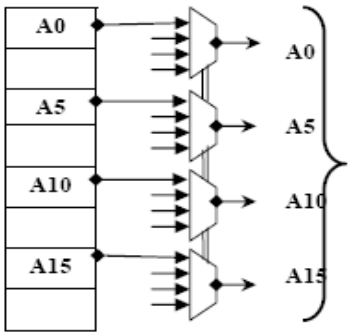


Fig3 exclusion of Shift Row is performed

OPTIMIZING THE DESIGN TO KEEP HANDY BALANCE BETWEEN THROUGH-PUT AND SILICON AREA:

Various types of hardware architectures for AES algorithm are possible. The best architecture is one which is having the best trade off between Clock Speed (data throughput) and Silicon Area. Firstly, our design had been tried to complete within 220 cycles. In this design, it is included only one S Box (Look Up Table). So at least 16 cycles are required to process 16 elements through S Box (assuming in each cycle one element is processed through S-Box Transformation). Each AES round is having 4 steps. Using pipelined structures within each round, total cycles required (for 10 round of AES Algorithm for 128 bit data & 128 bit key) is approx 220 cycles. Here silicon area is saved but losing high data throughput. In the extreme case, whole 10 rounds can be completed within 44 cycles using 16 SBox (LUT). In this case, though high throughputs are achieved, silicon area may be too much wasted.

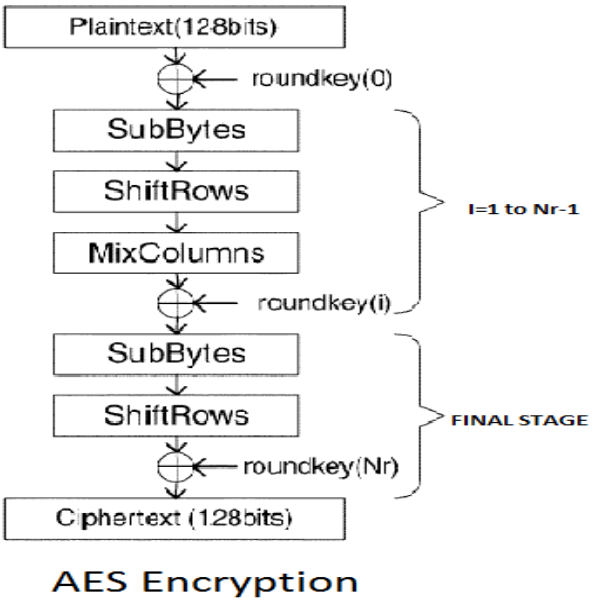


Fig4: aes encryption

The Encryption/Decryption process of Advanced Encryption Standard algorithm is presented below, in fig1. This block diagram is generic for AES specifications. It consists of a number of different transformations applied consecutively over the data block bits, in a fixed number of iterations, called rounds, where Nr = 10, 12 and 14. The number of rounds depends on the length of the key used for the encryption/Decryption process, and the blocks are described below.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

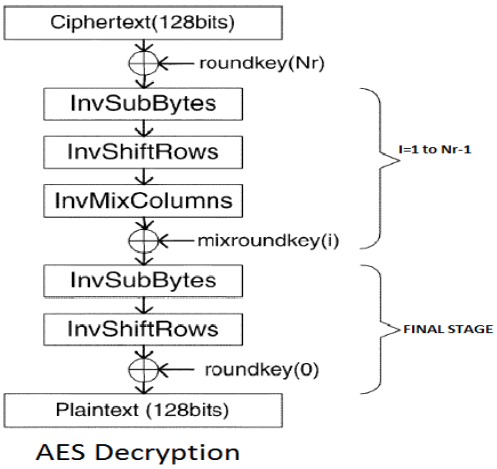


Fig5: Aes decryption

Bytes substitution The bytes substitution transformation Bytesub (state) is a non-linear substitution of bytes that operates independently on each byte of the State using a substitution table(S-box) and for Decryption inverse (S-Box).

Shift Rows :

In the Shift Rows transformation, the bytes in the last three rows of the State are cyclically shifted over different numbers of bytes. The first row is not shifted. The second row is left-shifted circularly one byte. For the third row, a 2-byte circular left shift is performed. For the fourth row, a 3-byte circular left shift is performed. And for the decryption process the cyclically shifting is to the right.

Mix Columns Transformation:

This transformation is based on Galois Field multiplication. Each byte of a column is replaced with another value that is a function of all four bytes in the given column. The Mix Columns transformation operates on the State column by column, treating each column as a four-term polynomial. The columns are considered as polynomials over GF (28) and multiplied modulo x4 + 1 with a fixed polynomial a(x), given by the following equation. a(x) = {03}x3+{01}x2+{01}x +{02}. This can be written as a matrix multiplication. Let S ' (x) = a(x) S(x)

AES AddRoundKey:

AddRoundKey operation is designed as a stream cipher; all the 128 bits of state are XORed with 4, 32-bit words of expanded key resulting from key expansion.

AddRound Key is the only operation that involves using the key to ensure security. The AES key expansion algorithm takes as input a four-word (16-byte) key and produces a linear array of 44 words (176 bytes). This is sufficient to provide a four word round key for the initial AddRoundKey stage and each of the 10 rounds of the cipher.

$$\begin{bmatrix} S'_{0,c} \\ S'_{0,c} \\ S'_{0,c} \\ S'_{0,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,c} \\ S_{0,c} \\ S_{0,c} \\ S_{0,c} \end{bmatrix}$$

For $0 \le c < Nb$. For decryption process the polynomial is,

$$\begin{bmatrix} S_{0C}{}' \\ S_{1C}{}' \\ S_{2C}{}' \\ S_{3C}{}' \end{bmatrix} = \begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} * \begin{bmatrix} S_{0C}{}' \\ S_{1C}{}' \\ S_{2C}{}' \\ S_{3C}{}' \end{bmatrix}$$

AES AddRoundKey :

AddRoundKey operation is designed as a stream cipher; all the 128 bits of state are XORed with 4, 32-bit words of expanded key resulting from key expansion. AddRoundKey is the only operation that involves using the key to ensure security. The AES key expansion algorithm takes as input a four-word (16-byte) key and produces a linear array of 44 words (176 bytes). This is sufficient to provide a four word round key for the initial AddRoundKey stage and each of the 10 rounds of the cipher. IMPLEMENTATION FPGA OF AES.

ENCRYPTION:

Encryption is the conversion of data into a form, called a cipher text, that cannot be easily understood by unauthorized people. Decryption is the process of converting encrypted data back into its original form, so it can be understood. The use of encryption/decryption is as old as the art of communication. In war time, a cipher, often incorrectly called a code, can be employed to keep the enemy from obtaining the contents of transmissions. (Technically, a code is a means of representing a signal without the intent of keeping

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

it secret; examples are Morse code and ASCII.) Simple ciphers include the substitution of letters for numbers, the rotation of letters in the alphabet, and the "scrambling" of voice signals by inverting the sideband frequencies. More complex ciphers work according to sophisticated computer algorithms that rearranges the data bits in digital signals. In order to easily recover the contents of an encrypted signal, the correct decryption key is required. The key is an algorithm that undoes the work of the encryption algorithm. Alternatively, a computer can be used in an attempt to break the cipher. The more complex the encryption algorithm, the more difficult it becomes to eavesdrop on the communications without access to the key.

Encryption/decryption is especially important in wireless communications. This is because wireless circuits are easier to tap than their hard-wired counterparts. Nevertheless, encryption/decryption is a good idea when carrying out any kind of sensitive transaction, such as a credit-card purchase online, or the discussion of a company secret between different departments in the organization. The stronger the cipher -- that is, the harder it is for unauthorized people to break it -- the better, in general. However, as the strength of encryption/decryption increases, so does the cost.

This means everyone who uses a cipher would be required to provide the government with a copy of the key. Decryption keys would be stored in a supposedly secure place, used only by authorities, and used only if backed up by a court order. Opponents of this scheme argue that criminals could hack into the key-escrow database and illegally obtain, steal, or alter the keys. Supporters claim that while this is a possibility, implementing the key escrow scheme would be better than doing nothing to prevent criminals from freely using encryption/decryption.

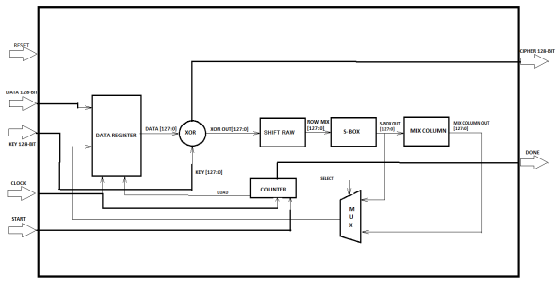


Fig6:Proposed architecture of encryption module

The AES encryption and decryption core unit contains key generation module as a common unit. This module gives necessary key expansion for both encryption and decryption functions. Fig.3 presents the block diagram of AES Rijndael encryption and decryption with Key Generation Module as a common unit. The key generation module consists of key register of 128 bits, S-Box and XOR gates for bitwise XOR peration.

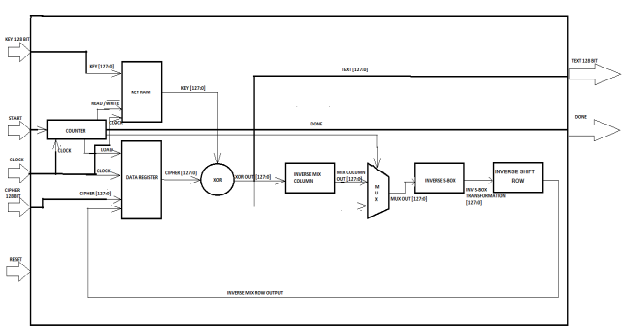


Fig7:Proposed architecture of Decryption module.

Inverse S-Box architecture uses the same design of S-Box. Entry of LUT is changed according to Inverse Sub Byte transformation. Mix Column operation is implemented using 256X8 ROM. Four such ROMs are designed for the Galois multiplication of 9, 11, 13 and 14. 4-Input XOR operation is designed by 16x1 ROM. Architecture of Decryption module is same as encryption module with all complimentary functions of encryption. Decryption unit contains an extra register for storing Round Keys. Storing key is important since first round decryption use tenth round key and second round use ninth round key and so on. Count register is synthesized as B-Ram to save number of slices. 'Count' input provides the address of key register location to be accessed. The Architecture of decryption module is shown in Fig.

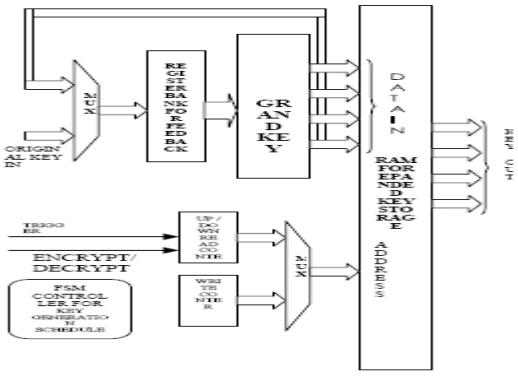


Fig8: produce round keys

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

It is designed to produce round keys on each positive edgeof the clock, when it is enabled. However in the proposedwork, the key generation architecture does not require anyhardware for shift operation and the port mapping betweenkey register and S-Box is done according to the required shift.Hence the proposed work offers the advantage in area. Also in the proposed work the bits are rearranged on data path from register to S-Box and the round constant required for eachrounds are stored in ROM and retrieved on each clockThe encryption module takes 128 bit text to be encryptedand receives round key from key generation module to doeach round of encryption

SALIENT FEATURES OF AES ENCRYPTION & DECRYPTION:

The salient features of the AES ENCRYPTION/DECRYPTION are summarized in the following manner:

HIGH THROUGHPUT:

For fully Pipelined implementation, area requirements increase with larger Key Size but throughput (No. of blocks processed per second) is unaffected.

PARAMETER FLEXIBILTY:

Any combination of Key sizes and Block sizes those are multiples of 32 bits can be accommodated. As a result, number of rounds can be modified.

IMPLENTATION FLEXIBILTY:

Decryption can be implemented in same structure as Encryption. (Though with different components)

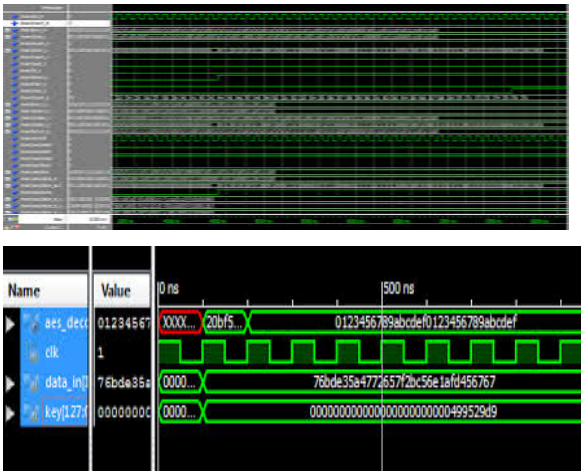
NO KNOWN SECURITY ATTACK:

Although it has received criticism due to its simple mathematical structure.

SIMULATION RESULTS:

AES Rijndael algorithm is simulated and synthesized using Xilinx 13.1 ISE tool and the targeted FPGA is 5vlx110tff1136-3 which belongs to Virtex-5 family. The design uses only LUTs, ROMs for all the operations of AES encryption and decryption.

This approach reduces device utilization and significantly improves the speed compared to other implementation .The key register in the decryption module is synthesized as Block-Ram to reduce the number of slices used.



Key Expansion : 60 cycles (approx)
Encryption : 90 cycles (approx)
Decryption : 80 cycles (approx

SYNTHESIS RESULTS: (Using Xilinx ISE 14.5.)

FUTURE WORK AND CONCLUSION

This AES algorithm can be parameterised by selection of cipher key bits (128,192 or 256).For higher throughput, 16 S-Box can be used completing whole processes around 44-50 cycles(at the same time,compromising the silicon area) Graphical Use Interface(GUI) can also be made which may be interacive wih the use.

ACKNOWLEDGMENTS:

I am S.Bhanusree and would like to thank the publishers, researchers for making their resources material available. I am greatly thankful to Associate Prof: Mr. B.Naresh Reddy for their guidance. We also thank the college authorities, PG coordinator and Principal for providing the required infrastructure and support. Finally, we would like to extend a heartfelt gratitude to friends and family members.

7.REFERENCES:

[1] NIST, “Advanced Encryption Standard (AES),” http://csrc.nist.-gov/publications/fips/fips197/fips-197.pdf, Nov. 2001.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[2] NIST, "Data Encryption Standard (DES)," http://csrc.nist.gov/ publications/fips/fips46-3/fips46-3.pdf, Oct. 1999.

[3] Joan Daemen Vincent Rijmen.AES Proposal:Rijdael
[4] Nicholas Weaver,John Wawizynek. Very High Performance Compact AES Implementations in Xilinx FPGAs

[5] Sounak Samamta.FPGA Implementation Of AES Encryption and Decryption

[6] Hoang Trang and Nguyen Van Loi HoChiMinh City, VietNam- "An efficient FPGA implementation of the Advanced Encryption Standard algorithm" (IEEE 2012)

[7] Yang Jun Ding Jun Li Na Guo Yixiong School of Information Science and Engineering, Yunnan University Kunming, China - " FPGA based design and implementation of reduced AES algorithm"(IEEE 2010).

[8] Adam J. Elbirt, W. Yip, B. Chetwynd, and C. Paar- "An FPGABased Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists" (IEEE 2001).

[9] WANG Wei, CHEN Jie & XU Fei, China-"An Implementation of AES Algorithm Based on FPGA" (IEEE2012).

Author's Details:

Ms.S.Bhanusree MTech student, in M.Tech Student, Dept of ECE in KITS for women's,kodad, T.S, India.

Mr.B.Naresh Reddy working as a Assosiate at ECE in KITS for women's,kodad, T.S, IndiaJNTUH Hyderabad. he has 6 years of UG/PG Teaching Experience.

.