International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 12, December 2014)

Hardware Implementation of AES Encryption and Decryption for Low Area & Low Power Consumption

B.S.S.VIDYA
Dept. name of organization (ECE)
Name of organization - acronyms acceptable
(PRAGATI ENGINEERING COLLEGE)
SURAMPALEM, E.G.DIST, AP. INDIA.

VENKATA SUBHA SREE GOPI
Dept. name of organization (ECE)
Name of organization - acronyms acceptable
(PRAGATI ENGINEERING COLLEGE)
SURAMPALEM, E.G.DIST, AP. INDIA.

***Abstract*-An AES algorithm is implemented on FPGA platform to improve the safety of data in transmission. AES algorithms can be implemented on FPGA in order to speed data processing and reduce time for key generating. We achieve higher performance by maintaining standard speed and reliability with low area and power. The 128 bit AES algorithm is implements on a FPGA using VHDL language with help of Xilinx tool.**

I INTRODUCTION

The main objective of this project is to code a Data Encryption System using Advanced Encryption Standard 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, a symmetric block cipher that can process data blocks of 128 bits, using cipher keys with lengths of 128, 192, and 256 bits. Security attacks against network are increasing significantly with time. (1) one can use invisible ink for writing the message or can send the message through the confidential person, and (2) one can use a scientific approach called “Cryptography”.
The process E of transforming a plaintext into a cipher is called encryption, while the opposite procedure D that turns a cipher text into a plaintext at the receiver side is said decryption. In symbols and shows in Fig 1

E (P) = C
D (C) =P

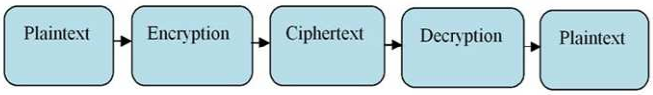


Fig1 The process of Encryption transforms Plaintext into Ciphertext and the process of Decryption transform Ciphertext into Plaintext.

II. ADVANCED ENCRYPTION STANDARD (AES)

AES is an iterated block cipher with a fixed block size of 128 and a variable key length. The different transformations operate on the intermediate results, called state. The state is a rectangular array of bytes and the block size is 128 bits, which is 16 bytes, the rectangular array is of dimensions 4x4. The block diagram is designed.

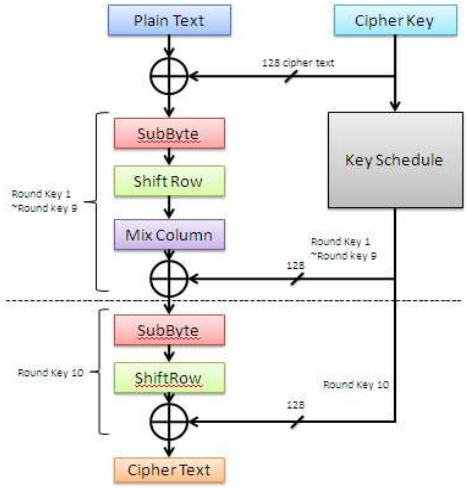


Fig 2 AES Algorithm Structure

The Advanced Encryption Standard (AES) is the United States Government standard for symmetric encryption, defined by AES is a block cipher that encrypts a 128-bit block (plaintext) to a 128-bit block (cipher text), or decrypts a 128-bit block (cipher text) to a 128-bit block (plaintext). AES uses a key (cipher key) whose length can be 128, 192, or 256 bits. Hence encryption /decryption with a cipher key of 128, 192, or 256 bits is denoted AES.

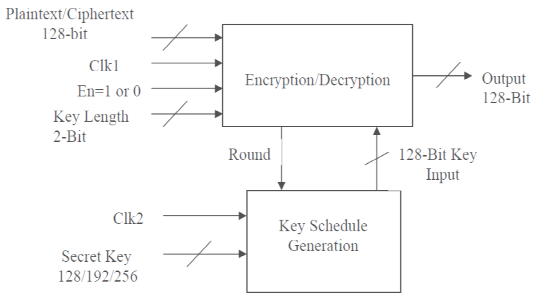


Fig3.Block Diagram of AES Architecture

The FSM, which is used in serial architecture loads sub-keys. It loads four sub-keys for each round of size 32-bit that was received from input as a 128-bit. To give input for

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 12, December 2014)

that round block we are using different multiplexers for different inputs. When to make selection line 1 and 0.Once again that is controlled by control FSM.

III. KEY GENERATION & ENCRYPTION MODULE

1. Key generation module
2. Encryption module
3. Decryption module

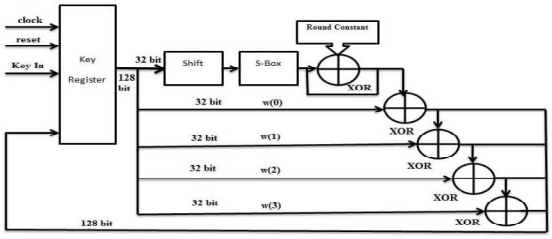


Fig 4Internal block diagram of Key Generation Module

The encryption module takes 128 bit text to be encrypts and receives round key from key generation module to do each round of encryption. Start, stop-mix, terminate are control signal produced by the control unit. The "done' signal is provide to indicate that encryption is done.

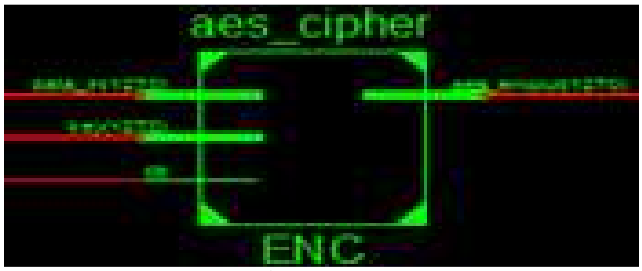


Fig 5 RTL of encryption Top Module

The control unit of key generation module which is a 4-bit counter is designed to control the entire function of encryption module. In the last round, rijandael algorithm skips Mix column operation. NAND gate and the 4-bit counter (controller) are used to set and reset selection line of multiplexer.

However on last round, count will be eleven so selection line will reset and pass sub byte output. After round key operation data is given to S-Box with require shift by port mapping the signal according to require shift in verilog HDL description of the design.

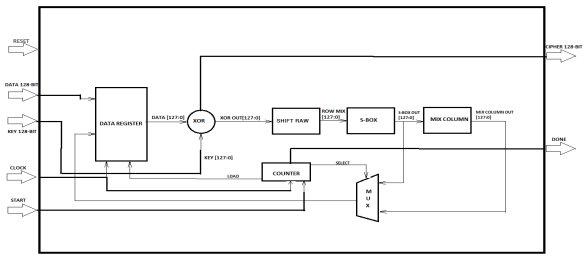


Fig 6 Internal block Diagram of encryption module

The mix column operation of AES consists of Galois multiplication and four input XOR operation. But unlike combinational implementation (8) of Galois field multiplication, for an 8-bit data there are 256 multiplication conditions and all the conditions are stored in (256 x 8) ROM. The mix column encryption hardware uses two of such ROM for Galois multiplication of '2' and '3' and for performing 4-input XOR operation.

IV. DECRYPTION MODULE

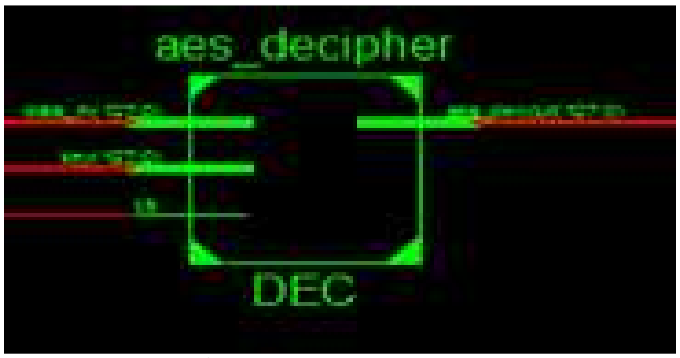


Fig 7 RTL of Decryption Top module

Block diagram 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.

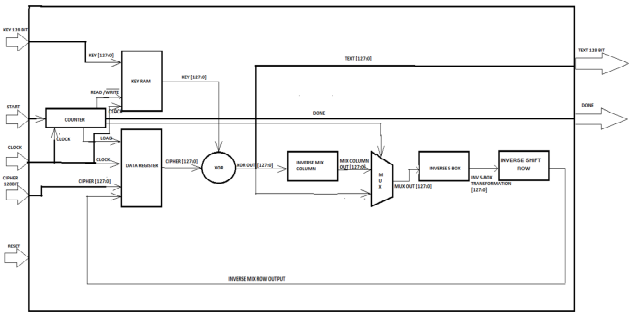


Fig 8 Internal block Diagram of decryption module

V. APPLICATIONS OF AES

You can use the AES in a variety of applications, including:

- Transfer funds electronically.
- E-mail.
- E-commerce (business conducted over the Internet).
- ATM machines.
- Cellular phones.
- Electronic financial transactions
- Secure communications
- Secure video surveillance systems
- Encrypted data storage

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 12, December 2014)

VI. ADVANTAGES

- *Simple*: This type of encryption is easy to carry out. All users have to do is specify and share the secret key and then begin to encrypt and decrypt messages.
- *Encrypt and decrypt your own files*: If you use encryption for messages or files which you alone intend to access, there is no need to create different keys. Single-key encryption is best for this.
- *Fast*: Symmetric key encryption is much faster than asymmetric key encryption.
- *Uses less computer resources*: Single-key encryption does not require a lot of computer resources when compared to public key encryption

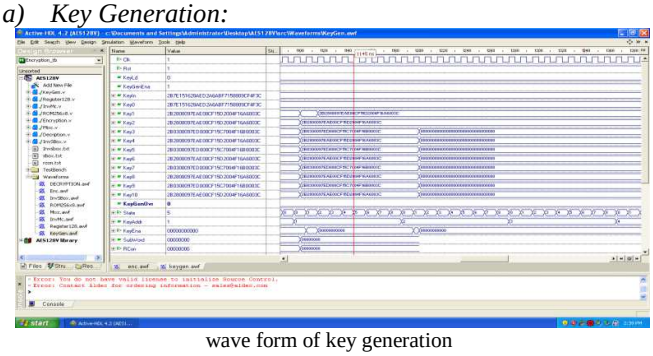
VII. DISADVANTAGES

Need for secure channel for secret key exchange: Sharing the secret key in the beginning is a problem in symmetric key encryption. It has to be exchanged in a way that ensures it remains secret.

Too many keys: A new shared key has to be generated for communication with every different party. This creates a problem with managing and ensuring the security of all these keys.

VIII. RESULTS

a) Key Generation:

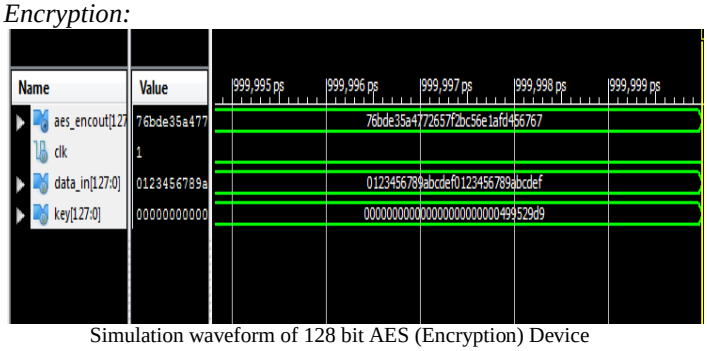


wave form of key generation

Device utilization summary for key Generation:
Selected Device: 4vsx25ff668-12
Number of Slices: 8 out of 10240 0%
Number of Slice Flip Flops: 8out of 20480 0%
Number of 4 input LUTs: 16 out of 20480 0%
Number of IOs: 1541
Number of bonded IOBs: 5 out of 320 1%
Number of GCLKs: 1 out of 32 3%

TIMING REPORT:
Timing Summary:
Speed Grade: -12
Minimum period: 1.549ns (Maximum Frequency: 645.619MHz)
Minimum input arrival time before clock: 2.370ns
Maximum output required time after clock: 4.518ns
Timing Detail:
All values displayed in nanoseconds (ns)
Timing constraint: Default period analysis for Clock 'Clk'

b) Encryption:

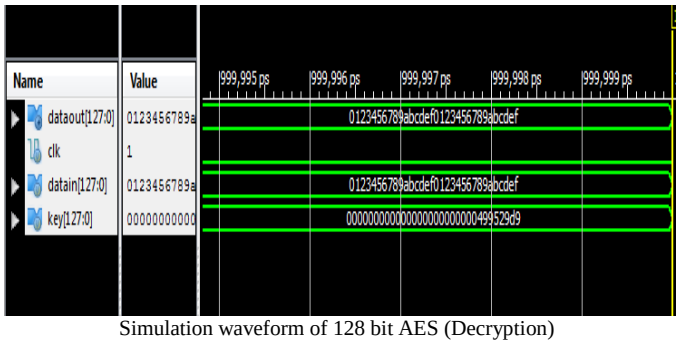


Simulation waveform of 128 bit AES (Encryption) Device

Utilization summary:
Selected Device: v50efg256-7
Number of Slices: 1006 out of 768 130% (*) Number of Slice Flip Flops: 1256 out of 1536 81%
Number of 4 input LUTs: 1859 out of 1536 121% (*)
Number of IOs: 518
Number of bonded IOBs: 518 out of 176 294% (*)
Number of GCLKs : 1 out of 4 25%

TIMING REPORT:
Timing Summary:
Minimum period: 9.902ns (Maximum Frequency: 100.990MHz)
Minimum input arrival time before clock: 15.021ns
Maximum output required time after clock: 10.050n

c) Decryption:



Simulation waveform of 128 bit AES (Decryption)

Device utilization summary 128 bit AES – Decryption block:
Selected Device: 4vsx25ff668-12
Number of Slices: 643 out of 10240 6%
Number of Slice Flip Flops: out of 20480 3%
Number of 4 input LUTs: 85 out of 20480 4%
Number of IOs: 1668
Number of bonded IOB: 260 out of 320 81%
Number of GCLKs:1 out of 32 3%

TIMING REPORT:
Timing Summary:

Speed Grade: -12
Minimum period: 2.681ns (Maximum Frequency: 372.977MHz)
Minimum input arrival time before clock: 3.937ns
Maximum output required time after clock: 5.123ns Maximum combinational path delay: No path found

IX. CONCLUSION

Advanced encryption standard is used now a day extensively in many network and multimedia applications to address security issues. Transmission and storage of sensitive data in open network environment is rapidly growing. It has high efficiency and high performance cipher core based on the multimode multiplier. As the internal block efficiently shares the hardware resources it saves more area and cost. This provides a comprehensive hardware solution for AES, offering high performance, software flexibility, and security against side channel attacks. AES has 10 rounds, meaning the main algorithm is repeated 10 times to produce the cipher text.

X. FUTURE SCOPE

AES operates with 128, 192 or 256 bit keys. These are considered long enough to be safe for the foreseeable future as they would take millions of millions of years to break on the fastest computers presently available. Recent encryptions have up to 256 bits of special keys so even a supercomputer would be slow in trying all the possible combinations. This of course ensures the security of data.

REFERENCES

[1] Chen-Hsing Wang, Chieh-Lin Chuang, and Cheng-Wen Wu An Efficient Multimode Multiplier Supporting AES and Fundamental Operations of Public-Key Cryptosystems. IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 18, NO. 4, APRIL 2010.
[2] M. Alam, S. Ray, D. Mukhopadhayay, S. Ghosh, D. RoyChowdhury, and I. Sengupta, “An area optimized reconfigurable encryptor for AESRijndael,” in Proc. Conf. DATE, Apr. 2007, pp. 1–6.
[3] Y.-K. Lai, L.-C. Chang, L.-F. Chen, C.-C. Chou, and C.-W. Chiu, “A novel memory less AES cipher architecture for networking applications,” in Proc. IEEE ISCAS, May 2004, pp. 333–336.
[4] C.-P. Su, T.-F. Lin, C.-T. Huang, and C.-W. Wu, “A high-throughput low-cost AES processor,” IEEE Commune. Mag., vol. 41, no. 12, pp. 86–91, Dec. 2003.
[5] C.-C. Lu and S.-Y. Tseng, “Integrated design of AES (Advanced Encryption Standard) encrypter and decrypter,” in Proc. IEEE Int. Conf. Appl.-Specific Syst. Architectures, Processors, Jul. 2002, pp. 277–285.
[6] A. F. Tenca and Ç. K. Koç, “A scalable architecture for modular multiplication based on Montgomery’s algorithm,” IEEE Trans. Compute., vol. 52, no. 9, pp. 1215–1221, Sep. 2003.